



Guía del usuario de la versión 1

# AWS Command Line Interface



# AWS Command Line Interface: Guía del usuario de la versión 1

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon, de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

---

# Table of Contents

.....	xiv
Acerca del AWS CLI .....	1
Acerca de la versión 1 de AWS CLI .....	2
Mantenimiento y soporte para las versiones principales SDK .....	2
Acerca de Amazon Web Services .....	2
Acerca de los ejemplos .....	3
Documentación y recursos adicionales .....	4
AWS CLI documentación y recursos .....	4
Otros y herramientas AWS SDKs .....	5
Instalar la AWS CLI .....	6
Requisitos de versión de Python .....	6
Amazon Linux .....	7
Requisitos previos .....	7
pip .....	8
yum .....	9
Solución de problemas de AWS CLI instalación y desinstalación .....	10
Linux .....	10
Requisitos previos .....	11
Instalación y desinstalación con el instalador incluido .....	11
Instalación y desinstalación con pip .....	17
Instalación y desinstalación con pip .....	20
Agregue el ejecutable de la AWS CLI versión 1 a la ruta de la línea de comandos .....	22
Solución de problemas de AWS CLI instalación y desinstalación .....	24
macOS .....	24
Requisitos previos .....	24
Instalación y desinstalación con el instalador incluido .....	25
Instalación o actualización con pip .....	30
Solución de problemas de AWS CLI instalación y desinstalación .....	34
Windows .....	34
Instale, actualice y desinstale mediante el MSI instalador .....	35
Instalación, actualización y desinstalación con Python y pip en Windows .....	37
Agregue el AWS CLI ejecutable a la ruta de la línea de comandos .....	39
Solución de problemas de AWS CLI instalación y desinstalación .....	40
Virtualenv .....	41

Requisitos previos .....	41
Instalación y actualización en un entorno virtual .....	41
Solución de problemas de AWS CLI instalación y desinstalación .....	43
Configure el AWS CLI .....	44
Prioridad de configuración y credenciales .....	44
Temas adicionales en esta sección .....	45
Opciones de los archivos de configuración y credenciales .....	45
Formato de los archivos de configuración y credenciales .....	46
¿Dónde se almacenan las opciones de configuración? .....	52
Usar perfiles con nombre .....	52
Definir y ver las opciones de configuración mediante comandos .....	53
Establecimiento de una nueva configuración y ejemplos de comandos de credenciales .....	55
Configuraciones del archivo config admitidas .....	57
Variables de entorno .....	74
Cómo configurar las variables de entorno .....	75
AWS CLI variables de entorno compatibles .....	76
Opciones de línea de comandos .....	87
Cómo utilizar las opciones de la línea de comandos .....	87
AWS CLI opciones de línea de comandos globales compatibles .....	88
Usos comunes de las opciones de la línea de comandos .....	91
Finalización de comandos .....	92
Funcionamiento .....	92
Configuración de la finalización de comandos en Linux o macOS .....	93
Configuración de la finalización de comandos en Windows .....	97
Reintentos .....	98
Modos de reintento disponibles .....	98
Configuración de un modo de reintento .....	101
Visualización de registros de reintentos .....	102
Usa un HTTP proxy .....	103
Uso de los ejemplos de la .....	103
Autenticación en un proxy .....	104
Uso de un proxy en las EC2 instancias de Amazon .....	105
Resolución de problemas .....	106
puntos de conexión .....	106
Establecer punto de conexión para un solo comando .....	106
Establezca un punto final global para todos Servicios de AWS .....	107

Está configurado para usar FIPs puntos finales para todos Servicios de AWS .....	108
Establecer para utilizar los puntos de conexión de doble pila para todos los Servicios de AWS .....	109
Establecer puntos de conexión específicos del servicio .....	110
Prioridad de la configuración y los ajustes del punto de conexión .....	114
Credenciales de autenticación y acceso .....	116
Prioridad de configuración y credenciales .....	116
Temas adicionales en esta sección .....	117
Credenciales a corto plazo .....	117
Roles de IAM .....	119
Requisitos previos .....	119
Descripción general del uso de IAM los roles .....	119
Configuración y uso de un rol .....	121
Usando MFA .....	123
Roles entre cuentas e ID externo .....	124
Especificación de un nombre de sesión de rol para simplificar las auditorías .....	125
Adopción de un rol con una identidad web .....	126
Borrado de las credenciales almacenadas en memoria caché .....	127
IAMusuarios .....	128
Paso 1: Crea tu IAM usuario .....	128
Paso 2: Obtener las claves de acceso .....	128
Configura el AWS CLI .....	129
Usa metadatos de EC2 instancias de Amazon para AWS CLI las credenciales .....	130
Requisitos previos .....	130
Configuración de un perfil para los EC2 metadatos de Amazon .....	131
Credenciales externas .....	132
Usa el AWS CLI .....	135
Obtención de ayuda .....	136
El comando de ayuda de AWS CLI integrado .....	136
AWS CLI guía de referencia .....	141
APIdocumentación .....	141
Solución de errores .....	142
Ayuda adicional .....	142
Estructura de comandos .....	142
Estructura de comandos .....	143
Comandos de espera .....	144

Especificar valores de parámetros .....	145
Tipos de parámetros comunes .....	146
Comillas con cadenas .....	151
Parámetros de archivos .....	156
Generar una plantilla de esqueleto de la CLI .....	159
Sintaxis abreviada .....	166
Salida del comando de control .....	168
Salida sensible .....	169
Opciones de salida del servidor frente a las del cliente .....	169
Output Format (Formato de salida) .....	170
Paginación .....	178
Filtrar salida de .....	181
Códigos de devolución .....	205
Alias .....	206
Requisitos previos .....	207
Paso 1: Creación del archivo de alias .....	207
Paso 2: Creación de un alias .....	208
Paso 3: Cómo llamar a un alias .....	211
Ejemplos de repositorio de alias .....	213
Recursos .....	215
Ejemplos de código .....	216
Ejemplos de comandos guiados .....	216
DynamoDB .....	217
Amazon EC2 .....	221
S3 Glacier .....	241
IAM .....	247
Amazon S3 .....	251
Amazon SNS .....	270
Ejemplos de comando .....	273
ACM .....	280
APIGateway .....	291
APIGateway HTTP y WebSocket API .....	350
APIAdministración de pasarelas API .....	397
App Mesh .....	399
App Runner .....	445
AWS AppConfig .....	480

Aplicación de escalado automático .....	516
Application Discovery Service .....	534
AppRegistry .....	541
Athena .....	553
Auto Scaling .....	588
Planes de Auto Scaling .....	659
AWS Backup .....	667
AWS Batch .....	674
AWS Budgets .....	689
Amazon Chime .....	700
Control en la nube API .....	776
AWS Cloud Map .....	782
AWS Cloud9 .....	792
AWS CloudFormation .....	801
CloudFront .....	853
Amazon CloudSearch .....	923
CloudTrail .....	924
CloudWatch .....	941
CloudWatch Registros .....	956
CloudWatch Supervisión de redes .....	962
CodeArtifact .....	975
CodeBuild .....	1005
CodeCommit .....	1070
CodeDeploy .....	1147
CodeGuru Revisor .....	1189
CodePipeline .....	1208
AWS CodeStar .....	1241
AWS CodeStar Notificaciones .....	1255
CodeConnections .....	1267
Amazon Cognito Identity .....	1275
Amazon Cognito Identity Provider .....	1281
Amazon Comprehend .....	1350
Amazon Comprehend Medical .....	1490
AWS Config .....	1526
Amazon Connect .....	1551
AWS Cost and Usage Report .....	1569

---

Servicio Cost Explorer .....	1571
Firehose .....	1580
Administrador de vida útil de datos de Amazon .....	1583
AWS Data Pipeline .....	1589
DataSync .....	1599
DAX .....	1603
Detective .....	1623
Device Farm .....	1635
AWS Direct Connect .....	1640
AWS Directory Service .....	1693
AWS DMS .....	1696
Amazon DocumentDB .....	1741
DynamoDB .....	1799
DynamoDB Streams .....	1897
Amazon EC2 .....	1905
Amazon EC2 Instance Connect .....	2590
Amazon ECR .....	2591
Amazon ECS .....	2623
Amazon EFS .....	2711
Amazon EKS .....	2720
Elastic Beanstalk .....	2800
Elastic Load Balancing: versión 1 .....	2831
Elastic Load Balancing: versión 2 .....	2860
Elastic Transcoder .....	2915
ElastiCache .....	2942
MediaStore .....	3050
Amazon EMR .....	3068
Amazon EMR en EKS .....	3118
EventBridge .....	3119
Firewall Manager .....	3126
AWS FIS .....	3137
Amazon GameLift .....	3156
Global Accelerator .....	3190
AWS Glue .....	3230
GuardDuty .....	3252
AWS Health .....	3271



HealthImaging .....	3278
HealthLake .....	3302
HealthOmics .....	3314
IAM .....	3385
Analizador de acceso de IAM .....	3530
Image Builder .....	3566
Incident Manager .....	3609
Contactos de Incident Manager .....	3632
Amazon Inspector .....	3656
AWS IoT .....	3688
AWS IoT 1-Click Dispositivos .....	3877
AWS IoT 1-Click Proyectos .....	3887
AWS IoT Analytics .....	3899
Asesor de dispositivos .....	3927
AWS IoT data .....	3943
AWS IoT Events .....	3946
AWS IoT Events-Data .....	3972
AWS IoT Greengrass .....	3998
AWS IoT Greengrass V2 .....	4088
AWS IoT Jobs SDK release .....	4114
AWS IoT SiteWise .....	4118
AWS IoT Things Graph .....	4169
AWS IoT Wireless .....	4198
Amazon IVS .....	4236
IVSChat de Amazon .....	4277
Transmisión IVS en tiempo real de Amazon .....	4291
Amazon Kendra .....	4317
Kinesis .....	4326
AWS KMS .....	4346
Lake Formation .....	4417
Lambda .....	4470
License Manager .....	4513
Lightsail .....	4527
Macie .....	4655
Amazon Managed Grafana .....	4660
MediaConnect .....	4662

---

MediaConvert .....	4678
MediaLive .....	4704
MediaPackage .....	4710
MediaPackage VOD .....	4726
MediaStore Plano de datos .....	4738
MediaTailor .....	4744
MemoryDB .....	4749
Amazon MSK .....	4787
Administrador de red .....	4796
Nimble Studio .....	4835
OpenSearch Servicio .....	4854
AWS OpsWorks .....	4868
AWS OpsWorks CM .....	4927
Organizations .....	4943
AWS Outposts .....	4982
AWS Payment Cryptography .....	4986
AWS Payment Cryptography Plano de datos .....	5007
Amazon Pinpoint .....	5017
Amazon Polly .....	5041
Lista de precios de AWS .....	5047
AWS Private CA .....	5052
AWS Proton .....	5060
QLDB .....	5073
Amazon RDS .....	5097
Amazon RDS Data Service .....	5300
Amazon RDS Performance Insights .....	5304
Amazon Redshift .....	5308
Amazon Rekognition .....	5391
AWS RAM .....	5468
Resource Explorer .....	5493
Resource Groups .....	5516
Etiquetado de Resource Groups API .....	5529
AWS RoboMaker .....	5533
Route 53 .....	5570
Registro de dominios de Route 53 .....	5585
Route 53 Resolver .....	5612

Amazon S3 .....	5658
Amazon S3 Control .....	5754
S3 Glacier .....	5770
Secrets Manager .....	5793
Security Hub .....	5822
AWS Serverless Application Repository .....	5904
Service Catalog .....	5906
Service Quotas .....	5938
Amazon SES .....	5949
Escudo .....	5962
Signer .....	5979
Snowball .....	5989
Amazon SNS .....	5991
Amazon SQS .....	6013
Storage Gateway .....	6035
AWS STS .....	6038
AWS Support .....	6047
Amazon SWF .....	6061
Systems Manager .....	6078
Amazon Textract .....	6259
Amazon Transcribe .....	6270
Amazon Translate .....	6315
Trusted Advisor .....	6317
Permisos verificados .....	6337
VPCEnrejado .....	6364
AWS WAF Classic .....	6393
AWS WAF Classic Regional .....	6398
AWS WAFV2 .....	6405
Amazon WorkDocs .....	6451
Amazon WorkMail .....	6484
Flujo de WorkMail mensajes de Amazon .....	6509
WorkSpaces .....	6511
X-Ray .....	6527
Ejemplos de script Bash .....	6545
DynamoDB .....	6546
Amazon EC2 .....	6618

HealthImaging .....	6725
IAM .....	6734
Amazon S3 .....	6789
AWS STS .....	6813
Seguridad .....	6817
Protección de los datos .....	6818
Cifrado de datos .....	6819
Identity and Access Management .....	6819
Público .....	6820
Autenticación con identidades .....	6820
Administración de acceso mediante políticas .....	6824
¿Cómo Servicios de AWS trabajar con IAM .....	6827
Solución de problemas AWS de identidad y acceso .....	6827
Validación de la conformidad .....	6829
Resiliencia .....	6830
Seguridad de infraestructuras .....	6831
Imponer una versión mínima TLS .....	6832
Errores de solución de problemas .....	6836
Solución de problemas generales para probar primero .....	6836
Comprueba el formato de tus AWS CLI comandos .....	6837
Compruebe si Región de AWS su AWS CLI comando está utilizando .....	6837
Confirme que está ejecutando una versión reciente de la AWS CLI .....	6838
Use la opción --debug .....	6838
Habilite y revise los registros del historial de AWS CLI comandos .....	6844
Confirme que el suyo está configurado AWS CLI .....	6844
Errores de comando no encontrado .....	6845
El comando "aws --version" devuelve una versión diferente a la instalada .....	6848
El comando "aws --version" devuelve una versión tras desinstalar la AWS CLI .....	6849
AWS CLI Procesó un comando con un nombre de parámetro incompleto .....	6850
Errores de acceso denegado .....	6851
Errores de credenciales y claves no válidas .....	6852
Errores de firma que no coincide .....	6854
Errores de no se ha encontrado la consola de Windows .....	6856
SSLerrores de certificado .....	6856
JSONErrores no válidos .....	6857
Recursos adicionales de .....	6859

---

Historial de revisión ..... 6860

Esta documentación es AWS CLI únicamente para la versión 1 de la versión. Para ver la documentación relacionada con la versión 2 de AWS CLI, consulte la [Guía del usuario de la versión 2](#).

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la version original de inglés, prevalecerá la version en inglés.

# ¿Cuál es la versión 1 de AWS Command Line Interface?

## Note

La AWS CLI versión 1 no es la última versión de AWS CLI. Algunas características introducidas en la versión 2 de AWS CLI no son compatibles con la versión 1 y debe realizar una actualización para acceder a esas características. Existen algunos cambios “notorios” de la versión 1 que pueden requerir que se modifiquen los scripts. Para obtener una lista de los cambios notorios de la versión 2, consulte [Cambios notorios](#) en la versión 2 de la guía del usuario de AWS CLI .

The AWS Command Line Interface (AWS CLI) es una herramienta de código abierto que le permite interactuar con AWS los servicios mediante comandos del shell de la línea de comandos. Con una configuración mínima, AWS CLI permite empezar a ejecutar comandos que implementan una funcionalidad equivalente a la proporcionada por el navegador AWS Management Console desde la línea de comandos del programa de terminal:

- Shells de Linux: utilice programas de shell comunes, tales como [bash](#), [zsh](#) y [tcsh](#), para ejecutar comandos en Linux o macOS.
- Línea de comandos de Windows: en Windows, ejecute los comandos en la línea de comandos de Windows o en PowerShell
- De forma remota: ejecute comandos en instancias de Amazon Elastic Compute Cloud (AmazonEC2) a través de un programa de terminal remoto como PuTTY o SSH, o with AWS Systems Manager.

Todas las funciones de AWS administración, gestión y acceso de IaaS (infraestructura como servicio) AWS Management Console están disponibles en cualquier momento. AWS API AWS CLI Las nuevas funciones y servicios de AWS IaaS proporcionan una AWS Management Console funcionalidad completa durante API y CLI en el momento del lanzamiento o dentro de los 180 días posteriores al lanzamiento.

AWS CLI Proporciona acceso directo al público APIs de los AWS servicios. Puede explorar las capacidades de un servicio con el AWS CLI y desarrollar scripts de shell para administrar sus recursos. Además de los comandos API equivalentes de bajo nivel, varios AWS servicios ofrecen

personalizaciones para el. AWS CLI Las personalizaciones pueden incluir comandos de nivel superior que simplifican el uso de un servicio con un complejo. API

## Acerca de la versión 1 de AWS CLI

La AWS CLI versión 1 es la original AWS CLI y seguimos ofreciéndola. Sin embargo, es posible que las principales funciones nuevas que se introduzcan en la AWS CLI versión 2 no estén incorporadas a la AWS CLI versión 1. Para utilizar esas funciones, debe instalar la AWS CLI versión 2. La AWS CLI versión 1 se creó con Python y, SDK por lo tanto, requiere que instale una versión compatible de Python.

Para instalar la AWS CLI versión 1, consulte [Instalar la AWS CLI](#).

Para comprobar la versión instalada actualmente, utilice el siguiente comando:

```
$ aws --version
aws-cli/1.33.33 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```

Para ver el historial de versiones, consulte el registro de [cambios de la AWS CLI versión 1 en GitHub](#)

## Mantenimiento y soporte para las versiones principales SDK

Para obtener información sobre el mantenimiento y el soporte de las versiones SDK principales y sus dependencias subyacentes, consulte lo siguiente en la [Guía de referencia de las herramientas AWS SDKs y herramientas](#):

- [AWS SDKs y política de mantenimiento de herramientas](#)
- [AWS SDKs matriz de soporte de versiones y herramientas](#)

## Acerca de Amazon Web Services

Amazon Web Services (AWS) es una colección de servicios de infraestructura digital que los desarrolladores pueden usar cuando desarrollan sus aplicaciones. Los servicios incluyen la sincronización de computación, almacenamiento, bases de datos y aplicaciones (mensajería y colas). AWS utiliza un modelo pay-as-you-go de servicio. Solo se le cobrará por los servicios que usted, o sus aplicaciones, utilicen. Además, para hacerla AWS más accesible como plataforma para la creación de prototipos y la experimentación, AWS ofrece un nivel de uso gratuito. En esta capa, los



servicios son gratuitos por debajo de determinado nivel de uso. Para obtener más información sobre AWS los costos y la capa gratuita, consulte la capa [AWS gratuita](#). Para obtener una AWS cuenta, abra la [página de AWS inicio](#) y, a continuación, seleccione Crear una AWS cuenta.

## Acerca de los AWS CLI ejemplos

Los ejemplos AWS Command Line Interface (AWS CLI) de esta guía están formateados según las siguientes convenciones:

- **Símbolo:** el símbolo del sistema utiliza el símbolo de Linux y se muestra como (`$` ). En los comandos específicos de Windows, `C:\>` se utiliza como símbolo del sistema. No incluya el símbolo al escribir comandos.
- **Directorio:** cuando los comandos se deben ejecutar desde un directorio específico, el nombre del directorio se muestra antes del símbolo de comando.
- **Entrada del usuario:** el texto del comando que se debe ingresar en la línea de comando se formatea como **user input**.
- **Texto reemplazable:** el texto variable, incluidos los nombres de los recursos que elija o IDs generado por AWS los servicios que debe incluir en los comandos, tiene el formato siguiente *replaceable text*. En los comandos de varias líneas o en los comandos en los que se requiere una entrada específica del teclado, los comandos del teclado también se pueden mostrar como texto reemplazable.
- **Salida:** la salida devuelta por AWS los servicios se muestra debajo de la entrada del usuario y tiene el formato siguiente. `computer output`

El siguiente ejemplo de comando **aws configure** demuestra la entrada del usuario, el texto reemplazable, y la salida:

1. Escriba **aws configure** en la línea de comandos y luego presione INTRO.
2. Muestra AWS CLI líneas de texto en las que se le pide que introduzca información adicional.
3. Introduzca cada una de sus claves de acceso y después pulse Intro.
4. A continuación, introduzca el nombre de una AWS región en el formato que se muestra, pulse Entrar y, a continuación, pulse Entrar por última vez para omitir la configuración del formato de salida.
5. El comando Intro final se muestra como texto reemplazable porque no hay entrada del usuario para esa línea.

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: ENTER
```

El siguiente ejemplo muestra un comando simple con salida. Para usar este ejemplo, introduzca el texto completo del comando (el texto resaltado después del símbolo del sistema) y pulse Intro. El nombre del grupo de seguridad, *my-sg*, se puede reemplazar por el nombre del grupo de seguridad que desee. Se imprime el JSON documentado, incluidos los corchetes. Si configura la salida en formato de texto o tabla, la salida tendrá un formato diferente. CLI [JSON](#) es el formato de salida predeterminado.

```
$ aws ec2 create-security-group --group-name my-sg --description "My security group"
{
  "GroupId": "sg-903004f8"
}
```

## Documentación y recursos adicionales

### AWS CLI documentación y recursos

Además de esta guía del usuario, los siguientes son valiosos recursos en línea cuando utilice el AWS CLI.

- [AWS CLI guía de referencia de la versión 1](#)
- AWS CLI Repositorio de [ejemplos de código de secuencias de comandos en Bash](#). Ejemplos de scripts bash de código abierto. Los ejemplos de secuencias de comandos de Bash están alojados en el repositorio de ejemplos de [AWS código](#) en GitHub
- [AWS CLI GitHub repositorio](#). Puede ver y bifurcar el código fuente de AWS CLI on GitHub. Únase a la comunidad de usuarios GitHub para enviar comentarios, solicitar funciones y enviar sus propias contribuciones. Esto incluye ver y proporcionar ejemplos de comandos para la AWS CLI documentación.
- [AWS CLI repositorio de ejemplos de alias](#) Puede ver y bifurcar ejemplos de AWS CLI alias en el GitHub.

- [AWS CLI registro de cambios de la versión 1](#)
- [AWS CLI registro de cambios de la versión 2](#)

## Otros y herramientas AWS SDKs

Según su caso de uso, es posible que desee elegir una de las AWS SDKs herramientas que mejor se adapte a sus necesidades:

- [AWS SDKs y guía de referencia de herramientas](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK para Kotlin](#)
- [AWS SDK for .NET](#)
- [AWS SDK for Python \(Boto\)](#)
- [AWS SDK for PHP](#)
- [AWS Tools for PowerShell](#)
- [AWS SDK for Ruby](#)
- [AWS SDK para Rust](#)
- [AWS SDK para SAP ABAP](#)
- [AWS SDK para Swift](#)
- [AWS Amplify](#)

# Instalar, actualizar y desinstalar la AWS CLI

En este tema, se proporcionan enlaces para instalar, actualizar y desinstalar la versión original de AWS Command Line Interface (AWS CLI). La versión 1 de AWS CLI es compatible actualmente, pero se agregan nuevas características a la versión 2 de AWS CLI que puede que no se agreguen a la versión 1 de AWS CLI. Para utilizar estas características, debe instalar la versión 2 de la AWS CLI. Para obtener información sobre cómo instalar la versión 2, consulte [Instalación de la versión 2 de AWS CLI](#).

Instrucciones de instalación, actualización y desinstalación de AWS CLI:

- [Requisitos de versión de Python](#)
- [Instalación, actualización y desinstalación de la AWS CLI versión 1 en Amazon Linux](#)
- [Instalación, actualización y desinstalación de la AWS CLI versión 1 en Linux](#)
- [Instalar, actualizar y desinstalar la AWS CLI versión 1 en macOS](#)
- [Instalar, actualizar y desinstalar la AWS CLI versión 1 en Windows](#)
- [Instalar y actualizar la AWS CLI versión 1 en un entorno virtual](#)

## Requisitos de versión de Python

La versión 1 de la AWS CLI se construye usando el SDK para Python y, por lo tanto, requiere que instale una versión compatible de Python.

Matriz de compatibilidad con versiones de Python

Version de la AWS CLI	Versión de Python compatible
1.32.0: actual	Python 3.8+
1.27.0: 1.31.x	Python 3.7+
1.20.0 – 1.26.x	Python 3.6+
1.19.0 – 1.19.x	Python: Python 3.6 y 2.7
1.17 – 1.18.x	Python 2.7+, Python 3.4+

Version de la AWS CLI	Versión de Python compatible
1.0 – 1.16.x	Python 2.6 y versiones anteriores, Python 3.3 y versiones anteriores

Para obtener información sobre los lanzamientos más recientes de AWS CLI, consulte el [registro de cambios de la versión 2 de la AWS CLI](#) en GitHub.

## Instalación, actualización y desinstalación de la AWS CLI versión 1 en Amazon Linux

La AWS CLI versión 1 viene preinstalada en Amazon Linux y Amazon Linux 2. Consulte la versión instalada actualmente utilizando el siguiente comando.

```
$ aws --version
aws-cli/1.33.33 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```

En función de cuándo haya creado la instancia de Amazon Linux, la AWS CLI versión 1 viene preinstalada mediante uno de los siguientes administradores de paquetes:

- [pip](#)
- [yum](#)

### Requisitos previos

Debe tener instalado Python 3.8 o una versión posterior. Para obtener instrucciones de instalación, consulte la página [Downloading Python](#) en la Guía para principiantes de Python.

Matriz de compatibilidad con versiones de Python

AWS CLI versión	Versión de Python compatible
1.32.0: actual	Python 3.8+
1.27.0: 1.31.x	Python 3.7+
1.20.0 – 1.26.x	Python 3.6+

AWS CLI versión	Versión de Python compatible
1.19.0 – 1.19.x	Python: Python 3.6 y 2.7
1.17 – 1.18.x	Python 2.7+, Python 3.4+
1.0 – 1.16.x	Python 2.6 y versiones anteriores, Python 3.3 y versiones anteriores

## Instalación, actualización o desinstalación con pip

La mayoría de las instancias de Amazon Linux utilizan pip para preinstalar la AWS CLI versión 1.

### Instale o actualice la AWS CLI versión 1 en Amazon Linux mediante pip

Para instalar la última versión de la AWS CLI versión 1 para el usuario actual, siga las siguientes instrucciones.

1. Si tiene instalada la versión 3 o posterior de Python, recomendamos que utilice pip3. pip3 install Úselo para instalar o actualizar a la última versión de la AWS CLI versión 1. Si ejecuta el comando desde un [entorno virtual de Python \(venv\)](#), no necesita utilizar la opción --user.

```
$ pip3 install --upgrade --user awscli
```

2. Asegúrese de que la carpeta que contiene awsforme parte de la variable PATH.
  - a. Busque el script de perfil de su shell en su directorio de usuario. Si no está seguro de cuál es el shell que tiene, ejecute echo \$SHELL.

```
$ ls -a ~  
.  ..  .bash_logout  .bash_profile  .bashrc  Desktop  Documents  Downloads
```

- Bash: .bash\_profile, .profile o .bash\_login.
- Zsh: .zshrc
- Tcsh: .tcshrc, .cshrc o .login.

- b. Añada un comando de exportación al final del script de su perfil igual que en el siguiente ejemplo.

```
export PATH=$HOME/.local/bin:$PATH
```

Este comando inserta la ruta, `$HOME/.local/bin` en este ejemplo, delante de la variable `$PATH` existente.

- c. Vuelva a cargar el perfil en la sesión actual para que esos cambios surtan efecto.

```
$ source ~/.bash_profile
```

3. Para comprobar que está ejecutando la nueva versión, utilice el comando `aws --version`.

```
$ aws --version
aws-cli/1.33.33 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```

## Desinstale la AWS CLI versión 1 usando pip

Si necesita desinstalar el AWS CLI, utilice `pip uninstall`.

```
$ pip3 uninstall awscli
```

## Instalación, actualización o desinstalación con yum

La mayoría de las instancias de Amazon Linux 2 utilizan yum para preinstalar la AWS CLI versión 1.

### Instale o actualice la AWS CLI versión 1 en Amazon Linux mediante yum

Para instalar la última versión de la AWS CLI versión 1 disponible en Amazon Linux, ejecute el siguiente comando.

```
$ sudo yum install awscli
```

Para actualizar a la última versión de la AWS CLI versión 1 disponible en Amazon Linux, ejecute el siguiente comando.

```
$ sudo yum update awscli
```

Para comprobar que está ejecutando la versión más reciente, utilice el comando `aws --version`.

```
$ aws --version
aws-cli/1.33.33 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```

## Desinstale la AWS CLI versión 1 mediante yum

Para desinstalar el AWS CLI, utilice `yum remove`.

```
$ sudo yum remove awscli
```

## Solución de problemas de AWS CLI instalación y desinstalación

Si tiene problemas después de instalar o desinstalar el AWS CLI, consulte los pasos de solución [Errores de solución de problemas](#) de problemas. Para conocer los pasos más relevantes de la solución de problemas, consulte [the section called “Errores de comando no encontrado”](#), [the section called “El comando "aws --version" devuelve una versión diferente a la instalada”](#) y [the section called “El comando "aws --version" devuelve una versión tras desinstalar la AWS CLI”](#).

## Instalación, actualización y desinstalación de la AWS CLI versión 1 en Linux

Puede instalar la versión 1 AWS Command Line Interface (AWS CLI) y sus dependencias en la mayoría de las distribuciones de Linux mediante el administrador de paquetes o el `pip` instalador incluido.

Aunque el `awscli` paquete está disponible en repositorios para otros administradores de paquetes, como `apt` y `yum`, estos no se producen, administran ni admiten. AWS Le recomendamos que lo instale únicamente AWS CLI desde los puntos de AWS distribución oficiales, tal y como se documenta en esta guía.

### Secciones

- [Requisitos previos](#)
- [Instale y desinstale la AWS CLI versión 1 en Linux mediante el instalador incluido](#)
- [Instale y desinstale la AWS CLI versión 1 usando pip](#)
- [Instale y desinstale la AWS CLI versión 1 con Snapcraft](#)
- [Agregue el ejecutable de la AWS CLI versión 1 a la ruta de la línea de comandos](#)



- [Solución de problemas de AWS CLI instalación y desinstalación](#)

## Requisitos previos

Debe tener instalado Python 3.8 o una versión posterior. Para obtener instrucciones de instalación, consulte la página [Downloading Python](#) en la Guía para principiantes de Python.

Matriz de compatibilidad con versiones de Python

AWS CLI versión	Versión de Python compatible
1.32.0: actual	Python 3.8+
1.27.0: 1.31.x	Python 3.7+
1.20.0 – 1.26.x	Python 3.6+
1.19.0 – 1.19.x	Python: Python 3.6 y 2.7
1.17 – 1.18.x	Python 2.7+, Python 3.4+
1.0 – 1.16.x	Python 2.6 y versiones anteriores, Python 3.3 y versiones anteriores

## Instale y desinstale la AWS CLI versión 1 en Linux mediante el instalador incluido

En Linux o macOS, también utilizar el instalador incluido para instalar la versión 1 de la AWS CLI. Este incluye todas las dependencias y se puede usar sin conexión.

### Note

El instalador empaquetado no admite la instalación en rutas que contienen espacios.

## Temas

- [Instale la AWS CLI versión 1 mediante el instalador incluido con sudo](#)
- [Instale la AWS CLI versión 1 utilizando el instalador incluido sin sudo](#)

- [Desinstale el instalador incluido en la AWS CLI versión 1](#)

## Instale la AWS CLI versión 1 mediante el instalador incluido con **sudo**

Los siguientes pasos le permiten instalar la AWS CLI versión 1 desde la línea de comandos en cualquier compilación de Linux o macOS.

A continuación, se muestra un resumen de los comandos de instalación que puede cortar y pegar para que se ejecuten como un único conjunto de comandos.

Para obtener la versión más reciente de AWS CLI, utilice el siguiente bloque de comandos:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle.zip" -o "awscli-bundle.zip"
unzip awscli-bundle.zip
sudo ./awscli-bundle/install -i /usr/local/aws -b /usr/local/bin/aws
```

Para una versión específica de AWS CLI, añada un guión y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión **1.16.312** daría `awscli-bundle-1.16.312.zip` como resultado el siguiente comando:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.312.zip" -o "awscli-bundle.zip"
unzip awscli-bundle.zip
sudo ./awscli-bundle/install -i /usr/local/aws -b /usr/local/bin/aws
```

Siga estos pasos desde la línea de comandos para instalar la AWS CLI versión 1 mediante el instalador incluido.

Para instalar la AWS CLI versión 1 mediante el instalador incluido

1. Descargue el instalador incluido de la AWS CLI versión 1 mediante uno de los métodos siguientes.
  - Descárguelo con el comando `curl`.

Para obtener la versión más reciente de AWS CLI, utilice el siguiente bloque de comandos:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle.zip" -o "awscli-bundle.zip"
```

Para una versión específica de AWS CLI, añada un guión y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión **1.16.312** daría `awscli-bundle-1.16.312.zip` como resultado el siguiente comando:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.312.zip" -o "awscli-bundle.zip"
```

- Descárguelo mediante el enlace directo.

Para obtener la última versión de AWS CLI: <https://s3.amazonaws.com/aws-cli/awscli-bundle.zip>

Para una versión específica de AWS CLI, añada un guión y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión **1.16.312** daría `awscli-bundle-1.16.312.zip` como resultado la siguiente URL <https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.312.zip>

2. Extraiga los archivos del paquete. Si no tiene `unzip` para extraer los archivos, utilice el administrador de paquetes integrado de la distribución de Linux para instalarlo.

```
$ unzip awscli-bundle.zip
```

3. Ejecute el programa de instalación. El instalador instala el AWS CLI at `/usr/local/aws` y crea el enlace simbólico `aws` en el `/usr/local/bin` directorio. Si se usa la opción `-b` para crear un symlink, no es necesario especificar el directorio de instalación en la variable `$PATH` del usuario. Esto debería permitir a todos los usuarios llamar a él AWS CLI ingresando `aws` desde cualquier directorio.

```
$ sudo ./awscli-bundle/install -i /usr/local/aws -b /usr/local/bin/aws
```

El script de instalación se ejecuta en la versión de Python predeterminada del sistema. Si ha instalado una versión alternativa de Python y desea utilizarla para instalarla AWS CLI, ejecute el script de instalación con esa versión siguiendo la ruta absoluta al ejecutable de Python, de la siguiente manera.

```
$ sudo /usr/local/bin/python3.7 awscli-bundle/install -i /usr/local/aws -b /usr/local/bin/aws
```

4. Compruebe que se AWS CLI ha instalado correctamente.

```
$ aws --version
aws-cli/1.33.33 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```

Si obtiene un error, consulte [Solucionar errores AWS CLI](#).

## Instale la AWS CLI versión 1 utilizando el instalador incluido sin **sudo**

Si no tiene sudo permisos o desea instalar AWS CLI solo para el usuario actual, puede usar una versión modificada de los comandos anteriores. Los dos primeros comandos son iguales.

Para obtener la versión más reciente de AWS CLI, utilice el siguiente bloque de comandos:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle.zip" -o "awscli-bundle.zip"
unzip awscli-bundle.zip
./awscli-bundle/install -b ~/bin/aws
```

Para una versión específica de AWS CLI, añada un guión y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión **1.16.312** daría `awscli-bundle-1.16.312.zip` como resultado el siguiente comando:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.312.zip" -o "awscli-bundle.zip"
unzip awscli-bundle.zip
./awscli-bundle/install -b ~/bin/aws
```

Para instalar la AWS CLI versión 1 para el usuario actual

1. Descargue el instalador incluido de la AWS CLI versión 1 de una de las siguientes maneras.
  - Descárguelo con el comando `curl`.

Para obtener la versión más reciente de AWS CLI, utilice el siguiente bloque de comandos:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle.zip" -o "awscli-bundle.zip"
```

Para una versión específica de AWS CLI, añada un guión y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión **1.16.312** daría `awscli-bundle-1.16.312.zip` como resultado el siguiente comando:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.312.zip" -o "awscli-bundle.zip"
```

- Descárguelo mediante el enlace directo.

Para obtener la última versión de AWS CLI: <https://s3.amazonaws.com/aws-cli/awscli-bundle.zip>

Para una versión específica de AWS CLI, añada un guión y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión **1.16.312** daría `awscli-bundle-1.16.312.zip` como resultado la siguiente URL <https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.312.zip>

2. Extraiga los archivos del paquete mediante `unzip`. Si no tiene `unzip`, utilice el administrador de paquetes integrado de la distribución de Linux para instalarlo.

```
$ unzip awscli-bundle.zip
```

3. Ejecute el programa de instalación. El instalador instala el AWS CLI at `/usr/local/aws` y crea el enlace simbólico `aws` en el `/usr/local/bin` directorio. El comando utiliza el parámetro `-b` para especificar el directorio en el que el instalador coloca el archivo del enlace simbólico de `aws`. Debe tener permisos de escritura en la carpeta especificada.

```
$ ./awscli-bundle/install -b ~/bin/aws
```

Esto instala el en la AWS CLI ubicación predeterminada (`~/local/lib/aws`) y crea un enlace simbólico (enlace simbólico) en. `~/bin/aws` Asegúrese de que `~/bin` está en la variable de entorno `PATH` para que el enlace simbólico funcione.

```
$ echo $PATH | grep ~/bin // See if $PATH contains ~/bin (output will be empty if it doesn't)
$ export PATH=~/bin:$PATH // Add ~/bin to $PATH if necessary
```

4. Asegúrese del directorio en el que la AWS CLI versión 1 forma parte de su `PATH` variable.

- a. Busque el script de perfil de su shell en su carpeta de usuario. Si no está seguro de cuál es el shell que tiene, ejecute `echo $SHELL`.

```
$ ls -a ~  
.  ..  .bash_logout  .bash_profile  .bashrc  Desktop  Documents  Downloads
```

- Bash: `.bash_profile`, `.profile` o `.bash_login`.
- Zsh: `.zshrc`
- Tcsh: `.tcshrc`, `.cshrc` o `.login`.

- b. Añada un comando de exportación al final del script de su perfil igual que en el siguiente ejemplo.

```
export PATH=~/.local/bin:$PATH
```

Este comando inserta la ruta, `~/.local/bin` en este ejemplo, delante de la variable `PATH` existente.

- c. Vuelva a cargar el perfil en la sesión actual para que esos cambios surtan efecto.

```
$ source ~/.bash_profile
```

5. Compruebe que se AWS CLI ha instalado correctamente.

```
$ aws --version  
aws-cli/1.33.33 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```


Si obtiene un error, consulte [Solucionar errores AWS CLI](#).

## Desinstale el instalador incluido en la AWS CLI versión 1

1. Si lo instaló AWS CLI mediante el instalador incluido, siga estas instrucciones. El instalador empaquetado no añade recursos fuera del directorio de instalación, salvo el symlink opcional. Por tanto, para desinstalar, solo es necesario eliminar esos dos elementos.

```
$ sudo rm -rf /usr/local/aws  
$ sudo rm -rf /usr/local/bin/aws
```

2. (Opcional) Elimine la información compartida AWS SDK y AWS CLI de configuración de la `.aws` carpeta.

 Warning

Estos ajustes de configuración y credenciales se comparten en todos AWS SDKs los AWS CLI. Si elimina esta carpeta, ninguna de las AWS SDKs que aún estén en su sistema no podrá acceder a ella.

La ubicación predeterminada de la `.aws` carpeta varía según las plataformas; de forma predeterminada, la carpeta se encuentra en `~/.aws/`. Si su usuario tiene permiso de escritura en este directorio, no es necesario que lo utilicesudo.

```
$ sudo rm -r ~/.aws/
```

## Instale y desinstale la AWS CLI versión 1 usando pip

### Temas

- [Instalar pip](#)
- [Instale y actualice la AWS CLI versión 1 usando pip](#)
- [Desinstale el AWS CLI pip que utiliza](#)

### Instalar pip

Si aún no ha instalado pip, puede hacerlo con el script proporcionado por Python Packaging Authority. Ejecute `pip --version` para saber si su versión de Linux ya incluye Python y pip. Si tiene instalada la versión 3 o posterior de Python, recomendamos que utilice el comando `pip3`.

1. Utilice el comando `curl` para descargar el script de instalación. El comando siguiente utiliza el parámetro `-O` (letra "O" mayúscula) para especificar que el archivo descargado debe almacenarse en el directorio actual con el mismo nombre que tiene en el host remoto.

```
$ curl -O https://bootstrap.pypa.io/get-pip.py
```

2. Ejecute el script con el comando `python` o `python3` para descargar e instalar la versión más reciente de `pip` y otros paquetes de soporte necesarios. Cuando se incluye el modificador `--user`, el script instala `pip` en la ruta `~/.local/bin`.

```
$ python3 get-pip.py --user
```

3. Asegúrese de que el directorio que contiene `pip` forma parte de la variable `PATH`.
  - a. Busque el script de perfil de su shell en su carpeta de usuario. Si no está seguro de cuál es el shell que tiene, ejecute `echo $SHELL`.

```
$ ls -a ~  
.  ..  .bash_logout  .bash_profile  .bashrc  Desktop  Documents  Downloads
```

- Bash: `.bash_profile`, `.profile` o `.bash_login`.
  - Zsh: `.zshrc`
  - Tcsh: `.tcshrc`, `.cshrc` o `.login`.
- b. Añada un comando de exportación al final del script de su perfil igual que en el siguiente ejemplo.

```
export PATH=~/.local/bin:$PATH
```

Este comando inserta la ruta, `~/.local/bin` en este ejemplo, delante de la variable `PATH` existente.

- c. Vuelva a cargar el perfil en la sesión actual para que esos cambios surtan efecto.

```
$ source ~/.bash_profile
```

4. Para comprobar que `pip` o `pip3` se ha instalado correctamente, ejecute el siguiente comando.

```
$ pip3 --version  
pip 24.0 from ~/.local/lib/python3.7/site-packages (python 3.7)
```



## Instale y actualice la AWS CLI versión 1 usando pip

1. Utilice el comando `pip` o `pip3` para instalar o actualizar la AWS CLI. Si utiliza la versión 3 o posterior de Python, recomendamos que utilice el comando `pip3`. El `--user` conmutador `pip` instala el AWS CLI to. `~/local/bin`

Para obtener la versión más reciente de AWS CLI, utilice el siguiente bloque de comandos:

```
$ pip3 install awscli --upgrade --user
```

Para una versión específica de AWS CLI, añada dos signos iguales = y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión `1.16.312` sería `==1.16.312` dando como resultado el siguiente comando:

```
$ pip3 install awscli==1.16.312 --upgrade --user
```

### Note

Utilice reglas de entrecomillado adecuadas para el terminal. Para utilizar el carácter =, es posible que tenga que usar comillas simples o dobles para aplicar escape correctamente. En el siguiente ejemplo se aplica escape mediante comillas simples:

```
$ pip3 install 'awscli==1.16.312' --upgrade --user
```

2. Compruebe que se AWS CLI ha instalado correctamente.

```
$ aws --version
aws-cli/1.33.33 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```

Si obtiene un error, consulte [Solucionar errores AWS CLI](#).

## Desinstale el AWS CLI pip que utiliza

1. Si instaló la AWS CLI versión 1 usandopip, también debe desinstalarla usandopip.


```
$ pip uninstall awscli
```

Si utiliza la versión 2 o 3 de Python, es posible que necesite usar el comando `pip2` o `pip3`. Use el `aws --version` comando para determinar la versión de Python asociada a la AWS CLI versión 1 instalada.

```
$ pip3 uninstall awscli
```

Es posible que deba reiniciar la ventana del símbolo del sistema o el equipo para eliminar todos los archivos.

2. (Opcional) Elimine la información compartida AWS SDK y de AWS CLI configuración de la `.aws` carpeta.

 Warning

Estos ajustes de configuración y credenciales se comparten en todos AWS SDKs los AWS CLI. Si elimina esta carpeta, ninguna de las AWS SDKs que aún estén en su sistema no podrá acceder a ella.

La ubicación predeterminada de la `.aws` carpeta varía según las plataformas; de forma predeterminada, la carpeta se encuentra en `~/.aws/`. Si su usuario tiene permiso de escritura en este directorio, no es necesario que lo utilicesudo.

```
$ sudo rm -r ~/.aws/
```

## Instale y desinstale la AWS CLI versión 1 con Snapcraft

### Temas

- [Instala snap](#)
- [Instale y actualice la AWS CLI versión 1 mediante snap](#)
- [Desinstale el AWS CLI archivo mediante snap](#)

## Instala snap

Si aún no lo has snap instalado, puedes hacerlo siguiendo las instrucciones que proporciona Canonical Snapcraft. Ejecuta `snap version` para comprobar si tu versión de Linux ya la incluye.

1. Instala Snapcraft en tu plataforma. Para obtener información sobre la instalación de Snapcraft, consulte [Instalación del daemon en la documentación](#) de Snap.
2. Reinicie el sistema para que PATH las variables se actualicen correctamente. Si tienes problemas de instalación, sigue los pasos que se indican en la sección [Solucionar problemas comunes](#) de la documentación de Snap.
3. Para comprobar que snap está instalado correctamente, ejecuta el siguiente comando.

```
$ snap version
```

## Instale y actualice la AWS CLI versión 1 mediante snap

1. Ejecute el siguiente `snap install` comando para la AWS CLI versión 1.

```
$ snap install aws-cli --channel=v1/stable --classic
```

En función de sus permisos, es posible que necesite añadir `sudo` algo al comando.

```
$ sudo snap install aws-cli --channel=v1/stable --classic
```

2. Compruebe que se AWS CLI ha instalado correctamente.

```
$ aws --version
aws-cli/1.33.33 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```

Si obtiene un error, consulte [Solucionar errores AWS CLI](#).

## Desinstale el AWS CLI archivo mediante snap

1. Si instaló la AWS CLI versión 1 usando `snap`, también debe desinstalarla usando `snap`.

```
$ snap remove aws-cli
```

Es posible que deba reiniciar la ventana del símbolo del sistema o el equipo para eliminar todos los archivos.

2. (Opcional) Elimine la información compartida AWS SDK y de AWS CLI configuración de la `.aws` carpeta.

#### Warning

Estos ajustes de configuración y credenciales se comparten en todos AWS SDKs los AWS CLI. Si elimina esta carpeta, ninguna de las AWS SDKs que aún estén en su sistema no podrá acceder a ella.

La ubicación predeterminada de la `.aws` carpeta varía según las plataformas; de forma predeterminada, la carpeta se encuentra en `~/.aws/`. Si tiene permisos de escritura en este directorio, no necesita usar `sudo`.

```
$ sudo rm -r ~/.aws/
```

## Agregue el ejecutable de la AWS CLI versión 1 a la ruta de la línea de comandos

Tras realizar la instalación con `pip osnap`, puede que tenga que añadir el `aws` ejecutable a la variable de `PATH` entorno del sistema operativo.

Para comprobar en qué carpeta `pip` se instaló AWS CLI el archivo, ejecute el siguiente comando.

```
$ which aws
/home/username/.local/bin/aws
```

Puede hacer referencia a esto como `~/.local/bin/` ya que `/home/username` se corresponde con `~` en Linux.

Si omitió el modificador `--user` y, por tanto, no realizó la instalación en modo de usuario, el archivo ejecutable podría estar en la carpeta `bin` de la instalación de Python. Si no sabe dónde se ha instalado Python, ejecute este comando.

```
$ which python
/usr/local/bin/python
```

La salida puede ser la ruta a un symlink, no el archivo ejecutable real. Ejecute `ls -al` para ver adónde apunta.

```
$ ls -al /usr/local/bin/python
/usr/local/bin/python -> ~/.local/Python/3.6/bin/python3.6
```

`pip` instala los programas en la misma carpeta que contiene la aplicación de Python. Añada esta carpeta a la variable `PATH`.

Para modificar la variable **PATH**

1. Busque el script de perfil de su shell en su directorio de usuario. Si no está seguro de cuál es el shell que tiene, ejecute `echo $SHELL`.

```
$ ls -a ~
.  ..  .bash_logout  .bash_profile  .bashrc  Desktop  Documents  Downloads
```

- Bash: `.bash_profile`, `.profile`, o `.bash_login`
  - Zsh – `.zshrc`
  - Tcsh: `.tcshrc`, `.cshrc`, o `.login`
2. Añada un comando de exportación al script de su perfil.

```
export PATH=~/.local/bin:$PATH
```

Este comando añade una ruta, `~/.local/bin` en este ejemplo, a la variable `PATH` actual.

3. Cargue el perfil actualizado en su sesión actual.

```
$ source ~/.bash_profile
```

## Solución de problemas de AWS CLI instalación y desinstalación

Si tiene problemas después de instalar o desinstalar el AWS CLI, consulte los pasos de solución [Errores de solución de problemas](#) de problemas. Para conocer los pasos más relevantes de la solución de problemas, consulte [the section called “Errores de comando no encontrado”](#), [the section called “El comando "aws --version" devuelve una versión diferente a la instalada”](#) y [the section called “El comando "aws --version" devuelve una versión tras desinstalar la AWS CLI”](#).

## Instalar, actualizar y desinstalar la AWS CLI versión 1 en macOS

Puedes instalar la versión 1 AWS Command Line Interface (AWS CLI) y sus dependencias en macOS mediante el instalador incluido o. pip

### Secciones

- [Requisitos previos](#)
- [Instale, actualice y desinstale la AWS CLI versión 1 en macOS mediante el instalador incluido](#)
- [Instale, actualice y desinstale la AWS CLI versión 1 usando pip](#)
- [Solución de problemas de AWS CLI instalación y desinstalación](#)

## Requisitos previos

Antes de poder instalar la AWS CLI versión 1 en macOS, asegúrate de tener instalado Python 3.8 o posterior. Para obtener instrucciones de instalación, consulte la página [Downloading Python](#) en la Guía para principiantes de Python.

### Matriz de compatibilidad con versiones de Python

AWS CLI versión	Versión de Python compatible
1.32.0: actual	Python 3.8+
1.27.0: 1.31.x	Python 3.7+
1.20.0 – 1.26.x	Python 3.6+
1.19.0 – 1.19.x	Python: Python 3.6 y 2.7
1.17 – 1.18.x	Python 2.7+, Python 3.4+

AWS CLI versión	Versión de Python compatible
1.0 – 1.16.x	Python 2.6 y versiones anteriores, Python 3.3 y versiones anteriores

## Instale, actualice y desinstale la AWS CLI versión 1 en macOS mediante el instalador incluido

En Linux o macOS, puede usar el instalador empaquetado para instalar la versión 1 de la AWS Command Line Interface (AWS CLI). Este incluye todas las dependencias y se puede usar sin conexión.

El instalador empaquetado no admite la instalación en rutas que contienen espacios.

### Temas

- [Instale la AWS CLI versión 1 mediante el instalador incluido con sudo](#)
- [Instale la AWS CLI versión 1 utilizando el instalador incluido sin sudo](#)
- [Desinstale el instalador incluido en la AWS CLI versión 1](#)

## Instale la AWS CLI versión 1 mediante el instalador incluido con **sudo**

Los siguientes pasos te permiten instalar la AWS CLI versión 1 desde la línea de comandos en cualquier compilación de macOS.

A continuación, se muestra un resumen de los comandos de instalación que puede cortar y pegar para que se ejecuten como un único conjunto de comandos.

Para obtener la versión más reciente de AWS CLI, usa el siguiente bloque de comandos:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle.zip" -o "awscli-bundle.zip"
unzip awscli-bundle.zip
sudo ./awscli-bundle/install -i /usr/local/aws -b /usr/local/bin/aws
```

Para una versión específica de AWS CLI, añada un guión y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión **1.16.312** daría `awscli-bundle-1.16.312.zip` como resultado el siguiente comando:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.312.zip" -o "awscli-bundle.zip"
unzip awscli-bundle.zip
sudo ./awscli-bundle/install -i /usr/local/aws -b /usr/local/bin/aws
```

Para instalar la AWS CLI versión 1 mediante el instalador incluido

1. Descargue el instalador incluido de la AWS CLI versión 1 de una de las siguientes maneras:
  - Descárguelo con el comando `curl`.

Para instalar la última versión del AWS CLI, utilice el siguiente bloque de comandos:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle.zip" -o "awscli-bundle.zip"
```

Para una versión específica de la AWS CLI, agregue un guión y el número de versión al nombre del archivo. Para este ejemplo, el nombre de archivo de la versión `1.16.312` daría `awscli-bundle-1.16.312.zip` como resultado el siguiente comando:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.312.zip" -o "awscli-bundle.zip"
```

- Descárguelo mediante el enlace directo.

Para obtener la última versión de AWS CLI: <https://s3.amazonaws.com/aws-cli/awscli-bundle.zip>

Para una versión específica de AWS CLI, añada un guión y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión `1.16.312` daría `awscli-bundle-1.16.312.zip` como resultado la siguiente URL <https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.312.zip>

2. Extraiga (descomprima) los archivos del paquete. Si no lo tienes `unzip`, usa el administrador de paquetes integrado en tu macOS distribución para instalarla.

```
$ unzip awscli-bundle.zip
```

3. Ejecute el programa de instalación. El instalador instala el AWS CLI at `/usr/local/aws` y crea el enlace simbólico `aws` en la `/usr/local/bin` carpeta. Si se usa la opción `-b` para crear un



enlace simbólico, no es necesario especificar la carpeta de instalación en la variable \$PATH del usuario. Esto debería permitir a todos los usuarios llamar a él AWS CLI ingresando `aws` desde cualquier directorio.

```
$ sudo ./awscli-bundle/install -i /usr/local/aws -b /usr/local/bin/aws
```

El script de instalación se ejecuta en la versión de Python predeterminada del sistema. Si ha instalado una versión alternativa de Python y desea utilizarla para instalarla AWS CLI, ejecute el script de instalación con esa versión siguiendo la ruta absoluta al ejecutable de Python, de la siguiente manera.

```
$ sudo /usr/local/bin/python3.7 awscli-bundle/install -i /usr/local/aws -b /usr/local/bin/aws
```

4. Compruebe que se AWS CLI ha instalado correctamente.

```
$ aws --version
aws-cli/1.33.33 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```

Si obtiene un error, consulte [Solucionar errores AWS CLI](#).

## Instale la AWS CLI versión 1 utilizando el instalador incluido sin **sudo**

Si no tiene `sudo` permisos o desea instalar AWS CLI solo para el usuario actual, puede usar una versión modificada de los comandos anteriores. Los dos primeros comandos son iguales.

Para obtener la versión más reciente de AWS CLI, utilice el siguiente bloque de comandos:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle.zip" -o "awscli-bundle.zip"
unzip awscli-bundle.zip
./awscli-bundle/install -b ~/bin/aws
```

Para una versión específica de AWS CLI, añada un guión y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión **1.16.312** daría `awscli-bundle-1.16.312.zip` como resultado el siguiente comando:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.312.zip" -o "awscli-bundle.zip"
unzip awscli-bundle.zip
```

```
./awscli-bundle/install -b ~/bin/aws
```

Para instalar la AWS CLI versión 1 para el usuario actual

1. Descargue el instalador incluido de la AWS CLI versión 1 mediante uno de los siguientes métodos:
  - Descárguelo con el comando `curl`.

Para instalar la última versión del AWS CLI, utilice el siguiente bloque de comandos:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle.zip" -o "awscli-bundle.zip"
```

Para una versión específica de la AWS CLI, agregue un guión y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión **1.16.312** daría `awscli-bundle-1.16.312.zip` como resultado el siguiente comando:

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.312.zip" -o "awscli-bundle.zip"
```

- Descárguelo mediante el enlace directo.

Para obtener la última versión de AWS CLI: <https://s3.amazonaws.com/aws-cli/awscli-bundle.zip>

Para una versión específica de AWS CLI, añada un guión y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión **1.16.312** daría `awscli-bundle-1.16.312.zip` como resultado la siguiente URL <https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.312.zip>

2. Extraiga los archivos del paquete. Si no tiene `unzip`, utilice el administrador de paquetes integrado de la distribución de Linux para instalarlo.

```
$ unzip awscli-bundle.zip
```

3. Ejecute el programa de instalación. El instalador instala el AWS CLI at `/usr/local/aws` y crea el enlace simbólico `aws` en el `/usr/local/bin` directorio. El comando utiliza el parámetro `-b` para especificar el directorio en el que el instalador coloca el archivo del enlace simbólico de `aws`. Debe tener permisos de escritura en el directorio especificado.

```
$ ./awscli-bundle/install -b ~/bin/aws
```

Esto instala el en la AWS CLI ubicación predeterminada (`~/local/lib/aws`) y crea un enlace simbólico (enlace simbólico) en `~/bin/aws`. Asegúrese de que `~/bin` está en la variable de entorno `$PATH` para que el enlace simbólico funcione.

```
$ echo $PATH | grep ~/bin // See if $PATH contains ~/bin (output will be empty
if it doesn't)
$ export PATH=~/bin:$PATH // Add ~/bin to $PATH if necessary
```

4. Asegúrese de que la carpeta en la que está instalada la AWS CLI versión 1 forme parte de la variable. `$PATH`
  - a. Busque el script de perfil de su shell en su carpeta de usuario. Si no está seguro de cuál es el shell que tiene, ejecute `echo $SHELL`.

```
$ ls -a ~
.  ..  .bash_logout  .bash_profile  .bashrc  Desktop  Documents  Downloads
```

- Bash: `.bash_profile`, `.profile` o `.bash_login`.
  - Zsh: `.zshrc`
  - Tcsh: `.tcshrc`, `.cshrc` o `.login`.
- b. Añada un comando de exportación al final del script de su perfil igual que en el siguiente ejemplo.

```
export PATH=~/local/bin:$PATH
```

Este comando inserta la ruta, `~/local/bin` en este ejemplo, delante de la variable `PATH` existente.

- c. Vuelva a cargar el perfil en la sesión actual para que esos cambios surtan efecto.

```
$ source ~/.bash_profile
```

5. Compruebe que se AWS CLI ha instalado correctamente.

```
$ aws --version
```

```
aws-cli/1.33.33 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```

Si obtiene un error, consulte [Solucionar errores AWS CLI](#).

## Desinstale el instalador incluido en la AWS CLI versión 1

1. El instalador incluido coloca todo en el directorio de instalación excepto el enlace simbólico opcional, por lo que, para efectuar la desinstalación, solo necesita eliminar esos dos elementos.

```
$ sudo rm -rf /usr/local/aws
$ sudo rm /usr/local/bin/aws
```

2. (Opcional) Elimine la información compartida AWS SDK y AWS CLI de configuración de la `.aws` carpeta.

### Warning

Estos ajustes de configuración y credenciales se comparten entre todos AWS SDKs y los AWS CLI. Si elimina esta carpeta, ninguna de las AWS SDKs que aún estén en su sistema no podrá acceder a ella.

La ubicación predeterminada de la `.aws` carpeta varía según las plataformas; de forma predeterminada, la carpeta se encuentra en `~/.aws/`. Si su usuario tiene permiso de escritura en este directorio, no es necesario que lo utilicesudo.

```
$ sudo rm ~/.aws/
```

## Instale, actualice y desinstale la AWS CLI versión 1 usando pip

Puede utilizar pip directamente para instalar la AWS CLI.

### Temas

- [Instalar pip](#)
- [Instale y actualice el pip AWS CLI mediante](#)
- [Agrega el ejecutable de la AWS CLI versión 1 a la ruta de línea de comandos de macOS](#)

- [Desinstale el AWS CLI pip mediante](#)

## Instalar pip

Si aún no ha instalado pip, puede hacerlo con el script proporcionado por Python Packaging Authority. Ejecute `pip --version` para saber si su versión de Linux ya incluye Python y pip. Si tiene instalada la versión 3 o posterior de Python, recomendamos que utilice el comando `pip3`.

1. Utilice el comando `curl` para descargar el script de instalación. El comando siguiente utiliza el parámetro `-O` (letra «O» mayúscula) para especificar que el archivo descargado debe almacenarse en la carpeta actual con el mismo nombre que tiene en el host remoto.

```
$ curl -O https://bootstrap.pypa.io/get-pip.py
```

2. Ejecute el script con el comando `python` o `python3` para descargar e instalar la versión más reciente de pip y otros paquetes de soporte necesarios. Cuando se incluye el modificador `--user`, el script instala pip en la ruta `~/.local/bin`.

```
$ python3 get-pip.py --user
```

## Instale y actualice el pip AWS CLI mediante

1. Utilice el comando `pip` o `pip3` para instalar la AWS CLI. Si utiliza la versión 3 o posterior de Python, recomendamos que utilice el comando `pip3`.

Para obtener la última versión de AWS CLI, utilice el siguiente bloque de comandos:

```
$ pip3 install awscli --upgrade --user
```

Para una versión específica de AWS CLI, añada dos signos iguales `=` y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión **1.16.312** sería **`==1.16.312`** dando como resultado el siguiente comando:

```
$ pip3 install awscli==1.16.312 --upgrade --user
```

**Note**

Utilice reglas de entrecomillado adecuadas para el terminal. Para utilizar el carácter =, es posible que tenga que usar comillas simples o dobles para aplicar escape correctamente. En el siguiente ejemplo se aplica escape mediante comillas simples:

```
$ pip3 install 'awscli==1.16.312' --upgrade --user
```

2. Compruebe que AWS CLI está instalado correctamente.

```
$ aws --version  
aws-cli/1.33.33 Python/3.11.6 Darwin/23.3.0 botocore/1.18.6
```

Si no se encuentra el programa, [añádalo a la ruta de la línea de comandos](#).

## Agrega el ejecutable de la AWS CLI versión 1 a la ruta de línea de comandos de macOS

Después de realizar la instalación con `pip`, es posible que necesite añadir el programa `aws` a la variable de entorno `PATH` del sistema operativo. La ubicación del programa depende de dónde se haya instalado Python.

Example AWS CLI ubicación de instalación: macOS con Python 3.6 y `pip` (modo usuario)

```
~/Library/Python/3.7/bin
```

Sustituya la versión de Python que tiene con la versión del ejemplo anterior.

Si no sabe dónde se ha instalado Python, ejecute `which python`.

```
$ which python  
/usr/local/bin/python
```

La salida puede ser la ruta a un symlink, no el programa real. Ejecute `ls -al` para ver adónde apunta.

```
$ ls -al /usr/local/bin/python
```

```
~/Library/Python/3.7/bin/python3.7
```

pip instala los programas en la misma carpeta que contiene la aplicación de Python. Añada esta carpeta a la variable PATH.

Para modificar la variable **PATH**

1. Busque el script de perfil de su shell en su directorio de usuario. Si no está seguro de cuál es el shell que tiene, ejecute `echo $SHELL`.

```
$ ls -a ~  
.  ..  .bash_logout  .bash_profile  .bashrc  Desktop  Documents  Downloads
```

- Bash: `.bash_profile`, `.profile`, o `.bash_login`
  - Zsh – `.zshrc`
  - Tcsh: `.tcshrc`, `.cshrc`, o `.login`
2. Añada un comando de exportación al script de su perfil.

```
export PATH=~/.local/bin:$PATH
```

Este comando añade una ruta, `~/.local/bin` en este ejemplo, a la variable PATH actual.

3. Cargue el perfil actualizado en su sesión actual.

```
$ source ~/.bash_profile
```

## Desinstale el AWS CLI pip mediante

1. Si instaló la AWS CLI versión 1 usando pip, también debe desinstalarla usando pip.

```
$ pip uninstall awscli
```

Si utiliza la versión 2 o 3 de Python, es posible que necesite usar el comando `pip2` o `pip3`. Use el `aws --version` comando para determinar la versión de Python asociada a la AWS CLI versión 1 instalada.

```
$ pip3 uninstall awscli
```

Es posible que deba reiniciar la ventana del símbolo del sistema o el equipo para eliminar todos los archivos.

2. (Opcional) Elimine la información compartida AWS SDK y de AWS CLI configuración de la `.aws` carpeta.

#### Warning

Estos ajustes de configuración y credenciales se comparten entre todos AWS SDKs y los AWS CLI. Si elimina esta carpeta, ninguna de las AWS SDKs que aún estén en su sistema no podrá acceder a ella.

La ubicación predeterminada de la `.aws` carpeta varía según las plataformas; de forma predeterminada, la carpeta se encuentra en `~/.aws/`. Si su usuario tiene permiso de escritura en este directorio, no es necesario que lo utilicesudo.

```
$ sudo rm ~/.aws/
```

## Solución de problemas de AWS CLI instalación y desinstalación

Si tiene problemas después de instalar o desinstalar el AWS CLI, consulte los pasos de solución [Errores de solución de problemas](#) de problemas. Para conocer los pasos más relevantes de la solución de problemas, consulte [the section called “Errores de comando no encontrado”](#), [the section called “El comando “aws --version” devuelve una versión diferente a la instalada”](#) y [the section called “El comando “aws --version” devuelve una versión tras desinstalar la AWS CLI”](#).

## Instalar, actualizar y desinstalar la AWS CLI versión 1 en Windows

Puede instalar la versión 1 de AWS Command Line Interface (AWS CLI) en Windows mediante un instalador independiente (recomendado) o pip un administrador de paquetes para Python.

No incluya el símbolo de comando (`C:\>`) al escribir un comando. Estos símbolos se incluyen en los listados del programa para diferenciar los comandos introducidos de los que devuelve la AWS CLI. En el resto de esta guía se utiliza el símbolo del sistema genérico (\$) excepto en los casos en los que un comando es específico de Windows.



## Temas

- [Instale, actualice y desinstale la AWS CLI versión 1 mediante el MSI instalador](#)
- [Instale, actualice y desinstale la AWS CLI versión 1 con Python y pip en Windows](#)
- [Añada el ejecutable de la AWS CLI versión 1 a la ruta de la línea de comandos](#)
- [Solución de problemas de AWS CLI instalación y desinstalación](#)

## Instale, actualice y desinstale la AWS CLI versión 1 mediante el MSI instalador

La AWS CLI versión 1 es compatible con Windows XP o versiones posteriores. Para los usuarios de Windows, el paquete de MSI instalación ofrece una forma cómoda y familiar de instalar la AWS CLI versión 1 sin necesidad de instalar ningún otro requisito previo.

### Instale y actualice la AWS CLI versión 1 mediante el MSI instalador

Consulte la página de [versiones](#) GitHub para ver cuándo se lanzó la última versión. Cuando se hayan publicado las actualizaciones, deberá repetir el proceso de instalación para obtener la última versión de la versión 1 de la AWS CLI .

1. Descarga el MSI instalador correspondiente:
  - AWS CLI MSIinstalador para Windows (64 bits): <https://s3.amazonaws.com/aws-cli/AWSCLI64PY3.msi>
  - AWS CLI MSIinstalador para Windows (32 bits): [.msi https://s3.amazonaws.com/aws-cli/AWSCLI32PY3](https://s3.amazonaws.com/aws-cli/AWSCLI32PY3)
  - AWS CLI archivo de configuración combinado para Windows: <https://s3.amazonaws.com/aws-cli/AWSCLISetup.exe> (incluye los MSI instaladores de 32 y 64 bits e instala automáticamente la versión correcta)
2. Ejecute el MSI instalador descargado o el archivo de configuración.
3. Siga las instrucciones en pantalla. De forma predeterminada, la AWS CLI versión 1 se instala en C:\Program Files\Amazon\AWSCLI (versión de 64 bits) o C:\Program Files (x86)\Amazon\AWSCLI (versión de 32 bits).
4. Para confirmar la instalación, utilice el comando `aws --version` en un símbolo del sistema (abra el menú Inicio y busque cmd para iniciar un símbolo del sistema).

```
C:\> aws --version
```

```
aws-cli/1.33.33 Python/3.11.6 Windows/10 botocore/1.18.6
```

Si Windows no encuentra el programa, puede que tenga que cerrar y volver a abrir la línea de comandos para actualizar la ruta o [añadir el directorio de instalación a la variable de PATH entorno manualmente](#).

## Desinstale la AWS CLI versión 1.

Para utilizar las siguientes instrucciones de desinstalación, debe haber instalado la AWS CLI versión 1 con el archivo de MSI instalación o configuración.

1. Abra Programas y características mediante una de las siguientes acciones:

- Abra el Panel de control y elija Programas y características.
- Abra un símbolo del sistema y escriba el comando siguiente.

```
C:\> appwiz.cpl
```

2. Seleccione la entrada denominada AWS Command Line Interface y elija Desinstalar para iniciar el desinstalador.
3. Confirme que desea desinstalar el AWS CLI.
4. (Opcional) Elimine la información compartida AWS SDK y de AWS CLI configuración de la `.aws` carpeta.

### Warning

Estos ajustes de configuración y credenciales se comparten en todos AWS SDKs los AWS CLI. Si elimina esta carpeta, ninguna de las AWS SDKs que aún estén en su sistema no podrá acceder a ella.

La ubicación predeterminada de la `.aws` carpeta varía según las plataformas; de forma predeterminada, la carpeta se encuentra en `%UserProfile%\.aws`.

```
$ rmdir %UserProfile%\.aws
```

# Instale, actualice y desinstale la AWS CLI versión 1 con Python y pip en Windows

La Python Software Foundation ofrece instaladores para Windows que incluyen pip.

## Requisitos previos

Debe tener instalado Python 3.8 o una versión posterior. Para obtener instrucciones de instalación, consulte la página [Downloading Python](#) en la Guía para principiantes de Python.

### Warning

Python 2.7 quedó obsoleta debido a la [Python Software Foundation](#) desde el 1 de enero de 2020. A partir de AWS CLI la versión 1.20.0, se requiere una versión mínima de Python 3.6. Para usar la AWS CLI versión 1 con una versión anterior de Python, debe instalar una versión anterior de la AWS CLI versión 1. Para ver la matriz de compatibilidad de la AWS CLI versión 1 de Python, consulte [the section called “Requisitos de versión de Python”](#).

## Instale y actualice la AWS CLI versión 1 mediante pip

1. Para instalar la AWS CLI versión 1, utilice el pip3 comando (si utiliza Python versión 3 o posterior) o el pip comando.

Para obtener la última versión de AWS CLI, utilice el siguiente bloque de comandos:

```
C:\> pip3 install awscli --upgrade --user
```

Para una versión específica de AWS CLI, añada un símbolo inferior a < y el número de versión al nombre del archivo. Para este ejemplo, el nombre del archivo de la versión **1.16.312** sería **<1.16.312** dando como resultado el siguiente comando:

```
C:\> pip3 install awscli<1.16.312 --upgrade --user
```

2. Compruebe que la AWS CLI versión 1 esté instalada correctamente. Si no hay respuesta, consulte la sección [Añada el ejecutable de la AWS CLI versión 1 a la ruta de la línea de comandos](#).

```
C:\> aws --version
```

```
aws-cli/1.33.33 Python/3.11.6 Windows/10 botocore/1.18.6
```

## Desinstale la AWS CLI versión 1 con pip

1. Si instaló la AWS CLI versión 1 usandopip, también debe desinstalarla usandopip.

```
C:\> pip uninstall awscli
```

Si utiliza la versión 2 o 3 de Python, es posible que necesite usar el comando `pip2` o `pip3`. Utilice el `aws --version` comando para determinar la versión de Python asociada a la AWS CLI versión 1 instalada.

```
C:\> pip3 uninstall awscli
```

Es posible que deba reiniciar la ventana del símbolo del sistema o el equipo para eliminar todos los archivos.

2. (Opcional) Elimine la información compartida AWS SDK y de AWS CLI configuración de la `.aws` carpeta.

### Warning

Estos ajustes de configuración y credenciales se comparten en todos AWS SDKs los AWS CLI. Si elimina esta carpeta, ninguna de las AWS SDKs que aún estén en su sistema no podrá acceder a ella.

La ubicación predeterminada de la `.aws` carpeta varía según las plataformas; de forma predeterminada, la carpeta se encuentra en `%UserProfile%\.aws`.

```
$ rmdir %UserProfile%\.aws
```

## Añada el ejecutable de la AWS CLI versión 1 a la ruta de la línea de comandos

Tras instalar la AWS CLI versión 1 con `pip`, añade el `aws` programa a la variable de `PATH` entorno del sistema operativo. Con una MSI instalación, esto debería ocurrir automáticamente. Pero si el comando `aws` no se ejecuta después de instalarlo, es posible que deba configurarlo manualmente.

1. Utilice el comando `where` para buscar la ubicación del archivo `aws`. De forma predeterminada, el comando `where` muestra dónde se encuentra un programa especificado en la variable `PATH` del sistema.

```
C:\> where aws
```

Las rutas que aparecen dependen de la plataforma y del método utilizado para instalar la AWS CLI. Los nombres de carpeta que incluyen los números de versión puede variar. Estos ejemplos reflejan el uso de Python versión 3.7. Reemplace la versión por el número de versión que está utilizando, según sea necesario. Las rutas típicas son las siguientes:

- Python 3 y **pip3**: `C:\Program Files\Python37\Scripts\`
- Python 3 y **pip3**: opción de usuario en versiones anteriores de Windows `%USERPROFILE%\AppData\Local\Programs\Python\Python37\Scripts`
- Python 3 y **pip3**, opción de usuario en Windows 10: `%USERPROFILE%\AppData\Roaming\Python\Python37\Scripts`
- MSI instalador (64 bits) — `C:\Program Files\Amazon\AWSCLI\bin`
- MSI instalador (32 bits) — `C:\Program Files (x86)\Amazon\AWSCLI\bin`

Siga estos pasos en función de si se devuelve una ruta de archivo.

A file path is returned

```
C:\> where aws
C:\Program Files\Amazon\AWSCLI\bin\aws.exe
```

Para encontrar dónde está instalado el programa de `aws`, ejecute el siguiente comando.

```
C:\> where c:\ aws
```

```
C:\Program Files\Python37\Scripts\aws
```

## A file path is NOT returned

Si el comando `where` devuelve el siguiente error, no está en la variable PATH del sistema y no puede ejecutarse simplemente escribiendo su nombre.

```
C:\> where c:\ aws
INFO: Could not find files for the given pattern(s).
```

En ese caso, ejecute el comando `where` con el parámetro `/R` *path* para indicarle que busque en todas las carpetas y agregue después la ruta manualmente. Utilice la línea de comandos o el Explorador de archivos para averiguar en qué lugar del equipo está instalado.

```
C:\> where /R c:\ aws
c:\Program Files\Amazon\AWSCLI\bin\aws.exe
c:\Program Files\Amazon\AWSCLI\bincompat\aws.cmd
c:\Program Files\Amazon\AWSCLI\runtime\Scripts\aws
c:\Program Files\Amazon\AWSCLI\runtime\Scripts\aws.cmd
...
```

2. Pulse la tecla de Windows y escriba **environment variables**.
3. Elija Edit environment variables for your account (Editar las variables de entorno de esta cuenta).
4. Elija y PATH, a continuación, elija Editar.
5. Agregue la ruta que encontró en el campo Valor de variable por ejemplo, ***C:\Program Files\Amazon\AWSCLI\bin\aws.exe***.
6. Elija OK (Aceptar) dos veces para aplicar la nueva configuración.
7. Cierre los símbolos del sistema en ejecución y vuelva a abrir la ventana de símbolo del sistema.

## Solución de problemas de AWS CLI instalación y desinstalación

Si tiene problemas después de instalar o desinstalar el AWS CLI, consulte los pasos de solución [Errores de solución de problemas](#) de problemas. Para conocer los pasos más relevantes de la solución de problemas, consulte [the section called “Errores de comando no encontrado”](#), [the section called “El comando "aws --version" devuelve una versión diferente a la instalada”](#) y [the section called “El comando "aws --version" devuelve una versión tras desinstalar la AWS CLI”](#).

# Instalar y actualizar la AWS CLI versión 1 en un entorno virtual

Puede evitar conflictos entre las versiones obligatorias y otros `pip` paquetes instalando la versión 1 de AWS Command Line Interface (AWS CLI) en un entorno virtual.

## Temas

- [Requisitos previos](#)
- [Instale y actualice la AWS CLI versión 1 en un entorno virtual](#)
- [Solución de problemas de AWS CLI instalación y desinstalación](#)

## Requisitos previos

- Python 3.8 o versiones posteriores. Para obtener instrucciones de instalación, consulte la página [Downloading Python](#) en la Guía para principiantes de Python.

### Matriz de compatibilidad con versiones de Python

AWS CLI versión	Versión de Python compatible
1.32.0: actual	Python 3.8+
1.27.0: 1.31.x	Python 3.7+
1.20.0 – 1.26.x	Python 3.6+
1.19.0 – 1.19.x	Python: Python 3.6 y 2.7
1.17 – 1.18.x	Python 2.7+, Python 3.4+
1.0 – 1.16.x	Python 2.6 y versiones anteriores, Python 3.3 y versiones anteriores

- Se ha instalado `pip` o `pip3`.

## Instale y actualice la AWS CLI versión 1 en un entorno virtual

1. Instale `virtualenv` mediante `pip`.

```
$ pip install --user virtualenv
```

2. Cree un entorno virtual y asígnele un nombre.

```
$ virtualenv ~/cli-ve
```

También puede utilizar la opción `-p` para especificar una versión de Python distinta de la predeterminada.

```
$ virtualenv -p /usr/bin/python37 ~/cli-ve
```

3. Active el entorno virtual nuevo.

Linux o macOS

```
$ source ~/cli-ve/bin/activate
```

Windows

```
$ %USERPROFILE%\cli-ve\Scripts\activate
```

El símbolo del sistema cambia para mostrar que el entorno virtual está activo.

```
(cli-ve)~$
```

4. Instale o actualice la AWS CLI versión 1 en su entorno virtual.

```
(cli-ve)~$ pip install --upgrade awscli
```

5. Compruebe que la AWS CLI versión 1 esté instalada correctamente.

```
$ aws --version  
aws-cli/1.33.33 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```

6. Puede utilizar el comando `deactivate` para salir del entorno virtual. Siempre que inicie una sesión nueva, debe activar el entorno de nuevo.



## Solución de problemas de AWS CLI instalación y desinstalación

Si tiene problemas después de instalar o desinstalar el AWS CLI, consulte los pasos de solución [Errores de solución de problemas](#) de problemas. Para conocer los pasos más relevantes de la solución de problemas, consulte [the section called “Errores de comando no encontrado”](#), [the section called “El comando "aws --version" devuelve una versión diferente a la instalada”](#) y [the section called “El comando "aws --version" devuelve una versión tras desinstalar la AWS CLI”](#).

# Configure el AWS CLI

En esta sección se explica cómo configurar los ajustes con los que AWS Command Line Interface (AWS CLI) interactúa AWS. Estos incluyen los siguientes:

- Las credenciales identifican quién llama a la API. Las credenciales de acceso se utilizan para cifrar la solicitud a los AWS servidores para confirmar su identidad y recuperar las políticas de permisos asociadas. Estos permisos determinan las acciones que puede realizar. Para obtener información sobre cómo configurar las credenciales, consulte [Credenciales de autenticación y acceso](#).
- Otros detalles de configuración para indicar AWS CLI cómo procesar las solicitudes, como el formato de salida predeterminado y la AWS región predeterminada.

## Note

AWS requiere que todas las solicitudes entrantes estén firmadas criptográficamente. Lo AWS CLI hace por ti. La "firma" incluye una marca temporal de fecha/hora. Por lo tanto, debe asegurarse de que la fecha y la hora de su equipo se haya establecido correctamente. Si no lo haces y la fecha y hora de la firma están muy alejadas de la fecha y hora reconocidas por el AWS servicio, AWS rechazará la solicitud.

## Prioridad de configuración y credenciales

Las credenciales y los valores de configuración se encuentran en varios lugares, como las variables de entorno del sistema o del usuario o los archivos de AWS configuración locales, o se declaran explícitamente en la línea de comandos como parámetros. Ciertas ubicaciones tienen prioridad sobre otras. Las AWS CLI credenciales y los ajustes de configuración tienen prioridad en el siguiente orden:

1. [Opciones de línea de comandos](#): anula la configuración en cualquier otra ubicación, como los parámetros `--region`, `--output` y `--profile`.
2. [Variables de entorno](#): puede almacenar valores en las variables de entorno de su sistema.
3. [Asumir rol](#): asuma los permisos de un rol de IAM mediante la configuración o el comando [aws sts assume-role](#).

4. [Asumir rol con identidad web](#): asuma los permisos de un rol de IAM con identidad web mediante la configuración o el comando `aws sts assume-role`.
5. [Archivo de credenciales](#): los archivos `credentials` y `config` se actualizan al ejecutar el comando `aws configure`. El archivo `credentials` se encuentra en `~/.aws/credentials` en Linux o macOS, o en `C:\Users\USERNAME\.aws\credentials` en Windows.
6. [Proceso personalizado](#): obtenga sus credenciales de un origen externo.
7. [Archivo de configuración](#): los archivos `credentials` y `config` se actualizan al ejecutar el comando `aws configure`. El archivo `config` se encuentra en `~/.aws/config` en Linux o macOS, o en `C:\Users\USERNAME\.aws\config` en Windows.
8. [Credenciales contenedor](#): puede asociar un rol de IAM con cada una de las definiciones de tareas de Amazon Elastic Container Service (Amazon ECS). Las credenciales temporales de ese rol estarán disponibles para los contenedores de esa tarea. Para obtener más información, consulte [Roles de IAM para tareas](#) en la Guía para desarrolladores de Amazon Elastic Container Service.
9. [Credenciales de perfil de instancia de Amazon EC2](#): puede asociar un rol de IAM con cada una de las instancias de Amazon Elastic Compute Cloud (Amazon EC2). Las credenciales temporales de ese rol estarán disponibles para el código que se ejecute en la instancia. Las credenciales se entregan a través del servicio de metadatos de Amazon EC2. Para obtener más información, consulte [Funciones de IAM para Amazon EC2](#) en la Guía del usuario de Amazon EC2 y [Uso de perfiles de instancia](#) en la Guía del usuario de IAM.

## Temas adicionales en esta sección

- [the section called “Opciones de los archivos de configuración y credenciales”](#)
- [the section called “Variables de entorno”](#)
- [the section called “Opciones de línea de comandos”](#)
- [the section called “Finalización de comandos”](#)
- [the section called “Reintentos”](#)
- [the section called “Usa un HTTP proxy”](#)

## Opciones de los archivos de configuración y credenciales

Puede guardar las opciones de configuración y las credenciales que utiliza con frecuencia en archivos que son mantenidos por la AWS CLI.

Los archivos se dividen en `profiles`. De forma predeterminada, AWS CLI utiliza la configuración que se encuentra en el perfil nombrado `default`. Para utilizar una configuración diferente, puede crear y hacer referencia a perfiles adicionales.

También es posible anular una opción individual estableciendo una de las variables de entorno compatibles o utilizando un parámetro de la línea de comandos. Para obtener más información sobre la prioridad de opciones de configuración, consulte [Configure el AWS CLI](#).

#### Note

Para obtener información sobre cómo configurar las credenciales, consulte [Credenciales de autenticación y acceso](#).

## Temas

- [Formato de los archivos de configuración y credenciales](#)
- [¿Dónde se almacenan las opciones de configuración?](#)
- [Usar perfiles con nombre](#)
- [Definir y ver las opciones de configuración mediante comandos](#)
- [Establecimiento de una nueva configuración y ejemplos de comandos de credenciales](#)
- [Configuraciones del archivo `config` admitidas](#)

## Formato de los archivos de configuración y credenciales

Los archivos `config` y `credentials` están organizados en secciones. Las secciones incluyen perfiles y servicios. Una sección es una colección con nombre de configuraciones y continúa hasta que se encuentra otra línea de definición de sección. Se pueden almacenar varios perfiles y secciones en los archivos `config` y `credentials`.

Son archivos de texto sin formato que utilizan el formato siguiente:

- Los nombres de las secciones aparecen entre corchetes `[]` como `[default]`, `[profile user1]` y `[sso-session]`.
- Todas las entradas de una sección adoptan el formato general de `setting_name=value`.
- Las líneas se pueden comentar si se inician con un carácter de almohadilla (`#`).



```
aws_access_key_id=ASIAI44QH8DHBEXAMPLE
aws_secret_access_key=je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY
aws_session_token
= fcZib3JpZ2luX2IqoJb3JpZ2luX2IqoJb3JpZ2luX2IqoJb3JpZ2luX2IqoJb3JpZVERYLONGSTRINGEXAMPLE
```

### Archivo de configuración

```
[default]
region=us-west-2
output=json

[profile user1]
region=us-east-1
output=text
```

### IAM role

Este ejemplo sirve para asumir un IAM rol. Los perfiles que utilizan IAM funciones obtienen las credenciales de otro perfil y, a continuación, aplican los permisos de IAM función. En los siguientes ejemplos, `default` es el perfil de origen de las credenciales y `user1` toma prestadas las mismas credenciales y, a continuación, asume un nuevo rol. Para obtener más información, consulte [the section called “Roles de IAM”](#).

#### Archivo de credenciales

El archivo `credentials` depende de la autenticación que utilice su perfil de origen. En el siguiente ejemplo, el perfil de origen utiliza credenciales de corta duración.

```
[default]
aws_access_key_id=ASIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
aws_session_token
= IQoJb3JpZ2luX2IqoJb3JpZ2luX2IqoJb3JpZ2luX2IqoJb3JpZ2luX2IqoJb3JpZVERYLONGSTRINGEXAMPLE
```

### Archivo de configuración

```
[default]
region=us-west-2
output=json

[profile user1]
role_arn=arn:aws:iam::777788889999:role/user1role
```

```
source_profile=default
role_session_name=session_user1
region=us-east-1
output=text
```

## Amazon EC2 instance metadata credentials

Este ejemplo es para las credenciales obtenidas de los metadatos de la EC2 instancia de Amazon de alojamiento. Para obtener más información, consulte [the section called “Usa metadatos de EC2 instancias de Amazon para AWS CLI las credenciales”](#).

### Archivo de credenciales

El archivo `credentials` no se utiliza para este método de autenticación.

### Archivo de configuración

```
[default]
role_arn=arn:aws:iam::123456789012:role/defaultrole
credential_source=Ec2InstanceMetadata
region=us-west-2
output=json

[profile user1]
role_arn=arn:aws:iam::777788889999:role/user1role
credential_source=Ec2InstanceMetadata
region=us-east-1
output=text
```

## Long-term credentials

### Warning

Para evitar riesgos de seguridad, no utilices a IAM los usuarios para autenticarte cuando desarrolles software específico o trabajes con datos reales. En cambio, utilice la federación con un proveedor de identidades como [AWS IAM Identity Center](#).

Este ejemplo corresponde a las credenciales de larga duración de AWS Identity and Access Management. Para obtener más información, consulte [the section called “IAM usuarios”](#).

### Archivo de credenciales

```
[default]
aws_access_key_id=AKIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY

[user1]
aws_access_key_id=AKIAI44QH8DHBEXAMPLE
aws_secret_access_key=je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY
```

## Archivo de configuración

```
[default]
region=us-west-2
output=json

[profile user1]
region=us-east-1
output=text
```

Para obtener más información y métodos adicionales de autorización y credenciales, consulte [the section called “IAMusuarios”](#).

## Tipo de sección: **services**

La `services` sección es un grupo de ajustes que configura puntos finales personalizados para las solicitudes. Servicio de AWS A continuación, se vincula un perfil a una sección de `services`.

```
[profile dev]
services = my-services
```

La sección de `services` se separa en subsecciones por líneas `<SERVICE> =`, donde `<SERVICE>` es la clave del identificador de Servicio de AWS. El Servicio de AWS identificador se basa en el del API modelo, sustituyendo todos los espacios `serviceId` por guiones bajos y minúsculas todas las letras. Para obtener una lista de todas las claves de identificación de servicio que se van a utilizar en la sección de `services`, consulte [Utilice puntos finales en el AWS CLI](#). La clave del identificador del servicio va seguida de configuraciones anidadas, cada una en su propia línea y marcada con dos espacios.

El siguiente ejemplo configura el punto final que se utilizará en las solicitudes realizadas al servicio en el Amazon DynamoDB `my-services` sección que se utiliza en el `dev` perfil. Todas las líneas



inmediatamente siguientes que estén sangradas se incluyen en esa subsección y se aplican a ese servicio.

```
[profile dev]
services = my-services

[services my-services]
dynamodb =
  endpoint_url = http://localhost:8000
```

Para obtener más información sobre puntos de conexión específicos del servicio, consulte [Utilice puntos finales en el AWS CLI](#).

Si su perfil tiene credenciales basadas en funciones configuradas mediante un `source_profile` parámetro para la funcionalidad de IAM asumir funciones, SDK solo utilizará configuraciones de servicio para el perfil especificado. No utiliza perfiles que estén vinculados a él por roles. Por ejemplo, mediante el siguiente archivo `config` compartido:

```
[profile A]
credential_source = Ec2InstanceMetadata
endpoint_url = https://profile-a-endpoint.aws/

[profile B]
source_profile = A
role_arn = arn:aws:iam::123456789012:role/roleB
services = profileB

[services profileB]
ec2 =
  endpoint_url = https://profile-b-ec2-endpoint.aws
```

Si utilizas el perfil B y haces una llamada en tu código a AmazonEC2, el punto final se resuelve como `https://profile-b-ec2-endpoint.aws`. Si el código realiza una solicitud a cualquier otro servicio, la resolución del punto de conexión no seguirá ninguna lógica personalizada. El punto de conexión no se convierte en el punto de conexión global definido en el perfil A. Para que un punto de conexión global surta efecto en el perfil B, tendrá que configurar `endpoint_url` directamente dentro del perfil B.

## ¿Dónde se almacenan las opciones de configuración?

AWS CLI Almacena la información confidencial sobre credenciales que especifique `aws configure` en un archivo local denominado `credentials`, en una carpeta denominada `.aws` en su directorio principal. Las opciones de configuración menos confidenciales especificadas con `aws configure` se guardan en un archivo local denominado `config`, que también se almacena en la carpeta `.aws` del directorio principal.

### Almacenamiento de credenciales en el archivo de configuración

Puede guardar toda la configuración de su perfil en un solo archivo, ya que AWS CLI puede leer las credenciales del `config` archivo. Si hay credenciales en ambos archivos para un perfil que comparte el mismo nombre, las claves del archivo de credenciales tienen prioridad. Sugerimos guardar las credenciales en los archivos `credentials`. Estos archivos también los utilizan los distintos kits de desarrollo de software lingüístico (SDKs). Si utiliza uno de ellos SDKs además del AWS CLI, confirme si las credenciales deben guardarse en su propio archivo.

La ubicación del directorio principal varía en función del sistema operativo, pero se hace referencia a él mediante las variables de entorno `%UserProfile%` en Windows y `$HOME` o `~` (tilde) en los sistemas basados en Unix. Puede especificar una ubicación no predeterminada para los archivos configurando las variables de entorno `AWS_CONFIG_FILE` y `AWS_SHARED_CREDENTIALS_FILE` en otra ruta local. Para obtener más información, consulte [Variables de entorno para configurar el AWS CLI](#).

Cuando se utiliza un perfil compartido que especifica un rol AWS Identity and Access Management (IAM), se AWS CLI llama a la AWS STS `AssumeRole` operación para recuperar las credenciales temporales. A continuación, se almacenan estas credenciales (en `~/.aws/cli/cache`). AWS CLI Los comandos siguientes utilizan las credenciales temporales almacenadas en caché hasta que caduquen y, en ese momento, las actualizan AWS CLI automáticamente.

## Usar perfiles con nombre

Si no se define explícitamente ningún perfil, se utilizará el perfil `default`.

Para usar un perfil con nombre, añada la opción `--profile profile-name` a su comando. En el siguiente ejemplo, se enumeran todas las EC2 instancias de Amazon que utilizan las credenciales y la configuración definidas en el `user1` perfil.

```
$ aws ec2 describe-instances --profile user1
```

Si desea utilizar un perfil con nombre para varios comandos, puede evitar especificarlo en cada comando mediante el establecimiento de la variable de entorno `AWS_PROFILE` como el perfil predeterminado. Puede invalidar esta configuración con el parámetro `--profile`.

Linux or macOS

```
$ export AWS_PROFILE=user1
```

Windows

```
C:\> setx AWS_PROFILE user1
```

El uso de [set](#) para configurar una variable de entorno cambia el valor usado hasta que finalice la sesión de símbolo de sistema actual o hasta que otorgue a la variable un valor diferente.

El uso de [setx](#) para establecer una variable de entorno cambia el valor de todos los shells de comandos que cree después de ejecutar el comando. No afecta a ningún shell de comandos que ya se esté ejecutando en el momento en que ejecuta el comando. Cierre y reinicie el shell de comandos para ver los efectos del cambio.

La configuración de la variable de entorno cambia el perfil predeterminado hasta que finalice la sesión del shell, o hasta que otorgue a la variable a un valor diferente. Puede hacer que las variables de entorno persistan en sesiones futuras poniéndolas en el script de inicio del shell. Para obtener más información, consulte [Variables de entorno para configurar el AWS CLI](#).

## Definir y ver las opciones de configuración mediante comandos

Hay varias formas de ver y establecer las opciones de configuración mediante comandos.

### [aws configure](#)

Ejecute este comando para establecer y ver rápidamente las credenciales de , la región y el formato de salida. En el ejemplo siguiente se muestran los valores de ejemplo.

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
```

```
AWS Secret Access Key [None]: wJa1rXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY  
Default region name [None]: us-west-2  
Default output format [None]: json
```

## [aws configure set](#)

Puede establecer las credenciales o las opciones de configuración mediante `aws configure set`. Especifique el perfil que desea ver o modificar con la opción `--profile`.

Por ejemplo, el siguiente comando configura `region` en el perfil denominado `integ`.

```
$ aws configure set region us-west-2 --profile integ
```

Para quitar una opción, utilice una cadena vacía como valor o elimine manualmente la opción de los archivos `config` y `credentials` en un editor de texto.

```
$ aws configure set cli_pager "" --profile integ
```

## [aws configure get](#)

Puede recuperar las credenciales o las opciones de configuración que haya establecido utilizando `aws configure get`. Especifique el perfil que desea ver o modificar con la opción `--profile`.

Por ejemplo, el siguiente comando recupera la opción `region` en el perfil denominado `integ`.

```
$ aws configure get region --profile integ  
us-west-2
```

Si la salida está vacía, significa que la configuración no está establecida de forma explícita y se utiliza el valor predeterminado.

## [aws configure list](#)

Para mostrar los datos de configuración, utilice el comando `aws configure list`. Este comando muestra el perfil, la clave de acceso, la clave secreta y la información de configuración de la región utilizada para el perfil especificado. Para cada elemento de configuración, muestra el valor, dónde se recuperó el valor de configuración y el nombre de la variable de configuración.

Por ejemplo, si proporciona el valor Región de AWS en una variable de entorno, este comando le mostrará el nombre de la región que ha configurado, que este valor proviene de una variable de entorno y el nombre de la variable de entorno.



este proceso, por lo que cada valor se establece con el comando `aws configure set`. Para obtener más información, consulte [the section called “Roles de IAM”](#).

```
$ aws configure set role_arn arn:aws:iam::123456789012:role/defaultrole
$ aws configure set source_profile default
$ aws configure set role_session_name session_user1
$ aws configure set region us-west-2
$ aws configure set output json
```

## Amazon EC2 instance metadata credentials

Este ejemplo es para las credenciales obtenidas de los metadatos de la EC2 instancia de Amazon de alojamiento. No existe ningún asistente para este proceso, por lo que cada valor se establece con el comando `aws configure set`. Para obtener más información, consulte [the section called “Usa metadatos de EC2 instancias de Amazon para AWS CLI las credenciales”](#).

```
$ aws configure set role_arn arn:aws:iam::123456789012:role/defaultrole
$ aws configure set credential_source Ec2InstanceMetadata
$ aws configure set region us-west-2
$ aws configure set output json
```

## Long-term credentials

### Warning

Para evitar riesgos de seguridad, no utilices a IAM los usuarios para autenticarte cuando desarrolles software específico o trabajes con datos reales. En cambio, utilice la federación con un proveedor de identidades como [AWS IAM Identity Center](#).

Este ejemplo corresponde a las credenciales de larga duración de AWS Identity and Access Management. Para obtener más información, consulte [the section called “IAM usuarios”](#).

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

## Configuraciones del archivo **config** admitidas

### Temas

- [Configuración global](#)
- [Configuración de comandos personalizados de S3](#)

Se admiten las siguientes opciones en el archivo `config`. Se utilizan los valores enumerados en el perfil especificado (o predeterminado) a menos que queden anulados por la presencia de una variable de entorno con el mismo nombre o una opción de línea de comandos con el mismo nombre. Para obtener más información sobre las opciones que tienen prioridad, consulte [Configure el AWS CLI](#)

### Configuración global

#### *api\_versions*

Algunos AWS servicios mantienen varias API versiones para permitir la compatibilidad con versiones anteriores. De forma predeterminada, AWS CLI los comandos utilizan la última API versión disponible. Puede especificar una API versión para utilizarla en un perfil incluyendo la `api_versions` configuración en el `config` archivo.

Se trata de una configuración «anidada» seguida de una o más líneas dentadas, cada una de las cuales identifica un AWS servicio y la API versión que se va a utilizar. Consulte la documentación de cada servicio para saber qué API versiones están disponibles.

El siguiente ejemplo muestra cómo especificar una API versión para dos AWS servicios. Estas API versiones se utilizan únicamente para los comandos que se ejecutan en el perfil que contiene esta configuración.

```
api_versions =  
  ec2 = 2015-03-01  
  cloudfront = 2015-09-017
```

Este ajuste no tiene una variable de entorno ni un parámetro de línea de comandos equivalente.

## aws\_access\_key\_id

Especifica la clave de AWS acceso utilizada como parte de las credenciales para autenticar la solicitud de comando. Aunque puede estar almacenada en el archivo `config`, recomendamos almacenar esta información en el archivo `credentials`.

Se puede anular mediante la variable de entorno `AWS_ACCESS_KEY_ID`. No se puede especificar el ID de clave de acceso como una opción de línea de comandos.

```
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
```

## aws\_secret\_access\_key

Especifica la clave AWS secreta utilizada como parte de las credenciales para autenticar la solicitud de comando. Aunque puede estar almacenada en el archivo `config`, recomendamos almacenar esta información en el archivo `credentials`.

Se puede anular mediante la variable de entorno `AWS_SECRET_ACCESS_KEY`. No se puede especificar la clave de acceso secreta como una opción de línea de comandos.

```
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
```

## aws\_session\_token

Especifica un token AWS de sesión. Un token de sesión solo es necesario si especifica manualmente credenciales de seguridad temporales. Aunque puede estar almacenada en el archivo `config`, recomendamos almacenar esta información en el archivo `credentials`.

Se puede anular mediante la variable de entorno `AWS_SESSION_TOKEN`. No se puede especificar un token de sesión como una opción de línea de comandos.

```
aws_session_token = AQoEXAMPLEH4aoAH0gNCAPyJxz4BlCFFxWNE1OPTgk5TthT  
+FvwnKwRc0IfrrRh3c/LTo6UDdyJw00vEVPvLXCrrrUtdnniCEXAMPLE/  
IvU1dYUg2RVAJBanLiHb4IgrmpRV3zrkuWJ0gQs8IZZaIv2BXIa2R40lgk
```

## ca\_bundle

Especifica un paquete de certificados de CA (un archivo con la `.pem` extensión) que se utiliza para comprobar SSL los certificados.



Se puede anular mediante la variable de entorno [AWS\\_CA\\_BUNDLE](#) o la opción de línea de comandos `--ca-bundle`.

```
ca_bundle = dev/apps/ca-certs/cabundle-2019mar05.pem
```

## cli\_follow\_urlparam

Especifica si los AWS CLI intentos de seguir URL los enlaces de los parámetros de la línea de comandos que comienzan por `http://` o `https://`. Cuando está activado, el contenido recuperado se utiliza como valor del parámetro en lugar del URL.

- `true` (verdadero): es el valor predeterminado. Cuando se configura, se obtienen todos los parámetros de cadena que comienzan por `http://` o `https://` y el contenido descargado se utiliza como valor del parámetro en el comando.
- `false`: si se especifica, AWS CLI no trata los valores de las cadenas de parámetros que comiencen por otras cadenas `http://` o de `https://` forma diferente a ellas.

Esta entrada no tiene una variable de entorno equivalente ni una opción de línea de comandos.

```
cli_follow_urlparam = false
```

## cli\_history

Está deshabilitado de forma predeterminada. Esta configuración habilita el historial de comandos para la AWS CLI. Tras activar esta configuración, AWS CLI registra el historial de `aws` comandos.

```
cli_history = enabled
```

Puede hacer una lista de su historial con el comando `aws history list` y utilizar los `command_ids` resultantes en el comando `aws history show` para obtener más información. Para obtener más información, consulte [aws history](#) en la Guía de referencia de AWS CLI .

## cli\_timestamp\_format

Especifica el formato de los valores de la marca temporal incluidos en la salida. Puede especificar cualquiera de los siguientes valores:

- `iso8601`: el valor predeterminado de la versión 2. AWS CLI [Si se especifica, AWS CLI reformatea todas las marcas de tiempo según 8601. ISO](#)

ISOLas marcas de tiempo formateadas 8601 tienen un aspecto similar al de los ejemplos siguientes. El primer ejemplo muestra la hora en la [Hora Universal Coordinada \(UTC\)](#) incluyendo una Z después de la hora. La fecha y la hora están separadas por una T.

```
2019-10-31T22:21:41Z
```

Para especificar una zona horaria diferente, en lugar de Z, especifique una + o - y el número de horas por delante o por detrás de la zona horaria deseada UTC, como un valor de dos dígitos. El siguiente ejemplo muestra la misma hora que en el ejemplo anterior, pero ajustada a la hora estándar del Pacífico, que tiene un retraso UTC de ocho horas.

```
2019-10-31T14:21:41-08
```

- cable: el valor por defecto de la AWS CLI versión 1. Si se especifica, AWS CLI muestra todos los valores de la marca de tiempo exactamente como se recibieron en la respuesta a la HTTP consulta.

Esta entrada no tiene una variable de entorno equivalente ni una opción de línea de comandos.

```
cli_timestamp_format = iso8601
```

## credential\_process

Especifica un comando externo que AWS CLI se ejecuta para generar o recuperar las credenciales de autenticación que se utilizarán en este comando. El comando debe devolver las credenciales en un formato específico. Para obtener más información sobre el uso de esta configuración, consulte [Obtener credenciales con un proceso externo](#).

Esta entrada no tiene una variable de entorno equivalente ni una opción de línea de comandos.

```
credential_process = /opt/bin/awscreds-retriever --username susan
```

## credential\_source

Se usa en EC2 las instancias o contenedores de Amazon para especificar dónde AWS CLI se pueden encontrar las credenciales que se utilizarán para asumir la función que especificó con el `role_arn` parámetro. No puede especificar `source_profile` y `credential_source` en el mismo perfil.

Este parámetro puede tener uno de tres valores:

- Entorno: especifica que AWS CLI se deben recuperar las credenciales de origen de las variables de entorno.
- Ec2 InstanceMetadata: especifica que AWS CLI se debe utilizar el IAM rol adjunto al [perfil de la EC2 instancia](#) para obtener las credenciales de origen.
- EcsContainer— Especifica que AWS CLI se utilizará el IAM rol adjunto al ECS contenedor como credenciales de origen.

```
credential_source = Ec2InstanceMetadata
```

## duration\_seconds

Especifica la duración máxima de la sesión de rol, en segundos. Este valor puede oscilar entre 900 segundos (15 minutos) y el valor de la duración máxima de la sesión para el rol (que puede ser 43200 como máximo). Se trata de un parámetro opcional y, de forma predeterminada, el valor se establece en 3600 segundos.

## endpoint\_url

Especifica el punto de conexión que se utiliza para todas las solicitudes de servicio. Si esta configuración se usa en la sección de [services](#) del archivo config, el punto de conexión se usa solo para el servicio especificado.

En el siguiente ejemplo, se utiliza el punto de conexión global `http://localhost:1234` y un punto de conexión específico del servicio de `http://localhost:4567` para Amazon S3.

```
[profile dev]
endpoint_url = http://localhost:1234
services = s3-specific

[services s3-specific]
s3 =
    endpoint_url = http://localhost:4567
```

Los valores de configuración del punto final se encuentran en varios lugares, como las variables de entorno del sistema o del usuario, los archivos de AWS configuración locales, o se declaran explícitamente en la línea de comandos como parámetros. Los ajustes de configuración del punto de conexión de la AWS CLI tienen prioridad en el siguiente orden:

1. La opción de línea de comandos [--endpoint-url](#).

2. Si está habilitada, la variable de entorno del punto de conexión [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global o la configuración del perfil [ignore\\_configure\\_endpoint\\_urls](#) para ignorar los puntos de conexión personalizados.
3. El valor proporcionado por una variable de entorno específica del servicio [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), como `AWS_ENDPOINT_URL_DYNAMODB`.
4. Los valores proporcionados por las variables de entorno [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#) y [AWS\\_ENDPOINT\\_URL](#).
5. El valor de punto de conexión específico del servicio proporcionado por la configuración [endpoint\\_url](#) de una sección de `services` del archivo compartido `config`.
6. El valor proporcionado por la configuración [endpoint\\_url](#) en un `profile` de un archivo compartido `config`.
7. Configuración de [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#) y [endpoint\\_url](#).
8. En último lugar se utiliza cualquier punto final predeterminado URL para el respectivo Servicio de AWS. Para obtener una lista de los puntos de enlace de servicio estándar disponibles en cada región, consulte [Regiones y puntos de enlace de AWS](#) en la Referencia general de Amazon Web Services.

## **ignore\_configure\_endpoint\_urls**

Si está activado, AWS CLI ignora todas las configuraciones de punto final personalizadas especificadas en el `config` archivo. Los valores válidos son **true** y **false**.

```
ignore_configure_endpoint_urls = true
```

Los ajustes de configuración de los puntos finales se encuentran en varios lugares, como las variables de entorno del sistema o del usuario, los archivos de AWS configuración locales, o se declaran explícitamente en la línea de comandos como parámetros. Los ajustes de configuración del punto de conexión de la AWS CLI tienen prioridad en el siguiente orden:

1. La opción de línea de comandos [--endpoint-url](#).
2. Si está habilitada, la variable de entorno del punto de conexión [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global o la configuración del perfil [ignore\\_configure\\_endpoint\\_urls](#) para ignorar los puntos de conexión personalizados.
3. El valor proporcionado por una variable de entorno específica del servicio [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), como `AWS_ENDPOINT_URL_DYNAMODB`.
4. Los valores proporcionados por las variables de entorno [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#) y [AWS\\_ENDPOINT\\_URL](#).

5. El valor de punto de conexión específico del servicio proporcionado por la configuración [endpoint\\_url](#) de una sección de `services` del archivo compartido `config`.
6. El valor proporcionado por la configuración [endpoint\\_url](#) en un `profile` de un archivo compartido `config`.
7. Configuración de [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#) y [endpoint\\_url](#).
8. En último lugar se utiliza cualquier punto final predeterminado URL para el respectivo Servicio de AWS. Para obtener una lista de los puntos de enlace de servicio estándar disponibles en cada región, consulte [Regiones y puntos de enlace de AWS](#) en la Referencia general de Amazon Web Services.

## [external\\_id](#)

Especifica un identificador único utilizado por terceros para adoptar un rol en las cuentas de los clientes. Se asigna al parámetro `ExternalId` en la operación `AssumeRole`. Este parámetro solo es necesario si la política de confianza del rol especifica un valor para `ExternalId`. Para obtener más información, consulte [Cómo utilizar un identificador externo al conceder acceso a sus AWS recursos a un tercero](#) en la Guía del IAM usuario.

## [max\\_attempts](#)

Especifica el valor máximo de intentos de reintento que utiliza el controlador de AWS CLI reintentos, donde la llamada inicial se tiene en cuenta para el `max_attempts` valor que usted proporciona.

Puede invalidar este valor mediante la variable de entorno de `AWS_MAX_ATTEMPTS`.

```
max_attempts = 3
```

## [mfa\\_serial](#)

El número de identificación de un MFA dispositivo que se va a utilizar al asumir una función. Esto es obligatorio solo si la política de confianza del rol que se asume incluye una condición que requiera la MFA autenticación. El valor puede ser un número de serie de un dispositivo de hardware (por ejemplo `GAHT12345678`) o un nombre de recurso de Amazon (ARN) para un MFA dispositivo virtual (por ejemplo `arn:aws:iam::123456789012:mfa/user`).

## output

Especifica el formato de la salida predeterminado para los comandos solicitados con este perfil. Puede especificar cualquiera de los siguientes valores:

- **json**— El resultado tiene el formato de una [JSON](#)cadena.
- **text**: la salida tiene el formato de varias líneas de valores de cadena separados por tabuladores. Esto puede ser útil para pasar la salida a un procesador de texto, como `grep`, `sed` o `awk`.
- **table**: el resultado tiene el formato de una tabla en la que se usan los caracteres `+|-` para los bordes de celda. Normalmente, la información se presenta en un formato que es más fácil de leer que los demás formatos, pero que no es útil para programar.

Se puede anular mediante la variable de entorno `AWS_DEFAULT_OUTPUT` o la opción de línea de comandos `--output`.

```
output = table
```

## **parameter\_validation**

Especifica si el AWS CLI cliente intenta validar los parámetros antes de enviarlos al punto final del AWS servicio.

- `true` (verdadero): es el valor predeterminado. Si se especifica, AWS CLI realiza la validación local de los parámetros de la línea de comandos.
- `false`: si se especifica, AWS CLI no valida los parámetros de la línea de comandos antes de enviarlos al punto final del AWS servicio.

Esta entrada no tiene una variable de entorno equivalente ni una opción de línea de comandos.

```
parameter_validation = false
```

## **region**

Especifica el Región de AWS lugar al que se deben enviar las solicitudes para los comandos solicitados mediante este perfil.

- Puede especificar cualquiera de los códigos de región disponibles para el servicio elegido que se muestran en [Regiones y puntos de conexión de AWS](#) en la Referencia general de Amazon Web Services.
- `aws_global` permite especificar el punto de enlace global para los servicios que admiten un punto de enlace global además de los puntos de enlace regionales, como AWS Security Token Service (AWS STS) y Amazon Simple Storage Service (Amazon S3).

Puede invalidar este valor mediante la variable de entorno `AWS_DEFAULT_REGION` o la opción de línea de comandos `--region`.

```
region = us-west-2
```

## retry\_mode

Especifica qué modo AWS CLI de reintento se utiliza. Hay tres modos de reintento disponibles: heredado (predeterminado), estándar y adaptativo. Para obtener más información acerca de los reintentos, consulte [AWS CLI vuelve a intentarlo](#).

Puede invalidar este valor mediante la variable de entorno de `AWS_RETRY_MODE`.

```
retry_mode = standard
```

## role\_arn

Especifica el nombre de recurso de Amazon (ARN) de un IAM rol que desea usar para ejecutar los AWS CLI comandos. También debe especificar uno de los siguientes parámetros para identificar las credenciales que tienen permiso para asumir este rol:

- `source_profile`
- `credential_source`

```
role_arn = arn:aws:iam::123456789012:role/role-name
```

La variable de entorno [AWS\\_ROLE\\_ARN](#) anula esta configuración.

Para obtener más información sobre el uso de las identidades web, consulte [the section called "Adopción de un rol con una identidad web"](#).

## role\_session\_name

Especifica el nombre que se va a asociar a la sesión de rol. Este valor se proporciona al `RoleSessionName` parámetro cuando AWS CLI llama a la `AssumeRole` operación y pasa a formar parte del rol asumido userARN: `arn:aws:sts::123456789012:assumed-role/role_name/role_session_name`. Se trata de un parámetro opcional. Si no proporciona este valor, se genera automáticamente un nombre de sesión. Este nombre aparece en los registros de AWS CloudTrail correspondientes a las entradas asociadas a esta sesión.

```
role_session_name = maria_garcia_role
```

La variable de entorno [AWS\\_ROLE\\_SESSION\\_NAME](#) anula esta configuración.

Para obtener más información sobre el uso de las identidades web, consulte [the section called “Adopción de un rol con una identidad web”](#).

## services

Especifica la configuración del servicio que se utilizará en el perfil.

```
[profile dev-s3-specific-and-global]
endpoint_url = http://localhost:1234
services = s3-specific

[services s3-specific]
s3 =
  endpoint_url = http://localhost:4567
```

Para obtener más información sobre la sección de `services`, consulte [the section called “services”](#).

La variable de entorno [AWS\\_ROLE\\_SESSION\\_NAME](#) anula esta configuración.

Para obtener más información sobre el uso de las identidades web, consulte [the section called “Adopción de un rol con una identidad web”](#).

## **sdk\_ua\_app\_id**

Varias aplicaciones de clientes Cuenta de AWS pueden utilizar una sola aplicación para realizar llamadas Servicios de AWS. El ID de aplicación identifica qué aplicación de origen realizó un conjunto de llamadas mediante un Servicio de AWS. AWS SDKs y los servicios no utilizan ni interpretan este valor más que para mostrarlo en las comunicaciones con los clientes. Por ejemplo, este valor se puede incluir en los correos electrónicos operativos para identificar de forma exclusiva qué aplicaciones están asociadas a la notificación.

El identificador de la aplicación es una cadena con una longitud máxima de 50 caracteres. Se permiten letras, números y los siguientes caracteres especiales: de forma ! \$ % & \* + - . , ^ \_ ` | ~ predeterminada, no se asigna ningún valor.

```
sdk_ua_app_id = prod1
```

Esta configuración se puede sobrescribir utilizando la variable de entorno [AWS\\_SDK\\_UA\\_APP\\_ID](#). No se puede establecer este valor como parámetro de la línea de comandos.



## source\_profile

Especifica un perfil con nombre con credenciales duraderas que la AWS CLI puede utilizar para asumir el rol que ha especificado con el parámetro `role_arn`. No puede especificar `source_profile` y `credential_source` en el mismo perfil.

```
source_profile = production-profile
```

## *sts\_regional\_endpoints*

Especifica cómo AWS CLI determina el punto final del AWS servicio que el AWS CLI cliente utiliza para comunicarse con el AWS Security Token Service (AWS STS). El valor predeterminado de la AWS CLI versión 1 es `legacy`.

Puede especificar uno de estos dos valores:

- **legacy**— Utiliza el STS punto final global, `sts.amazonaws.com`, para las siguientes AWS regiones: `ap-northeast-1`, `ap-south-1`, `ap-southeast-1`, `ap-southeast-2`, `aws-global`, `ca-central-1`, `eu-central-1`, `eu-north-1`, `eu-west-1`, `eu-west-2`, `eu-west-3`, `sa-east-1`, `us-east-1`, `us-east-2`, `us-west-1`, `yus-west-2`. Todas las demás regiones utilizan automáticamente su punto de conexión regional respectivo.
- **regional**— AWS CLI Siempre utiliza el AWS STS punto final de la región configurada actualmente. Por ejemplo, si el cliente está configurado para usar `us-west-2`, todas las llamadas AWS STS se realizan al punto final regional `sts.us-west-2.amazonaws.com` en lugar de al `sts.amazonaws.com` punto final global. Para enviar una solicitud al punto de enlace global mientras esta configuración está habilitada, puede establecer la región en `aws-global`.

Esta configuración se puede sobrescribir utilizando la variable de entorno `AWS_STS_REGIONAL_ENDPOINTS`. No se puede establecer este valor como parámetro de la línea de comandos.

## *use\_dualstack\_endpoint*

Permite el uso de puntos finales de doble pila para enviar AWS solicitudes. Para obtener más información sobre los puntos de enlace de doble pila, que admiten tanto como IPv4 el IPv6 tráfico,

consulte [Uso de los puntos de enlace de doble pila de Amazon S3 en la Guía del usuario](#) de Amazon Simple Storage Service. Los puntos de conexión de doble pila están disponibles para algunos servicios en algunas regiones. Si no existe un punto final de doble pila para el servicio o Región de AWS si la solicitud falla. Esta opción está deshabilitada de forma predeterminada.

Es mutuamente excluyente con la configuración `use_accelerate_endpoint`.

Los ajustes de configuración del punto final se encuentran en varios lugares, como las variables de entorno del sistema o del usuario, los archivos de AWS configuración locales, o se declaran explícitamente en la línea de comandos como parámetros. Los ajustes de configuración del punto de conexión de la AWS CLI tienen prioridad en el siguiente orden:

1. La opción de línea de comandos `--endpoint-url`.
2. Si está habilitada, la variable de entorno del punto de conexión `AWS_IGNORE_CONFIGURED_ENDPOINT_URLS` global o la configuración del perfil `ignore_configure_endpoint_urls` para ignorar los puntos de conexión personalizados.
3. El valor proporcionado por una variable de entorno específica del servicio `AWS_ENDPOINT_URL_<SERVICE>`, como `AWS_ENDPOINT_URL_DYNAMODB`.
4. Los valores proporcionados por las variables de entorno `AWS_USE_DUALSTACK_ENDPOINT`, `AWS_USE_FIPS_ENDPOINT` y `AWS_ENDPOINT_URL`.
5. El valor de punto de conexión específico del servicio proporcionado por la configuración `endpoint_url` de una sección de `services` del archivo compartido `config`.
6. El valor proporcionado por la configuración `endpoint_url` en un `profile` de un archivo compartido `config`.
7. Configuración de `use_dualstack_endpoint`, `use_fips_endpoint` y `endpoint_url`.
8. En último lugar se utiliza cualquier punto final predeterminado URL para el respectivo Servicio de AWS. Para obtener una lista de los puntos de enlace de servicio estándar disponibles en cada región, consulte [Regiones y puntos de enlace de AWS](#) en la Referencia general de Amazon Web Services.

### *use\_fips\_endpoint*

Algunos AWS servicios ofrecen terminales que, en algunos casos, son compatibles con la [Norma Federal de Procesamiento de la Información \(FIPS\) 140-2](#). Regiones de AWS Cuando el AWS servicio lo admite FIPS, esta configuración específica qué FIPS punto final AWS CLI debe usar. A diferencia de AWS los puntos finales estándar, los FIPS puntos finales utilizan una biblioteca de TLS software que cumple con FIPS la norma 140-2. Las empresas que trabajan con el gobierno de los Estados Unidos pueden requerir estos puntos de conexión.

Si esta configuración está habilitada, pero no existe un FIPS punto final para el servicio en su servidor Región de AWS, es posible que se produzca un error en el AWS comando. En este caso, especifique manualmente el punto de conexión que se va a utilizar en el comando mediante la opción `--endpoint-url` o utilice los [puntos de conexión específicos del servicio](#).

Para obtener más información sobre cómo especificar FIPS puntos de enlace por Región de AWS, consulte [FIPSPuntos finales por servicio](#).

Los ajustes de configuración de los terminales se encuentran en varios lugares, como las variables de entorno del sistema o del usuario, los archivos de AWS configuración locales, o se declaran explícitamente en la línea de comandos como parámetros. Los ajustes de configuración del punto de conexión de la AWS CLI tienen prioridad en el siguiente orden:

1. La opción de línea de comandos `--endpoint-url`.
2. Si está habilitada, la variable de entorno del punto de conexión `AWS_IGNORE_CONFIGURED_ENDPOINT_URLS` global o la configuración del perfil `ignore_configure_endpoint_urls` para ignorar los puntos de conexión personalizados.
3. El valor proporcionado por una variable de entorno específica del servicio `AWS_ENDPOINT_URL_<SERVICE>`, como `AWS_ENDPOINT_URL_DYNAMODB`.
4. Los valores proporcionados por las variables de entorno `AWS_USE_DUALSTACK_ENDPOINT`, `AWS_USE_FIPS_ENDPOINT` y `AWS_ENDPOINT_URL`.
5. El valor de punto de conexión específico del servicio proporcionado por la configuración `endpoint_url` de una sección de `services` del archivo compartido `config`.
6. El valor proporcionado por la configuración `endpoint_url` en un `profile` de un archivo compartido `config`.
7. Configuración de `use_dualstack_endpoint`, `use_fips_endpoint` y `endpoint_url`.
8. En último lugar se utiliza cualquier punto final predeterminado URL para el respectivo Servicio de AWS. Para obtener una lista de los puntos de enlace de servicio estándar disponibles en cada región, consulte [Regiones y puntos de enlace de AWS](#) en la Referencia general de Amazon Web Services.

### [web\\_identity\\_token\\_file](#)

Especifica la ruta a un archivo que contiene un token de acceso OAuth 2.0 o un token de ID de OpenID Connect proporcionado por un proveedor de identidad. La AWS CLI carga el contenido de este archivo y lo pasa como argumento `WebIdentityToken` a la operación `AssumeRoleWithWebIdentity`.

La variable de entorno [AWS\\_WEB\\_IDENTITY\\_TOKEN\\_FILE](#) anula esta configuración.

Para obtener más información sobre el uso de las identidades web, consulte [the section called “Adopción de un rol con una identidad web”](#).

### *tcp\_keepalive*

Especifica si el AWS CLI cliente usa paquetes TCP Keep-Alive.

Esta entrada no tiene una variable de entorno equivalente ni una opción de línea de comandos.

```
tcp_keepalive = false
```

## Configuración de comandos personalizados de S3

Amazon S3 admite varios ajustes que configuran la forma en que AWS CLI realiza las operaciones de Amazon S3. Algunos se aplican a todos los comandos de S3 tanto en los espacios de nombres `s3` como `s3api`. Otros son específicos para los comandos «personalizados» de S3, que abstraen operaciones comunes y hacen algo más que one-to-one mapear una API operación. Los comandos de `aws s3` de transferencia `cp`, `sync`, `mv` y `rm` tienen opciones adicionales que puede utilizar para controlar las transferencias de S3.

Para configurar todas estas opciones, especifique la configuración `s3` anidada en su archivo `config`. A continuación, se introduce con sangría cada configuración en su propia línea.

### Note

Estos ajustes son completamente opcionales. Debe poder utilizar los comandos de transferencia de `aws s3` sin configurar ninguno de estos ajustes. Estas opciones se ofrecen para afinar el rendimiento o para tener en cuenta el entorno específico en el que se ejecutan estos comandos de `aws s3`.

Estos ajustes se establecen con una clave de `s3` de alto nivel en el archivo `config`, tal y como se muestra en el siguiente ejemplo del perfil `development`.

```
[profile development]
s3 =
    max_concurrent_requests = 20
```

```
max_queue_size = 10000
multipart_threshold = 64MB
multipart_chunksize = 16MB
max_bandwidth = 50MB/s
use_accelerate_endpoint = true
addressing_style = path
```

Los siguientes ajustes se aplican a cualquier comando de S3 en el espacio de nombres `s3` o `s3api`.

## **addressing\_style**

Especifica qué estilo de direccionamiento debe usarse. Esto controla si el nombre del bucket está en el nombre del host o forma parte del URL. Los valores válidos son `path`, `virtual` y `auto`. El valor predeterminado es `auto`.

Hay dos estilos para crear un punto enlace de Amazon S3. El primero se llama `virtual` e incluye el nombre del bucket como parte del nombre de host. Por ejemplo: `https://bucketname.s3.amazonaws.com`. Como alternativa, con el `path` estilo, se trata el nombre del bucket como si fuera una ruta del URI; por ejemplo, `https://s3.amazonaws.com/bucketname`. El valor predeterminado CLI es `auto`, que intenta usar el `virtual` estilo siempre que puede, pero volverá a usar el `path` estilo cuando sea necesario. Por ejemplo, si el nombre del depósito no es DNS compatible, el nombre del depósito no puede formar parte del nombre de host y debe estar en la ruta. Con `auto` él, CLI detectará esta condición y cambiará automáticamente a su `path` estilo. Si estableces el estilo de direccionamiento en `path`, debes asegurarte de que la AWS región en la que configuraste AWS CLI coincida con la región de tu bucket.

## **payload\_signing\_enabled**

Especifica si se SHA256 firmarán las cargas útiles sigv4. De forma predeterminada, está deshabilitada para las subidas en streaming (`UploadPart`) cuando se utiliza HTTPS. De forma predeterminada, está configurado `false` para las subidas en streaming (`UploadPart`), pero solo si `ContentMD5` hay un (se genera de forma predeterminada) y el terminal lo usa. HTTPS

Si se establece en `true`, las solicitudes de S3 reciben una validación de contenido adicional en forma de una SHA256 suma de comprobación que se calcula automáticamente y se incluye en la firma de la solicitud. Si se establece en `false`, la suma de comprobación no se calcula. Deshabilitar esta opción puede ser útil para reducir la sobrecarga de rendimiento creada por el cálculo de la suma de comprobación.

## use\_accelerate\_endpoint

Utilice el punto de enlace de aceleración de Amazon S3 para todos los comandos `s3` y `s3api`. El valor predeterminado es `false`. Es mutuamente excluyente con la configuración `use_dualstack_endpoint`.

Si se establece en `true`, AWS CLI dirige todas las solicitudes de Amazon S3 al S3 Accelerate punto de conexión `ens3-accelerate.amazonaws.com`. Para utilizar este punto de enlace, debe habilitar el bucket para que use S3 Accelerate. Todas las solicitudes se envían mediante el direccionamiento al bucket de estilo virtual: *my-bucket*.s3-accelerate.amazonaws.com. Las solicitudes `ListBuckets`, `CreateBucket` y `DeleteBucket` no se envían al punto de enlace de S3 Accelerate, ya que este punto de enlace no admite estas operaciones. Este comportamiento también se puede establecer si el parámetro `--endpoint-url` está establecido en `https://s3-accelerate.amazonaws.com` o `http://s3-accelerate.amazonaws.com` para cualquier comando `s3api` o `s3`.

Estos ajustes de configuración solo se aplican a los comandos del conjunto de comandos del espacio de nombres de `s3`:

## max\_bandwidth

Especifica el ancho de banda máximo que puede ser consumido para cargar y descargar datos desde y hacia Amazon S3. El valor predeterminado es sin límite.

Esto limita el ancho de banda máximo que los comandos de S3 pueden utilizar para transferir datos desde y hacia Amazon S3. Este valor se aplica únicamente a cargas y descargas; no se aplica a copias o eliminaciones. El valor se expresa como bytes por segundo. El valor puede especificarse como:

- Un número entero. Por ejemplo, `1048576` establece el uso de ancho de banda máximo en 1 megabyte por segundo.
- Un número entero seguido de un sufijo de velocidad. Puede especificar sufijos de velocidad mediante: `KB/s`, `MB/s` o `GB/s`. Por ejemplo, `300KB/s`, `10MB/s`.

En general, se recomienda intentar reducir primero el consumo de ancho de banda, disminuyendo las `max_concurrent_requests`. Si así no se reduce el consumo de ancho de banda lo suficiente para conseguir la velocidad deseada, puede utilizar la opción `max_bandwidth` para limitar más el consumo de ancho de banda. Esto se debe a que `max_concurrent_requests` controla la cantidad de subprocesos que están ejecutando en este momento. Si en su lugar reduce

primero `max_bandwidth` pero deja `max_concurrent_requests` en un valor alto, puede provocar que los subprocesos tengan que esperar innecesariamente. Esto puede provocar un consumo excesivo de recursos y que se agoten los tiempos de espera de conexión.

## **max\_concurrent\_requests**

Especifica el número máximo de solicitudes concurrentes. El valor predeterminado es 10.

Los comandos de transferencia de `aws s3` tienen varios subprocesos. En un momento dado, es posible que se estén ejecutando varias solicitudes de Amazon S3. Por ejemplo, cuando usas el comando `aws s3 cp localdir s3://bucket/ --recursive` para cargar archivos en un bucket de S3, AWS CLI puedes cargar los archivos `localdir/file1` y `localdir/file3` en paralelo. `localdir/file2` La configuración `max_concurrent_requests` especifica el número máximo de operaciones de transferencia que se pueden ejecutar al mismo tiempo.

Es posible que tenga que cambiar este valor por varios motivos:

- Reducir el valor: en algunos entornos, el valor predeterminado de 10 solicitudes simultáneas puede ser demasiado para un sistema. Esto puede provocar tiempos de espera de conexión o ralentizar la capacidad de respuesta del sistema. Al reducir este valor, los comandos de transferencia de S3 consumen menos recursos. La desventaja es que las transferencias de S3 puede tardar más en completarse. Puede ser necesario reducir este valor si utiliza una herramienta para limitar el ancho de banda.
- Aumentar el valor: en algunos casos, puede ser conveniente que las transferencias de Amazon S3 se completen lo más rápido posible, con el mayor ancho de banda de red según sea necesario. En este caso, el número predeterminado de solicitudes simultáneas podría no ser suficiente para utilizar todo el ancho de banda de red disponible. Al aumentar este valor puede mejorar el tiempo que se tarda en completar una transferencia de Amazon S3.

## **max\_queue\_size**

Especifica el número máximo de tareas en la cola. El valor predeterminado es 1000.

AWS CLI internamente, utiliza un modelo en el que pone en cola las tareas de Amazon S3 que luego son ejecutadas por consumidores cuyo número está limitado por `max_concurrent_requests`. Una tarea, por lo general, se asigna a una sola operación de Amazon S3. Por ejemplo, una tarea puede ser `PutObjectTask`, `GetObjectTask` o `UploadPartTask`. La velocidad a la que se añaden tareas a la cola puede ser mucho más rápida que la velocidad en la que los consumidores finalizan las tareas. Para evitar el crecimiento ilimitado, el tamaño de la cola de tareas se limita a un tamaño específico. Esta configuración cambia el valor de ese número máximo.

Por lo general, no tendrá que cambiar este ajuste. Esta configuración también corresponde al número de tareas que AWS CLI sabe que deben ejecutarse. Esto significa que, de forma predeterminada, solo AWS CLI puede ver 1000 tareas por delante. Si se aumenta este valor, AWS CLI se puede saber más rápidamente el número total de tareas necesarias, suponiendo que el ritmo de espera sea más rápido que el ritmo de finalización de las tareas. La desventaja es que un valor mayor en `max_queue_size`, exige también más memoria.

### **multipart\_chunksize**

Especifica el tamaño del fragmento que se AWS CLI utiliza para las transferencias multipartes de archivos individuales. El valor predeterminado es 8 MB, con un mínimo de 5 MB.

Cuando la transferencia de un archivo supera el `multipart_threshold`, la AWS CLI divide el archivo en fragmentos de este tamaño. Este valor puede especificarse utilizando la misma sintaxis que en `multipart_threshold`, ya sea especificando el número de bytes como un número entero o especificando un tamaño y un sufijo.

### **multipart\_threshold**

Especifica el umbral de tamaño que se AWS CLI utiliza para las transferencias multipartes de archivos individuales. El valor predeterminado es 8 MB.

Al cargar, descargar o copiar un archivo, los comandos de Amazon S3 cambian a operaciones multiparte si el archivo supera este tamaño. Puede especificar este valor de una de las dos formas siguientes:

- El tamaño del archivo en bytes. Por ejemplo, 1048576.
- El tamaño del archivo con un sufijo de tamaño. Puede utilizar KB, MB, GB o TB. Por ejemplo: 10MB, 1GB.

#### Note

S3 puede imponer restricciones en los valores válidos que se pueden utilizar para las operaciones multiparte. Para obtener más información, consulte la [documentación Carga multiparte de S3](#) en la Guía de del usuario de Amazon Simple Storage Service.

## Variables de entorno para configurar el AWS CLI

Las variables de entorno proporcionan otra forma de especificar las opciones de configuración y las credenciales, y pueden ser útiles para la creación de scripts.



## Prioridad de las opciones

- Si especifica una opción mediante una de las variables de entorno que se describen en este tema, se invalidará cualquier valor carga desde un perfil en el archivo de configuración.
- Si especifica una opción mediante un parámetro de la línea de AWS CLI comandos, anulará cualquier valor de la variable de entorno correspondiente o de un perfil del archivo de configuración.

Para obtener más información sobre la prioridad y cómo AWS CLI determina qué credenciales utilizar, consulte. [Configure el AWS CLI](#)

## Temas

- [Cómo configurar las variables de entorno](#)
- [AWS CLI variables de entorno compatibles](#)

## Cómo configurar las variables de entorno

En los siguientes ejemplos se muestra cómo se pueden configurar las variables de entorno para el usuario predeterminado.

### Linux or macOS

```
$ export AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE
$ export AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
$ export AWS_DEFAULT_REGION=us-west-2
```

La configuración de la variable de entorno cambia el valor usado hasta el final de su sesión del intérprete de comandos o hasta que otorgue a la variable un valor diferente. Puede hacer que las variables persistan en sesiones futuras configurándolas en el script de startup del intérprete de comandos.

### Windows Command Prompt

Para configurar todas las sesiones

```
C:\> setx AWS_ACCESS_KEY_ID AKIAIOSFODNN7EXAMPLE
C:\> setx AWS_SECRET_ACCESS_KEY wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
C:\> setx AWS_DEFAULT_REGION us-west-2
```

El uso de `setx` para establecer una variable de entorno cambia el valor usado en la sesión de símbolo del sistema actual y en todas las sesiones de símbolo del sistema que cree después de ejecutar el comando. La operación no afecta a otros comandos del shell que ya se están ejecutando en el momento de ejecutar el comando. Puede que tenga que reiniciar el terminal para que se cargue la configuración.

Para configurar solo para la sesión actual

El uso de `set` para configurar una variable de entorno cambia el valor usado hasta que finalice la sesión de símbolo de sistema actual o hasta que otorgue a la variable un valor diferente.

```
C:\> set AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE
C:\> set AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
C:\> set AWS_DEFAULT_REGION=us-west-2
```

## PowerShell

```
PS C:\> $Env:AWS_ACCESS_KEY_ID="AKIAIOSFODNN7EXAMPLE"
PS C:\> $Env:AWS_SECRET_ACCESS_KEY="wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY"
PS C:\> $Env:AWS_DEFAULT_REGION="us-west-2"
```

Si establece una variable de entorno en la PowerShell línea de comandos, como se muestra en los ejemplos anteriores, guardará el valor únicamente durante la sesión actual. Para que la configuración de la variable de entorno sea persistente en todas las sesiones PowerShell y en la línea de comandos, guárdela mediante la aplicación Sistema del Panel de control. Como alternativa, puede configurar la variable para todas las PowerShell sesiones futuras añadiéndola a su PowerShell perfil. Consulte la [PowerShell documentación](#) para obtener más información sobre cómo almacenar variables de entorno o cómo conservarlas en todas las sesiones.

## AWS CLI variables de entorno compatibles

AWS CLI Soporta las siguientes variables de entorno.

### **AWS\_ACCESS\_KEY\_ID**

Especifica una clave de AWS acceso asociada a una IAM cuenta.

Si se define, esta variable de entorno anula el valor de la configuración del perfil `aws_access_key_id`. No se puede especificar el ID de clave de acceso mediante una opción de línea de comandos.

## AWS\_CA\_BUNDLE

Especifica la ruta a un paquete de certificados que se utilizará para la validación de HTTPS certificados.

Si se define, esta variable de entorno anula el valor de la configuración del perfil [ca\\_bundle](#). Puede anular esta variable de entorno mediante el parámetro de la línea de comandos [--ca-bundle](#).

## AWS\_CLI\_S3\_MV\_VALIDATE\_SAME\_S3\_PATHS

Si los cubos de origen y destino son los mismos al utilizar el `s3 mv` comando `custom the`, el archivo u objeto fuente se puede mover sobre sí mismo, lo que puede provocar la eliminación accidental del archivo u objeto de origen. La variable de `AWS_CLI_S3_MV_VALIDATE_SAME_S3_PATHS` entorno y la `--validate-same-s3-paths` opción especifican si se debe validar el punto de acceso ARNs o los alias del punto de acceso en el origen o destino URIs de Amazon S3.

### Note

La validación de rutas `s3 mv` requiere API llamadas adicionales.

## AWS\_CONFIG\_FILE

Especifica la ubicación del archivo que AWS CLI utiliza para almacenar los perfiles de configuración. La ruta predeterminada es `~/.aws/config`.

No puede especificar este valor en una configuración de perfil con nombre o mediante un parámetro de la línea de comandos.

## AWS\_DATA\_PATH

Una lista de directorios adicionales para comprobar fuera de la ruta de búsqueda integrada `~/.aws/models` al cargar AWS CLI los datos. La configuración de esta variable de entorno indica los directorios adicionales que hay que comprobar primero antes de recurrir a las rutas de búsqueda integradas. Varias entradas se deben separar con el carácter `os.pathsep` que es `:` en Linux o macOS y `;` en Windows.

## AWS\_DEFAULT\_OUTPUT

Especifica el [formato de salida](#) que se va a utilizar.

Si se define, esta variable de entorno anula el valor de la configuración del perfil `output`. Puede anular esta variable de entorno mediante el parámetro de la línea de comandos `--output`.

## **AWS\_DEFAULT\_REGION**

`Default region name` identifica la AWS región a cuyos servidores quieres enviar tus solicitudes de forma predeterminada. Suele ser la región más cercana a usted, pero puede ser cualquier región. Por ejemplo, puede escribir `us-west-2` para utilizar EE.UU. Oeste (Oregón). Esta es la región a la que se envían todas las solicitudes posteriores, a menos que especifique lo contrario en un comando concreto.

### Note

Debe especificar una AWS región cuando utilice la AWS CLI, ya sea de forma explícita o configurando una región predeterminada. Para obtener una lista de las regiones disponibles, consulte [Regiones y puntos de enlace](#). Los designadores de región que utilizan AWS CLI son los mismos nombres que aparecen en AWS Management Console URLs los puntos finales del servicio.

Si se define, esta variable de entorno anula el valor de la configuración del perfil `region`. Puede anular esta variable de entorno mediante el parámetro de línea de `--region` comandos .

## **AWS\_EC2\_METADATA\_DISABLED**

Inhabilita el uso del servicio de metadatos de EC2 instancias de Amazon (IMDS).

Si se establece en `true`, no se solicitan las credenciales de usuario ni la configuración (como la región). IMDS

## **AWS\_ENDPOINT\_URL**

Especifica el punto de conexión que se utiliza para todas las solicitudes de servicio.

Los valores de configuración del punto final se encuentran en varios lugares, como las variables de entorno del sistema o del usuario o los archivos de AWS configuración locales, o se declaran explícitamente en la línea de comandos como parámetros. Los ajustes de configuración del punto de conexión de la AWS CLI tienen prioridad en el siguiente orden:

1. La opción de línea de comandos [--endpoint-url](#).

2. Si está habilitada, la variable de entorno del punto de conexión [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global o la configuración del perfil [ignore\\_configure\\_endpoint\\_urls](#) para ignorar los puntos de conexión personalizados.
3. El valor proporcionado por una variable de entorno específica del servicio [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), como `AWS_ENDPOINT_URL_DYNAMODB`.
4. Los valores proporcionados por las variables de entorno [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#) y [AWS\\_ENDPOINT\\_URL](#).
5. El valor de punto de conexión específico del servicio proporcionado por la configuración [endpoint\\_url](#) de una sección de `services` del archivo compartido `config`.
6. El valor proporcionado por la configuración [endpoint\\_url](#) en un `profile` de un archivo compartido `config`.
7. Configuración de [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#) y [endpoint\\_url](#).
8. En último lugar se utiliza cualquier punto final predeterminado URL para el respectivo Servicio de AWS. Para obtener una lista de los puntos de enlace de servicio estándar disponibles en cada región, consulte [Regiones y puntos de enlace de AWS](#) en la Referencia general de Amazon Web Services.

### **AWS\_ENDPOINT\_URL\_<SERVICE>**

Especifica un punto final personalizado que se utiliza para un servicio específico, donde `<SERVICE>` se sustituye por el Servicio de AWS identificador. Por ejemplo, Amazon DynamoDB tiene una `serviceId` de [DynamoDB](#). Para este servicio, la variable de URL entorno del punto final es `AWS_ENDPOINT_URL_DYNAMODB`.

Para obtener una lista de todas las variables de entorno específicas del servicio, consulte [Lista de identificadores específicos del servicio](#).

Los valores de configuración del punto final se encuentran en varios lugares, como las variables de entorno del sistema o del usuario, los archivos de AWS configuración locales, o se declaran explícitamente en la línea de comandos como parámetros. Los ajustes de configuración del punto de conexión de la AWS CLI tienen prioridad en el siguiente orden:

1. La opción de línea de comandos [--endpoint-url](#).
2. Si está habilitada, la variable de entorno del punto de conexión [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global o la configuración del perfil [ignore\\_configure\\_endpoint\\_urls](#) para ignorar los puntos de conexión personalizados.
3. El valor proporcionado por una variable de entorno específica del servicio [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), como `AWS_ENDPOINT_URL_DYNAMODB`.

4. Los valores proporcionados por las variables de entorno [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#) y [AWS\\_ENDPOINT\\_URL](#).
5. El valor de punto de conexión específico del servicio proporcionado por la configuración [endpoint\\_url](#) de una sección de `services` del archivo compartido `config`.
6. El valor proporcionado por la configuración [endpoint\\_url](#) en un `profile` de un archivo compartido `config`.
7. Configuración de [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#) y [endpoint\\_url](#).
8. En último lugar se utiliza cualquier punto final predeterminado URL para el respectivo Servicio de AWS . Para obtener una lista de los puntos de enlace de servicio estándar disponibles en cada región, consulte [Regiones y puntos de enlace de AWS](#) en la Referencia general de Amazon Web Services.

## AWS\_IGNORE\_CONFIGURED\_ENDPOINT\_URLS

Si está activado, AWS CLI ignora todas las configuraciones de punto final personalizadas. Los valores válidos son **true** y **false**.

Los ajustes de configuración de los puntos finales se encuentran en varios lugares, como las variables de entorno del sistema o del usuario, los archivos de AWS configuración locales, o se declaran explícitamente en la línea de comandos como parámetros. Los ajustes de configuración del punto de conexión de la AWS CLI tienen prioridad en el siguiente orden:

1. La opción de línea de comandos [--endpoint-url](#).
2. Si está habilitada, la variable de entorno del punto de conexión [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global o la configuración del perfil [ignore\\_configure\\_endpoint\\_urls](#) para ignorar los puntos de conexión personalizados.
3. El valor proporcionado por una variable de entorno específica del servicio [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), como `AWS_ENDPOINT_URL_DYNAMODB`.
4. Los valores proporcionados por las variables de entorno [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#) y [AWS\\_ENDPOINT\\_URL](#).
5. El valor de punto de conexión específico del servicio proporcionado por la configuración [endpoint\\_url](#) de una sección de `services` del archivo compartido `config`.
6. El valor proporcionado por la configuración [endpoint\\_url](#) en un `profile` de un archivo compartido `config`.
7. Configuración de [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#) y [endpoint\\_url](#).
8. En último lugar se utiliza cualquier punto final predeterminado URL para el respectivo Servicio de AWS . Para obtener una lista de los puntos de enlace de servicio estándar disponibles

en cada región, consulte [Regiones y puntos de enlace de AWS](#) en la Referencia general de Amazon Web Services.

### **AWS\_MAX\_ATTEMPTS**

Especifica el valor máximo de reintentos que utiliza el controlador de AWS CLI reintentos, donde la llamada inicial se tiene en cuenta para el valor que usted proporciona. Para obtener más información acerca de los reintentos, consulte [AWS CLI vuelve a intentarlo](#).

Si se define, esta variable de entorno anula el valor de la configuración de los perfiles `max_attempts`.

### **AWS\_METADATA\_SERVICE\_NUM\_ATTEMPTS**

Al intentar recuperar las credenciales en una EC2 instancia de Amazon que se ha configurado con un IAM rol, AWS CLI intenta recuperar las credenciales una vez del servicio de metadatos de la instancia antes de detenerse. Si sabes que tus comandos se ejecutarán en una EC2 instancia de Amazon, puedes aumentar este valor para AWS CLI volver a intentarlo varias veces antes de darte por vencido.

### **AWS\_METADATA\_SERVICE\_TIMEOUT**

El número de segundos antes de que una conexión al servicio de metadatos de la instancia agote el tiempo de espera. Al intentar recuperar las credenciales de una EC2 instancia de Amazon que está configurada con un IAM rol, se agota el tiempo de espera de la conexión al servicio de metadatos de la instancia después de 1 segundo de forma predeterminada. Si sabes que estás ejecutando una EC2 instancia de Amazon con un IAM rol configurado, puedes aumentar este valor si es necesario.

### **AWS\_PROFILE**

Especifica el nombre del AWS CLI perfil con las credenciales y las opciones que se van a utilizar. Puede ser el nombre de un perfil almacenado en un archivo `credentials` o `config`, o el valor `default` para utilizar el perfil predeterminado.

Si se define, esta variable de entorno anula el comportamiento de utilizar el perfil llamado `[default]` en el archivo de configuración. Puede anular esta variable de entorno mediante el parámetro de la línea de comandos `--profile`.

### **AWS\_RETRY\_MODE**

Especifica qué modo de reintento se AWS CLI utiliza. Hay tres modos de reintento disponibles: heredado (predeterminado), estándar y adaptativo. Para obtener más información acerca de los reintentos, consulte [AWS CLI vuelve a intentarlo](#).

Si se define, esta variable de entorno anula el valor de la configuración de los perfiles `retry_mode`.

## **AWS\_ROLE\_ARN**

Especifica el nombre de recurso de Amazon (ARN) de un IAM rol con un proveedor de identidad web que desea usar para ejecutar los AWS CLI comandos.

Se utiliza con las variables de entorno `AWS_WEB_IDENTITY_TOKEN_FILE` y `AWS_ROLE_SESSION_NAME`.

Si se define, esta variable de entorno anula el valor de la configuración del perfil [role\\_arn](#). No se puede especificar un nombre de sesión de rol como parámetro de línea de comandos.

### Note

Esta variable de entorno solo se aplica a un rol asumido con proveedor de identidad web y no se aplican a la configuración general del proveedor de roles asumidos.

Para obtener más información sobre el uso de identidades web, consulte [the section called “Adopción de un rol con una identidad web”](#).

## **AWS\_ROLE\_SESSION\_NAME**

Especifica el nombre que se va a asociar a la sesión de rol. Este valor se proporciona al `RoleSessionName` parámetro cuando AWS CLI llama a la `AssumeRole` operación y pasa a formar parte del rol asumido userARN: `arn:aws:sts::123456789012:assumed-role/role_name/role_session_name`. Se trata de un parámetro opcional. Si no proporciona este valor, se genera automáticamente un nombre de sesión. Este nombre aparece en AWS CloudTrail los registros de las entradas asociadas a esta sesión.

Si se define, esta variable de entorno anula el valor de la configuración del perfil [role\\_session\\_name](#).

Se utiliza con las variables de entorno `AWS_ROLE_ARN` y `AWS_WEB_IDENTITY_TOKEN_FILE`.

Para obtener más información sobre el uso de identidades web, consulte [the section called “Adopción de un rol con una identidad web”](#).



**Note**

Esta variable de entorno solo se aplica a un rol asumido con proveedor de identidad web y no se aplican a la configuración general del proveedor de roles asumidos.

**AWS\_SDK\_UA\_APP\_ID**

Varias aplicaciones de clientes Cuenta de AWS pueden utilizar una sola aplicación para realizar llamadas Servicios de AWS. El ID de aplicación identifica qué aplicación de origen realizó un conjunto de llamadas mediante un Servicio de AWS. AWS SDKs y los servicios no utilizan ni interpretan este valor más que para mostrarlo en las comunicaciones con los clientes. Por ejemplo, este valor se puede incluir en los correos electrónicos operativos para identificar de forma exclusiva qué aplicaciones están asociadas a la notificación.

De forma predeterminada, no hay ningún valor.

El identificador de la aplicación es una cadena con una longitud máxima de 50 caracteres. Se permiten letras, números y los siguientes caracteres especiales:

! \$ % & \* + - . , ^ \_ ` | ~

Si se define, esta variable de entorno anula el valor de la configuración del perfil [sdk\\_ua\\_app\\_id](#). No puede especificar el ID de la aplicación como opción de línea de comandos.

**AWS\_SECRET\_ACCESS\_KEY**

Especifica la clave secreta asociada a la clave de acceso. Se trata básicamente de la "contraseña" de la clave de acceso.

Si se define, esta variable de entorno anula el valor de la configuración del perfil `aws_secret_access_key`. No se puede especificar la ID de clave de acceso secreta como una opción de línea de comandos.

**AWS\_SESSION\_TOKEN**

Especifica el valor del token de sesión que se requiere si utiliza credenciales de seguridad temporales que ha recuperado directamente de las operaciones de AWS STS . Para obtener más información, consulte la sección [Output del comando de rol asumido](#) en la Referencia de comando de AWS CLI .

Si se define, esta variable de entorno anula el valor de la configuración del perfil `aws_session_token`.

## **AWS\_SHARED\_CREDENTIALS\_FILE**

Especifica la ubicación del archivo que AWS CLI utiliza para almacenar las claves de acceso. La ruta predeterminada es `~/.aws/credentials`.

No puede especificar este valor en una configuración de perfil con nombre o mediante un parámetro de la línea de comandos.

## **AWS\_STS\_REGIONAL\_ENDPOINTS**

Especifica cómo AWS CLI determina el punto final del AWS servicio que el AWS CLI cliente utiliza para comunicarse con el AWS Security Token Service (AWS STS). El valor predeterminado para la versión 1 de AWS CLI es `legacy`.

Puede especificar uno de estos dos valores:

- **legacy**— Utiliza el STS punto final `globalsts.amazonaws.com`, para las siguientes AWS regiones: `ap-northeast-1`, `ap-south-1`, `ap-southeast-1`, `ap-southeast-2`, `aws-global`, `ca-central-1`, `eu-central-1`, `eu-north-1`, `eu-west-1`, `eu-west-2`, `eu-west-3`, `sa-east-1`, `us-east-1`, `us-east-2`, `us-west-1`, `yus-west-2`. Todas las demás regiones utilizan automáticamente su punto de conexión regional respectivo.
- **regional**— AWS CLI Siempre usa el AWS STS punto final de la región configurada actualmente. Por ejemplo, si el cliente está configurado para usar `us-west-2`, todas las llamadas AWS STS se realizan al punto final regional `sts.us-west-2.amazonaws.com` en lugar de al `sts.amazonaws.com` punto final global. Para enviar una solicitud al punto de enlace global mientras esta configuración está habilitada, puede establecer la región en `aws-global`.

## **AWS\_USE\_DUALSTACK\_ENDPOINT**

Permite el uso de puntos finales de doble pila para enviar AWS solicitudes. Para obtener más información sobre los puntos de enlace de doble pila, que admiten tanto como IPv4 el IPv6 tráfico, consulte [Uso de los puntos de enlace de doble pila de Amazon S3 en la Guía del usuario](#) de Amazon Simple Storage Service. Los puntos de conexión de doble pila están disponibles para algunos servicios en algunas regiones. Si no existe un punto final de doble pila para el servicio o Región de AWS si la solicitud falla. Esta opción está deshabilitada de forma predeterminada.

Los ajustes de configuración del punto final se encuentran en varios lugares, como las variables de entorno del sistema o del usuario, los archivos de AWS configuración locales, o se declaran

explícitamente en la línea de comandos como parámetros. Los ajustes de configuración del punto de conexión de la AWS CLI tienen prioridad en el siguiente orden:

1. La opción de línea de comandos [--endpoint-url](#).
2. Si está habilitada, la variable de entorno del punto de conexión [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global o la configuración del perfil [ignore\\_configure\\_endpoint\\_urls](#) para ignorar los puntos de conexión personalizados.
3. El valor proporcionado por una variable de entorno específica del servicio [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), como [AWS\\_ENDPOINT\\_URL\\_DYNAMODB](#).
4. Los valores proporcionados por las variables de entorno [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#) y [AWS\\_ENDPOINT\\_URL](#).
5. El valor de punto de conexión específico del servicio proporcionado por la configuración [endpoint\\_url](#) de una sección de `services` del archivo compartido `config`.
6. El valor proporcionado por la configuración [endpoint\\_url](#) en un `profile` de un archivo compartido `config`.
7. Configuración de [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#) y [endpoint\\_url](#).
8. En último lugar se utiliza cualquier punto final predeterminado URL para el respectivo Servicio de AWS. Para obtener una lista de los puntos de enlace de servicio estándar disponibles en cada región, consulte [Regiones y puntos de enlace de AWS](#) en la Referencia general de Amazon Web Services.

## AWS\_USE\_FIPS\_ENDPOINT

Algunos AWS servicios ofrecen terminales que, en algunos casos, son compatibles con la [Norma Federal de Procesamiento de la Información \(FIPS\) 140-2](#). Regiones de AWS Cuando el AWS servicio lo admite FIPS, esta configuración especifica qué FIPS punto final AWS CLI debe usar. A diferencia de AWS los puntos finales estándar, los FIPS puntos finales utilizan una biblioteca de TLS software que cumple con FIPS la norma 140-2. Las empresas que trabajan con el gobierno de los Estados Unidos pueden requerir estos puntos de conexión.

Si esta configuración está habilitada, pero no existe un FIPS punto final para el servicio en su servidor Región de AWS, es posible que se produzca un error en el AWS comando. En este caso, especifique manualmente el punto de conexión que se va a utilizar en el comando mediante la opción [--endpoint-url](#) o utilice los [puntos de conexión específicos del servicio](#).

Para obtener más información sobre cómo especificar FIPS puntos de enlace por Región de AWS, consulte [FIPSPuntos finales por servicio](#).

Los ajustes de configuración de los terminales se encuentran en varios lugares, como las variables de entorno del sistema o del usuario, los archivos de AWS configuración locales, o se declaran explícitamente en la línea de comandos como parámetros. Los ajustes de configuración del punto de conexión de la AWS CLI tienen prioridad en el siguiente orden:

1. La opción de línea de comandos [--endpoint-url](#).
2. Si está habilitada, la variable de entorno del punto de conexión [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global o la configuración del perfil [ignore\\_configure\\_endpoint\\_urls](#) para ignorar los puntos de conexión personalizados.
3. El valor proporcionado por una variable de entorno específica del servicio [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), como [AWS\\_ENDPOINT\\_URL\\_DYNAMODB](#).
4. Los valores proporcionados por las variables de entorno [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#) y [AWS\\_ENDPOINT\\_URL](#).
5. El valor de punto de conexión específico del servicio proporcionado por la configuración [endpoint\\_url](#) de una sección de services del archivo compartido config.
6. El valor proporcionado por la configuración [endpoint\\_url](#) en un profile de un archivo compartido config.
7. Configuración de [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#) y [endpoint\\_url](#).
8. En último lugar se utiliza cualquier punto final predeterminado URL para el respectivo Servicio de AWS . Para obtener una lista de los puntos de enlace de servicio estándar disponibles en cada región, consulte [Regiones y puntos de enlace de AWS](#) en la Referencia general de Amazon Web Services.

## [AWS\\_WEB\\_IDENTITY\\_TOKEN\\_FILE](#)

Especifica la ruta a un archivo que contiene un token de acceso OAuth 2.0 o un token de ID de OpenID Connect proporcionado por un proveedor de identidad. La AWS CLI carga el contenido de este archivo y lo pasa como argumento `WebIdentityToken` a la operación `AssumeRoleWithWebIdentity`.

Se utiliza con las variables de entorno `AWS_ROLE_ARN` y `AWS_ROLE_SESSION_NAME`.

Si se define, esta variable de entorno anula el valor de la configuración del perfil `web_identity_token_file`.

Para obtener más información sobre el uso de identidades web, consulte [the section called "Adopción de un rol con una identidad web"](#).

**Note**

Esta variable de entorno solo se aplica a un rol asumido con proveedor de identidad web y no se aplican a la configuración general del proveedor de roles asumidos.

## Opciones de línea de comandos

En el AWS CLI, las opciones de línea de comandos son parámetros globales que puede utilizar para anular los ajustes de configuración predeterminados, cualquier ajuste de perfil correspondiente o parámetro de variable de entorno para ese único comando. No se pueden utilizar las opciones de línea de comandos para especificar credenciales directamente, aunque es posible especificar qué perfil se debe utilizar.

### Temas

- [Cómo utilizar las opciones de la línea de comandos](#)
- [AWS CLI opciones de línea de comandos globales compatibles](#)
- [Usos comunes de las opciones de la línea de comandos](#)

## Cómo utilizar las opciones de la línea de comandos

La mayoría de las opciones de la línea de comandos son cadenas simples, como el nombre del perfil `profile1` en el siguiente ejemplo:

```
$ aws s3 ls --profile profile1
example-bucket-1
example-bucket-2
...
```

Cada opción toma un argumento que requiere un espacio o un signo igual (=) que separa el argumento del nombre de la opción. Si el valor del argumento es una cadena que contiene un espacio, debe utilizar comillas en torno al argumento. Para obtener más información sobre los tipos de argumentos y el formato de los parámetros, consulte [Especifique los valores de los parámetros para AWS CLI](#).

## AWS CLI opciones de línea de comandos globales compatibles

En el, AWS CLI puede utilizar las siguientes opciones de línea de comandos para anular los ajustes de configuración predeterminados, cualquier ajuste de perfil correspondiente o parámetro de variable de entorno para ese único comando.

`--ca-bundle <string>`

Especifica el paquete de certificados de la entidad emisora de certificados (CA) que se va a utilizar al comprobar SSL los certificados.

Si se define, esta opción anula el valor de la configuración del perfil [ca\\_bundle](#) y la variable del entorno de [AWS\\_CA\\_BUNDLE](#).

`--cli-connect-timeout <integer>`

Especifica el máximo tiempo de conexión del conector en segundos. Si el valor se establece en cero (0), la conexión del socket espera de forma indefinida (bloqueo) y no se agota el tiempo de espera.

`--cli-read-timeout <integer>`

Especifica el máximo tiempo de lectura del conector en segundos. Si el valor se establece en cero (0), la lectura del socket espera de forma indefinida (bloqueo) y no se agota el tiempo de espera.

`--color <string>`

Especifica la compatibilidad con el color de salida. Los valores válidos son on, off y auto. El valor predeterminado es auto.

`--debug`

Modificador booleano que permite registrar la depuración. De forma AWS CLI predeterminada, proporciona información clara sobre cualquier éxito o error relacionado con los resultados de los comandos en la salida del comando. La `--debug` proporciona los registros completos de Python. Esto incluye información de diagnóstico adicional de `stderr` sobre la operación del comando que puede ser útil para resolver problemas si un comando proporciona resultados inesperados. Para ver fácilmente los registros de depuración, sugerimos enviar los registros a un archivo para buscar la información más fácilmente. Puede hacer esto utilizando de los siguientes métodos.

Para enviar solo la información de diagnóstico de `stderr`, agregue `2> debug.txt` donde `debug.txt` es el nombre que desea usar para su archivo de depuración:

```
$ aws servicename commandname options --debug 2> debug.txt
```

Para enviar ambos la salida y la información de diagnóstico de `stderr`, agregue `&> debug.txt` donde `debug.txt` es el nombre que desea usar para su archivo de depuración:

```
$ aws servicename commandname options --debug &> debug.txt
```

### `--endpoint-url` *<string>*

Especifica el destinatario al que se debe URL enviar la solicitud. Para la mayoría de los comandos, determina AWS CLI automáticamente el en URL función del servicio seleccionado y de la AWS región especificada. Sin embargo, algunos comandos requieren que especifique una cuenta específica. URL También puedes configurar algunos AWS servicios para [alojar un punto final directamente en tu cuenta privada VPC](#), que puede que luego tengas que especificar.

El siguiente ejemplo de comando usa un punto de conexión Amazon S3 personalizadoURL.

```
$ aws s3 ls --endpoint-url http://localhost:4567
```

Los ajustes de configuración del punto final se encuentran en varios lugares, como las variables de entorno del sistema o del usuario, los archivos de AWS configuración locales, o se declaran explícitamente en la línea de comandos como parámetros. Los ajustes de configuración del punto de conexión de la AWS CLI tienen prioridad en el siguiente orden:

1. La opción de línea de comandos `--endpoint-url`.
2. Si está habilitada, la variable de entorno del punto de conexión `AWS_IGNORE_CONFIGURED_ENDPOINT_URLS` global o la configuración del perfil `ignore_configure_endpoint_urls` para ignorar los puntos de conexión personalizados.
3. El valor proporcionado por una variable de entorno específica del servicio `AWS_ENDPOINT_URL_<SERVICE>`, como `AWS_ENDPOINT_URL_DYNAMODB`.
4. Los valores proporcionados por las variables de entorno `AWS_USE_DUALSTACK_ENDPOINT`, `AWS_USE_FIPS_ENDPOINT` y `AWS_ENDPOINT_URL`.
5. El valor de punto de conexión específico del servicio proporcionado por la configuración `endpoint_url` de una sección de `services` del archivo compartido `config`.
6. El valor proporcionado por la configuración `endpoint_url` en un `profile` de un archivo compartido `config`.
7. Configuración de `use_dualstack_endpoint`, `use_fips_endpoint` y `endpoint_url`.

- En último lugar se utiliza cualquier punto final predeterminado URL para el respectivo Servicio de AWS . Para obtener una lista de los puntos de enlace de servicio estándar disponibles en cada región, consulte [Regiones y puntos de enlace de AWS](#) en la Referencia general de Amazon Web Services.

`--no-paginate`


Un conmutador booleano que desactiva las llamadas múltiples y recibe automáticamente todos los resultados de los comandos, AWS CLI lo que crea una paginación del resultado. Esto significa que solo se muestra la primera página de su salida.

`--no-sign-request`

Un conmutador booleano que desactiva la firma de las solicitudes en el punto final del servicio. HTTP AWS Impide que se carguen las credenciales.

`--no-verify-ssl`

De forma predeterminada, se AWS CLI utiliza SSL cuando se comunica con los servicios. AWS Para cada SSL conexión y llamada, AWS CLI verifica los SSL certificados. El uso de esta opción anula el comportamiento predeterminado de la verificación SSL de los certificados.

 Warning

Esta opción no es una práctica recomendada. Si la usa `--no-verify-ssl`, el tráfico entre el cliente y AWS los servicios ya no está protegido. Esto significa que su tráfico es un riesgo para la seguridad y vulnerable a los man-in-the-middle exploits. Si tiene problemas con los certificados, es mejor que los resuelva. Para conocer los pasos para solucionar problemas de certificados, consulte [the section called “SSLerrores de certificado”](#).

`--output <string>`

Especifica el formato de salida que se va a utilizar con este comando. Puede especificar cualquiera de los siguientes valores:

- `json`— La salida está formateada como una [JSON](#)cadena.
- `text`: la salida tiene el formato de varias líneas de valores de cadena separados por tabuladores. Esto puede ser útil para pasar la salida a un procesador de texto, como `grep`, `sed` o `awk`.



- **table**: el resultado tiene el formato de una tabla en la que se usan los caracteres +|- para los bordes de celda. Normalmente, la información se presenta en un formato que es más fácil de leer que los demás formatos, pero que no es útil para programar.

--profile **<string>**

Especifica el [perfil con nombre](#) que se va a utilizar con este comando. Para configurar perfiles con nombre adicionales, puede utilizar el comando `aws configure` con la opción `--profile`.

```
$ aws configure --profile <profilename>
```

--consulta **<string>**

Especifica una [JMESPathconsulta](#) que se utilizará para filtrar los datos de respuesta. Para obtener más información, consulte [AWS CLI Salida de filtro](#).

--region **<string>**

Especifica a qué AWS región se va a enviar la AWS solicitud de este comando. Para obtener una lista de todas las regiones que se pueden especificar, consulte [Regiones y puntos de enlace de AWS](#) en la Referencia general de Amazon Web Services.

--version

Un modificador booleano que muestra la versión actual del AWS CLI programa que se está ejecutando.

## Usos comunes de las opciones de la línea de comandos

Entre los usos habituales de las opciones de la línea de comandos se incluyen la comprobación de sus recursos en varias regiones de AWS y el cambio del formato de salida para obtener una mayor legibilidad o facilidad de uso al generar scripts. En los siguientes ejemplos, ejecutamos el comando `describe-instances` en cada región hasta que encontremos en qué región se encuentra la instancia.

```
$ aws ec2 describe-instances --output table --region us-west-1
-----
|DescribeInstances|
+-----+
$ aws ec2 describe-instances --output table --region us-west-2
-----
|                               DescribeInstances                               |
+-----+
```

```

||                               Reservations                               ||
|+-----+-----+-----+-----+-----+-----+-----+-----+-----+|
||  OwnerId                       |  012345678901                       ||
||  ReservationId                  |  r-abcdefgh                       ||
|+-----+-----+-----+-----+-----+-----+-----+-----+-----+|
|||                               Instances                               ||| |
||+-----+-----+-----+-----+-----+-----+-----+-----+-----+||
|||  AmiLaunchIndex                |  0                                 |||
|||  Architecture                  |  x86_64                           |||
...

```

## Finalización de comandos

El AWS Command Line Interface (AWS CLI) incluye una función de finalización de comandos compatible con bash que permite utilizar la tecla Tab para completar un comando introducido parcialmente. En la mayoría de los sistemas, debe configurarlo manualmente.

### Temas

- [Funcionamiento](#)
- [Configuración de la finalización de comandos en Linux o macOS](#)
- [Configuración de la finalización de comandos en Windows](#)

## Funcionamiento

Cuando se introduce parcialmente un comando, parámetro u opción, la función de finalización de comandos completa automáticamente el comando o muestra una lista sugerida de comandos. Para acelerar la finalización de un comando, se introduce parcialmente un comando y se pulsa la tecla de finalización, que suele ser *Tab* en la mayoría de las conchas.

En los ejemplos siguientes se muestran diferentes formas de utilizar la finalización de comandos:

- Introduzca parcialmente un comando y pulse *Tab* para mostrar una lista sugerida de comandos.

```

$ aws dynamodb dTAB
delete-backup                describe-global-table
delete-item                  describe-global-table-settings
delete-table                 describe-limits
describe-backup              describe-table
describe-continuous-backups  describe-table-replica-auto-scaling

```

```
describe-contributor-insights      describe-time-to-live
describe-endpoints
```

- Introduzca parcialmente un parámetro y pulse *Tab* para mostrar una lista de parámetros sugerida.

```
$ aws dynamodb delete-table --TAB
--ca-bundle           --endpoint-url       --profile
--cli-connect-timeout --generate-cli-skeleton --query
--cli-input-json      --no-paginate        --region
--cli-read-timeout    --no-sign-request    --table-name
--color               --no-verify-ssl      --version
--debug               --output
```

- Introduzca un parámetro y pulse *Tab* para mostrar una lista sugerida de valores de recursos. Esta función solo está disponible en la AWS CLI versión 2.

```
$ aws dynamodb db delete-table --table-name TAB
Table 1                Table 2                Table 3
```

## Configuración de la finalización de comandos en Linux o macOS

Para configurar la finalización de comandos en Linux o macOS, debe conocer el nombre del shell que se está utilizando y la ubicación del script `aws_completer`.

### Note

La finalización de comandos se configura y habilita automáticamente de forma predeterminada en EC2 las instancias de Amazon que ejecutan Amazon Linux.

### Temas

- [Confirme que la carpeta del Completer está en su ruta](#)
- [Habilitar la finalización de comandos](#)
- [Verifique la finalización de comandos](#)

## Confirme que la carpeta del Completer está en su ruta

Para que el AWS completador funcione correctamente, `aws_completer` debe estar en la ruta del shell. El comando `which` puede verificar si el Completer está en su ruta.

```
$ which aws_completer
/usr/local/bin/aws_completer
```

Si el comando que no puede encontrar el completador, siga los pasos siguientes para agregar la carpeta del completer a su ruta.

### Paso 1: Localiza el completador AWS

La ubicación del AWS completador puede variar según el método de instalación utilizado.

- **Package Manager:** programa como `pip`, `yumbrew`, por `apt-get` lo general, instala el AWS completador (o un enlace simbólico al mismo) en una ubicación de ruta estándar.
  - Si utilizó `pip` sin el parámetro `--user`, la ruta predeterminada es `/usr/local/bin/aws_completer`.
  - Si utilizó `pip` con el parámetro `--user`, la ruta predeterminada es `/home/username/.local/bin/aws_completer`.
- **Instalador agrupado:** si utilizó el instalador agrupado, la ruta predeterminada es `/usr/local/bin/aws_completer`.

Si todo lo demás falla, puede usar el `find` comando para buscar el completador en su sistema de archivos.

```
$ find / -name aws_completer
/usr/local/bin/aws_completer
```

### Paso 2: Identificación de su shell

Para identificar qué shell está utilizando, puede usar uno de los siguientes comandos.

- `echo $ SHELL`: muestra el nombre del archivo de programa del shell. Normalmente coincide con el nombre del shell en uso, a menos que haya iniciado otro shell tras iniciar sesión.

```
$ echo $SHELL
```

```
/bin/bash
```

- `ps`: muestra los procesos en ejecución para el usuario actual. Uno de ellos es el shell.

```
$ ps
  PID TTY          TIME CMD
 2148 pts/1    00:00:00 bash
 8756 pts/1    00:00:00 ps
```

### Paso 3: agregar la carpeta del completer a la ruta

1. Busque el script de perfil de su shell en su carpeta de usuario.

```
$ ls -a ~/
.  ..  .bash_logout  .bash_profile  .bashrc  Desktop  Documents  Downloads
```

- Bash: `.bash_profile`, `.profile` o `.bash_login`
  - Zsh: `.zshrc`
  - Tcsh: `.tcshrc`, `.cshrc` o `.login`
2. Añada un comando de exportación al final del script de su perfil igual que en el siguiente ejemplo. Reemplace `/usr/local/bin/` por el nombre de la carpeta que ha creado en la sección anterior.

```
export PATH=/usr/local/bin/:$PATH
```

3. Vuelva a cargar el perfil en la sesión actual para que esos cambios surtan efecto. Reemplace `.bash_profile` por el nombre del script de shell que ha creado en la primera sección.

```
$ source ~/.bash_profile
```

## Habilitar la finalización de comandos

Después de confirmar que el completer está en su ruta, habilite la finalización de comandos ejecutando el comando apropiado para el shell que está utilizando. Puede agregar el comando a su perfil del shell para que se ejecute cada vez que abra un nuevo shell. En cada comando, sustituya el `/usr/local/bin/` ruta por la que se encuentra en su sistema [Confirme que la carpeta del Completer está en su ruta.](#)

- **bash:** use el comando `complete` integrado.

```
$ complete -C '/usr/local/bin/aws_completer' aws
```

Agregue el comando anterior a `~/ .bashrc` para que se ejecute cada vez que abra un nuevo shell. Su `~/ .bash_profile` debe obtener `~/ .bashrc` para asegurarse de que el comando también se ejecuta en los shells de inicio de sesión.

- **zsh:** para ejecutar la finalización de comandos, debe ejecutar `bashcompinit` añadiendo la siguiente línea de carga automática al final del script de perfil de `~/ .zshrc`.

```
$ autoload bashcompinit && bashcompinit
$ autoload -Uz compinit && compinit
```

Para habilitar la finalización de comandos, utilice el comando integrado `complete`.

```
$ complete -C '/usr/local/bin/aws_completer' aws
```

Agregue el comando anterior a `~/ .zshrc` para que se ejecute cada vez que abra un nuevo shell.

- **tcsh:** la finalización de `tcsh` requiere un tipo de palabra y un patrón para definir el comportamiento de finalización.

```
> complete aws 'p*/`aws_completer`/'
```

Agregue el comando anterior a `~/ .tshrc` para que se ejecute cada vez que abra un nuevo shell.

Después de habilitar la finalización del comando, [Verifique la finalización de comandos](#) funciona.

## Verifique la finalización de comandos

Tras habilitar la finalización de comandos, vuelva a cargar su shell, escriba parcialmente un comando y pulse la tecla Tab (Tabulación) para ver los comandos disponibles.

```
$ aws sTAB
s3          ses          sqs          sts          swf
s3api       sns          storagegateway support
```

## Configuración de la finalización de comandos en Windows

### Note

Para obtener información sobre cómo PowerShell gestiona su finalización, incluídas sus distintas claves de finalización, consulte [about\\_TAB\\_Expansion en Microsoft Docs](#). PowerShell

Para habilitar la finalización de comandos PowerShell en Windows, complete los siguientes pasos en PowerShell

1. Abra el archivo \$PROFILE con el siguiente comando.

```
PS C:\> Notepad $PROFILE
```

Si no dispone de un \$PROFILE, cree un perfil de usuario mediante el siguiente comando.

```
PS C:\> if (!(Test-Path -Path $PROFILE ))  
{ New-Item -Type File -Path $PROFILE -Force }
```

Para obtener más información sobre PowerShell los perfiles, consulte [Cómo usar perfiles en Windows en](#) el PowerShell ISE sitio web de Microsoft Docs.

2. Para habilitar la finalización de comandos, agregue el siguiente bloque de código a su perfil, guarde y luego cierre el archivo.

```
Register-ArgumentCompleter -Native -CommandName aws -ScriptBlock {  
    param($commandName, $wordToComplete, $cursorPosition)  
    $env:COMP_LINE=$wordToComplete  
    if ($env:COMP_LINE.Length -lt $cursorPosition){  
        $env:COMP_LINE=$env:COMP_LINE + " "  
    }  
    $env:COMP_POINT=$cursorPosition  
    aws_completer.exe | ForEach-Object {  
        [System.Management.Automation.CompletionResult]::new($_, $_,  
'ParameterValue', $_)  
    }  
    Remove-Item Env:\COMP_LINE  
    Remove-Item Env:\COMP_POINT
```

```
}
```

- Tras habilitar la finalización de comandos, vuelva a cargar el shell, ingrese parcialmente un comando y pulse la tecla Tab (Tabulación) para recorrer los comandos disponibles.

```
$ aws sTab
```

```
$ aws s3
```

Para ver todos los comandos disponibles para completar, ingrese parcialmente un comando y pulse Ctrl (Control) + Space Espacio.

```
$ aws sCtrl + Space
s3          ses          sqs          sts          swf
s3api       sns          storagegateway support
```

## AWS CLI vuelve a intentarlo

En este tema se describe cómo AWS CLI es posible que las llamadas a AWS los servicios fallen debido a problemas inesperados. Estos problemas pueden ocurrir en el lado del servidor o puede fallar debido a la limitación de velocidad del servicio de AWS al que está intentando llamar. Este tipo de errores generalmente no requieren un manejo especial y la llamada se realiza de nuevo automáticamente, a menudo después de un breve período de espera. AWS CLI Ofrece numerosas funciones que ayudan a reintentar las llamadas de los clientes a AWS los servicios cuando se producen este tipo de errores o excepciones.

### Temas

- [Modos de reintento disponibles](#)
- [Configuración de un modo de reintento](#)
- [Visualización de registros de reintentos](#)

## Modos de reintento disponibles

AWS CLI Tiene varios modos para elegir según la versión:

- [Modo de reintento heredado](#)



- [Modo de reintento estándar](#)
- [Modo de reintento flexible](#)

## Modo de reintento heredado

El modo heredado es el modo predeterminado utilizado en la AWS CLI versión 1. El modo heredado utiliza un controlador de reintentos más antiguo con una funcionalidad limitada que incluye:

- Un valor predeterminado de 4 reintentos máximo, lo que hace un total de 5 intentos de llamada. Este valor se puede sobrescribir a través del parámetro de configuración de `max_attempts`.
- DynamoDB tiene un valor predeterminado de 9 reintentos como máximo, lo que hace un total de 10 intentos de llamada. Este valor se puede sobrescribir a través del parámetro de configuración de `max_attempts`.
- Reintentos para el siguiente límite de errores y excepciones:
  - Errores generales de toma o conexión:
    - `ConnectionError`
    - `ConnectionClosedError`
    - `ReadTimeoutError`
    - `EndpointConnectionError`
  - Errores y excepciones de límites y de la limitación controlada del lado del servicio:
    - `Throttling`
    - `ThrottlingException`
    - `ThrottledException`
    - `RequestThrottledException`
    - `ProvisionedThroughputExceededException`
- Vuelva a intentarlo con varios códigos de HTTP estado, incluidos 429, 500, 502, 503, 504 y 509.
- Cualquier reintento incluirá un retroceso exponencial por un factor base de 2.

## Modo de reintento estándar

El modo estándar es un conjunto estándar de reglas de reintento AWS SDKs con más funcionalidad que el anterior. El modo estándar se creó para la AWS CLI versión 2 y se retroporta a la AWS CLI versión 1. La funcionalidad del modo estándar incluye:

- Un valor predeterminado de 2 para los intentos máximos de reintento, haciendo un total de 3 intentos de llamada. Este valor se puede sobrescribir a través del parámetro de configuración de `max_attempts`.
- Reintentos para la siguiente lista ampliada de errores y excepciones:
  - Errores y excepciones transitorios
    - `RequestTimeout`
    - `RequestTimeoutException`
    - `PriorRequestNotComplete`
    - `ConnectionError`
    - `HTTPClientError`
  - Errores y excepciones de límites y de la limitación controlada del lado del servicio:
    - `Throttling`
    - `ThrottlingException`
    - `ThrottledException`
    - `RequestThrottledException`
    - `TooManyRequestsException`
    - `ProvisionedThroughputExceededException`
    - `TransactionInProgressException`
    - `RequestLimitExceeded`
    - `BandwidthLimitExceeded`
    - `LimitExceededException`
    - `RequestThrottled`
    - `SlowDown`
    - `EC2ThrottledException`
- Reintentos de códigos de error transitorios y no descriptivos. En concreto, estos códigos de HTTP estado: 500, 502, 503, 504.
- Cualquier reintento incluirá un retroceso exponencial por un factor base de 2 para un tiempo máximo de retroceso de 20 segundos.

## Modo de reintento flexible

### Warning

El modo flexible es un modo experimental y está sujeto a cambios, tanto en las características como en el comportamiento.

El modo de reintento flexible es un modo de reintento experimental que incluye todas las características del modo estándar. Además de las características del modo estándar, el modo flexible también presenta la limitación de la velocidad del lado del cliente mediante el uso del algoritmo token bucket y variables de límite de velocidad que se actualizan en forma dinámica con cada reintento. Este modo ofrece flexibilidad en los reintentos del lado del cliente y se adapta a la respuesta de error o estado de excepción de un servicio. AWS

Con cada nuevo intento, el modo adaptativo modifica las variables de límite de velocidad en función del código de error, excepción o HTTP estado presentado en la respuesta del servicio. AWS Estas variables de límite de velocidad se utilizan para calcular una nueva tasa de llamada para el cliente. Cada HTTP respuesta de excepción, error o fracaso (incluida en la lista anterior) de un AWS servicio actualiza las variables de límite de velocidad a medida que se reintentan hasta que se alcanza el éxito, se agota el depósito de fichas o se alcanza el valor máximo de intentos configurado.

## Configuración de un modo de reintento

AWS CLI Incluye una variedad de configuraciones de reintentos y métodos de configuración que se deben tener en cuenta al crear el objeto de cliente.

### Métodos de configuración disponibles

En el AWS CLI, los usuarios pueden configurar los reintentos de las siguientes maneras:

- Variables de entorno
- AWS CLI archivo de configuración

Los usuarios pueden personalizar las siguientes opciones de reintento:

- Modo de reintento: especifica qué modo de reintento utiliza. AWS CLI Como se ha comentado anteriormente, hay tres modos de reintento disponibles: heredado, estándar y adaptativo. El valor predeterminado de la AWS CLI versión 1 es la versión anterior. La AWS CLI estándar.

- Número máximo de intentos: especifica el valor máximo de reintentos que utiliza el controlador de AWS CLI reintentos, donde la llamada inicial cuenta para el valor que usted proporciona. El valor predeterminado es 5.

## Definición de una configuración de reintento en las variables de entorno

Para definir la configuración de reintentos AWS CLI, actualice las variables de entorno del sistema operativo.

Las variables de entorno de reintento son:

- AWS\_RETRY\_MODE
- AWS\_MAX\_ATTEMPTS

Para obtener más información sobre las variables de entorno, consulte [Variables de entorno para configurar el AWS CLI](#).

## Visualización de registros de reintentos

AWS CLI Utiliza la metodología de reintentos y el registro de Boto3. Puede utilizar la opción --debug en cualquier comando para recibir registros de depuración. Para obtener información sobre cómo utilizar la opción --debug, consulte [Opciones de línea de comandos](#).

Si busca “reintento” en sus registros de depuración, encontrará la información sobre de reintentos que necesita. Las entradas del registro del cliente correspondientes a los reintentos dependen del modo de reintento habilitado.

Modo heredado:

Los mensajes de reintento se generan mediante botocore.retryhandler. Verá uno de los tres mensajes:

- No retry needed
- Retry needed, action of: *<action\_name>*
- Reached the maximum number of retry attempts: *<attempt\_number>*

Modo estándar o flexible:

Los mensajes de reintento se generan a través de `botocore.retries.standard`. Verá uno de los tres mensajes:

- No retrying request
- Retry needed, retrying request after delay of: `<delay_value>`
- Retry needed but retry quota reached, not retrying request

[Para ver el archivo de definición completa de los reintentos de botocore, consulte `\_retry.json` en el repositorio de botocore. GitHub](#)

## Usa un HTTP proxy

Para acceder a AWS través de servidores proxy, puede configurar las variables de `HTTPS_PROXY` entorno `HTTP_PROXY` y las variables de entorno con los nombres de DNS dominio o las direcciones IP y los números de puerto que utilizan sus servidores proxy.

### Temas

- [Uso de los ejemplos de la](#)
- [Autenticación en un proxy](#)
- [Uso de un proxy en las EC2 instancias de Amazon](#)
- [Resolución de problemas](#)

## Uso de los ejemplos de la

### Note

En los ejemplos siguientes, se muestra el nombre de la variable de entorno con todas las letras en mayúscula. Sin embargo, si especifica una variable dos veces utilizando mayúsculas y minúsculas diferentes, prevalecen las letras minúsculas. Recomendamos que defina cada variable solo una vez para evitar la confusión de sistema y un comportamiento inesperado.

Los siguientes ejemplos muestran cómo puede utilizar la dirección IP explícita de su proxy o un DNS nombre que se resuelva en la dirección IP de su proxy. Cualquiera puede ir seguido de dos puntos y el número de puerto al que se deben enviar las consultas.

## Linux or macOS

```
$ export HTTP_PROXY=http://10.15.20.25:1234
$ export HTTP_PROXY=http://proxy.example.com:1234
$ export HTTPS_PROXY=http://10.15.20.25:5678
$ export HTTPS_PROXY=http://proxy.example.com:5678
```

## Windows Command Prompt

Para configurar todas las sesiones

```
C:\> setx HTTP_PROXY http://10.15.20.25:1234
C:\> setx HTTP_PROXY http://proxy.example.com:1234
C:\> setx HTTPS_PROXY http://10.15.20.25:5678
C:\> setx HTTPS_PROXY http://proxy.example.com:5678
```

El uso de [setx](#) para establecer una variable de entorno cambia el valor usado en la sesión de símbolo del sistema actual y en todas las sesiones de símbolo del sistema que cree después de ejecutar el comando. La operación no afecta a otros comandos del shell que ya se están ejecutando en el momento de ejecutar el comando.

Para configurar solo para la sesión actual

El uso de [set](#) para configurar una variable de entorno cambia el valor usado hasta que finalice la sesión de símbolo de sistema actual o hasta que otorgue a la variable un valor diferente.

```
C:\> set HTTP_PROXY=http://10.15.20.25:1234
C:\> set HTTP_PROXY=http://proxy.example.com:1234
C:\> set HTTPS_PROXY=http://10.15.20.25:5678
C:\> set HTTPS_PROXY=http://proxy.example.com:5678
```

## Autenticación en un proxy

### Note

AWS CLI No admite NTLM proxies. [Si utiliza un proxy de protocolo Kerberos NTLM o Kerberos, es posible que pueda conectarse a través de un proxy de autenticación como Cntlm.](#)

AWS CLI admite la autenticación básica HTTP. Especifique el nombre de usuario y la contraseña en el proxy URL de la siguiente manera.

Linux or macOS

```
$ export HTTP_PROXY=http://username:password@proxy.example.com:1234  
$ export HTTPS_PROXY=http://username:password@proxy.example.com:5678
```

Windows Command Prompt

Para configurar todas las sesiones

```
C:\> setx HTTP_PROXY http://username:password@proxy.example.com:1234  
C:\> setx HTTPS_PROXY http://username:password@proxy.example.com:5678
```

Para configurar solo la sesión actual

```
C:\> set HTTP_PROXY=http://username:password@proxy.example.com:1234  
C:\> set HTTPS_PROXY=http://username:password@proxy.example.com:5678
```

## Uso de un proxy en las EC2 instancias de Amazon

Si configuras un proxy en una EC2 instancia de Amazon lanzada con un IAM rol asociado, asegúrate de eximir la dirección utilizada para acceder a los [metadatos de la instancia](#). Para ello, establezca la variable de entorno NO\_PROXY en la dirección IP del servicio de metadatos de la instancia, 169.254.169.254. Esta dirección no varía.

Linux or macOS

```
$ export NO_PROXY=169.254.169.254
```

Windows Command Prompt

Para configurar todas las sesiones

```
C:\> setx NO_PROXY 169.254.169.254
```

Para configurar solo la sesión actual

```
C:\> set NO_PROXY=169.254.169.254
```

## Resolución de problemas

Si tienes problemas con el AWS CLI, consulta los pasos [Errores de solución de problemas](#) de solución de problemas. Para conocer los pasos más relevantes de la resolución de problemas, consulte [the section called “SSLerrores de certificado”](#).

## Utilice puntos finales en el AWS CLI

Para conectarse mediante programación a un punto final Servicio de AWS, utilice un punto final. Un punto final es el punto URL de entrada de un servicio AWS web. El AWS Command Line Interface (AWS CLI) utiliza automáticamente el punto final predeterminado para cada servicio de un Región de AWS, pero puedes especificar un punto final alternativo para tus API solicitudes.

Temas sobre puntos de conexión

- [Establecer punto de conexión para un solo comando](#)
- [Establezca un punto final global para todos Servicios de AWS](#)
- [Está configurado para usar FIPs puntos finales para todos Servicios de AWS](#)
- [Establecer para utilizar los puntos de conexión de doble pila para todos los Servicios de AWS](#)
- [Establecer puntos de conexión específicos del servicio](#)
  - [Puntos de conexión específicos del servicio: variables de entorno](#)
  - [Puntos de conexión específicos del servicio: archivo config compartido](#)
  - [Puntos de conexión específicos del servicio: lista de identificadores específicos del servicio](#)
- [Prioridad de la configuración y los ajustes del punto de conexión](#)

## Establecer punto de conexión para un solo comando

Para invalidar la configuración del punto de conexión o las variables de entorno de un solo comando, utilice la opción de línea de comandos [--endpoint-url](#). El siguiente ejemplo de comando usa un punto de conexión Amazon S3 personalizadoURL.

```
$ aws s3 ls --endpoint-url http://localhost:4567
```



## Establezca un punto final global para todos Servicios de AWS

Para dirigir las solicitudes de todos los servicios a un punto final personalizadoURL, utilice una de las siguientes configuraciones:

- Variables de entorno:

- [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#)- Ignore el punto final configuradoURLs.

Linux or macOS

```
$ export AWS_IGNORE_CONFIGURED_ENDPOINT_URLS=true
```

Windows Command Prompt

Para configurar todas las sesiones

```
C:\> setx AWS_IGNORE_CONFIGURED_ENDPOINT_URLS true
```

Para configurar solo la sesión actual

```
C:\> set AWS_IGNORE_CONFIGURED_ENDPOINT_URLS=true
```

PowerShell

```
PS C:\> $Env:AWS_IGNORE_CONFIGURED_ENDPOINT_URLS="true"
```

- [AWS\\_ENDPOINT\\_URL](#)- Establecer un punto final globalURL.

Linux or macOS

```
$ export AWS_ENDPOINT_URL=http://localhost:4567
```

Windows Command Prompt

Para configurar todas las sesiones

```
C:\> setx AWS_ENDPOINT_URL http://localhost:4567
```

Para configurar solo la sesión actual

```
C:\> set AWS_ENDPOINT_URL=http://localhost:4567
```

## PowerShell

```
PS C:\> $Env:AWS_ENDPOINT_URL="http://localhost:4567"
```

- El archivo config:
  - [ignore\\_configure\\_endpoint\\_urls](#)- Ignore el punto final configuradoURLs.

```
ignore_configure_endpoint_urls = true
```

- [endpoint\\_url](#)- Establecer un punto final globalURL.

```
endpoint_url = http://localhost:4567
```

Los puntos de conexión específicos del servicio y la opción de línea de comandos `--endpoint-url` invalidan cualquier punto de conexión global.

## Está configurado para usar FIPs puntos finales para todos Servicios de AWS

Para enrutar las solicitudes de todos los servicios para que utilicen FIPs puntos finales, utilice una de las siguientes opciones:

- [AWS\\_USE\\_FIPS\\_ENDPOINT](#) variable de entorno

### Linux or macOS

```
$ export AWS_USE_FIPS_ENDPOINT=true
```

### Windows Command Prompt

Para configurar todas las sesiones

```
C:\> setx AWS_USE_FIPS_ENDPOINT true
```

Para configurar solo la sesión actual

```
C:\> set AWS_USE_FIPS_ENDPOINT=true
```

## PowerShell

```
PS C:\> $Env:AWS_USE_FIPS_ENDPOINT="true"
```

- Configuración de archivos [use\\_fips\\_endpoint](#).

```
use_fips_endpoint = true
```

Algunos AWS servicios ofrecen terminales que, en algunos casos, son compatibles con la [Norma Federal de Procesamiento de Información \(FIPS\) 140-2](#). Regiones de AWS Cuando el AWS servicio lo admite FIPS, esta configuración especifica qué FIPS punto final AWS CLI debe usar. A diferencia de AWS los puntos finales estándar, los FIPS puntos finales utilizan una biblioteca de TLS software que cumple con FIPS la norma 140-2. Las empresas que trabajan con el gobierno de los Estados Unidos pueden requerir estos puntos de conexión.

Si esta configuración está habilitada, pero no existe un FIPS punto final para el servicio en su servidor Región de AWS, es posible que se produzca un error en el AWS comando. En este caso, especifique manualmente el punto de conexión que se va a utilizar en el comando mediante la opción [--endpoint-url](#) o utilice los [puntos de conexión específicos del servicio](#).

Para obtener más información sobre cómo especificar FIPS puntos de enlace por Región de AWS, consulte [FIPSPuntos finales por servicio](#).

## Establecer para utilizar los puntos de conexión de doble pila para todos los Servicios de AWS

Para dirigir las solicitudes de todos los servicios para usar puntos de conexión de doble pila cuando estén disponibles, utilice una de las siguientes configuraciones:

- [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#) variable de entorno

### Linux or macOS

```
$ export AWS_USE_DUALSTACK_ENDPOINT=true
```

## Windows Command Prompt

Para configurar todas las sesiones

```
C:\> setx AWS_USE_DUALSTACK_ENDPOINT true
```

Para configurar solo la sesión actual

```
C:\> set AWS_USE_DUALSTACK_ENDPOINT=true
```

## PowerShell

```
PS C:\> $Env:AWS_USE_DUALSTACK_ENDPOINT="true"
```

- Configuración de archivos [use\\_dualstack\\_endpoint](#).

```
use_dualstack_endpoint = true
```

Permite el uso de puntos finales de doble pila para enviar solicitudes. AWS Para obtener más información sobre los puntos de enlace de doble pila, que admiten tanto como IPv4 el IPv6 tráfico, consulte [Uso de los puntos de enlace de doble pila de Amazon S3 en la Guía del usuario](#) de Amazon Simple Storage Service. Los puntos de conexión de doble pila están disponibles para algunos servicios en algunas regiones. Si no existe un punto final de doble pila para el servicio o Región de AWS si la solicitud falla. Esta opción está deshabilitada de forma predeterminada.

## Establecer puntos de conexión específicos del servicio

La configuración de punto final específico del servicio ofrece la opción de usar un punto final persistente de su elección para las solicitudes. AWS CLI Esta configuración proporciona flexibilidad para admitir puntos finales locales, puntos finales y entornos de VPC desarrollo locales de terceros. AWS Se pueden usar diferentes puntos de conexión para los entornos de prueba y producción. Puede especificar un punto final URL para una persona. Servicios de AWS

Los puntos de conexión específicos del servicio se pueden especificar de las siguientes maneras:

- La opción de línea de comandos [--endpoint-url](#) para un solo comando.
- Variables de entorno:

- [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#)- Ignore todos los puntos finales configuradosURLs, a menos que se especifique en la línea de comandos.
- [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#): Especifica un punto de conexión personalizado que se utiliza para un servicio específico, donde <SERVICE> se sustituye por el identificador de Servicio de AWS . Para ver todas las variables específicas del servicio, consulte [the section called “Lista de identificadores específicos del servicio”](#).
- Archivo config:
  - [ignore\\_configure\\_endpoint\\_urls](#)- Ignore todos los puntos finales configuradosURLs, a menos que se especifique mediante variables de entorno o en la línea de comandos.
  - La sección [services](#) del archivo config combinada con la configuración del archivo [endpoint\\_url](#).

Temas de puntos de conexión específicos del servicio:

- [Puntos de conexión específicos del servicio: variables de entorno](#)
- [Puntos de conexión específicos del servicio: archivo config compartido](#)
- [Puntos de conexión específicos del servicio: lista de identificadores específicos del servicio](#)

## Puntos de conexión específicos del servicio: variables de entorno

Las variables de entorno invalidan la configuración del archivo de configuración, pero no invalidan las opciones especificadas en la línea de comandos. Use variables de entorno si quiere que todos los perfiles usen los mismos puntos de conexión en el dispositivo.

A continuación, se muestran variables de entorno específicas del servicio:

- [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#)- Ignore todos los terminales configuradosURLs, a menos que se especifique en la línea de comandos.

Linux or macOS

```
$ export AWS_IGNORE_CONFIGURED_ENDPOINT_URLS=true
```

Windows Command Prompt

Para configurar todas las sesiones

```
C:\> setx AWS_IGNORE_CONFIGURED_ENDPOINT_URLS true
```

Para configurar solo la sesión actual

```
C:\> set AWS_IGNORE_CONFIGURED_ENDPOINT_URLS=true
```

PowerShell

```
PS C:\> $Env:AWS_IGNORE_CONFIGURED_ENDPOINT_URLS="true"
```

- [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#)- Especifica un punto final personalizado que se utiliza para un servicio específico, donde <SERVICE> se sustituye por el Servicio de AWS identificador. Para ver todas las variables específicas del servicio, consulte [the section called “Lista de identificadores específicos del servicio”](#).

Los siguientes ejemplos de variables de entorno establecen un punto de conexión para AWS Elastic Beanstalk:

Linux or macOS

```
$ export AWS_ENDPOINT_URL_ELASTIC_BEANSTALK=http://localhost:4567
```

Windows Command Prompt

Para configurar todas las sesiones

```
C:\> setx AWS_ENDPOINT_URL_ELASTIC_BEANSTALK http://localhost:4567
```

Para configurar solo la sesión actual

```
C:\> set AWS_ENDPOINT_URL_ELASTIC_BEANSTALK=http://localhost:4567
```

PowerShell

```
PS C:\> $Env:AWS_ENDPOINT_URL_ELASTIC_BEANSTALK="http://localhost:4567"
```

Para obtener más información sobre la configuración de las variables de entorno, consulte [the section called “Variables de entorno”](#).

## Puntos de conexión específicos del servicio: archivo **config** compartido

En el archivo compartido `config`, `endpoint_url` se usa en varias secciones. Para configurar un punto de conexión específico del servicio, use la configuración anidada `endpoint_url` en una clave de identificador de servicio dentro de una sección `services`. Para obtener más información sobre cómo definir una sección de `services` en el archivo compartido `config`, consulte [the section called “services”](#).

En el siguiente ejemplo, se utiliza una `services` sección para configurar un punto de enlace específico del servicio URL para Amazon S3 y un punto de enlace global personalizado que se utiliza para todos los demás servicios:

```
[profile dev1]
endpoint_url = http://localhost:1234
services = s3-specific

[services testing-s3]
s3 =
  endpoint_url = http://localhost:4567
```

Un único perfil puede configurar puntos de conexión para varios servicios. El siguiente ejemplo establece el punto de enlace específico del servicio URLs para Amazon S3 y AWS Elastic Beanstalk en el mismo perfil.

Para obtener una lista de todas las claves de identificación de servicio que se van a utilizar en la sección de `services`, consulte [Lista de identificadores específicos del servicio](#).

```
[profile dev1]
services = testing-s3-and-eb

[services testing-s3-and-eb]
s3 =
  endpoint_url = http://localhost:4567
elastic_beanstalk =
  endpoint_url = http://localhost:8000
```

La sección de configuración de servicios se puede utilizar en varios perfiles. En el siguiente ejemplo, dos perfiles utilizan la misma definición de `services`:

```
[profile dev1]
output = json
services = testing-s3

[profile dev2]
output = text
services = testing-s3

[services testing-s3]
s3 =
  endpoint_url = https://localhost:4567
```

## Puntos de conexión específicos del servicio: lista de identificadores específicos del servicio

El Servicio de AWS identificador se basa en el del API modelo, sustituyendo todos los espacios `serviceId` por guiones bajos y poniendo todas las letras en minúsculas.

En el siguiente ejemplo de identificador de servicio se utiliza. AWS Elastic Beanstalk AWS Elastic Beanstalk tiene un `serviceId` de [Elastic Beanstalk](#), por lo que la clave del identificador de servicio es `elastic_beanstalk`.

La siguiente tabla muestra todos los identificadores específicos del servicio, las claves de archivo `config` y las variables de entorno.

## Prioridad de la configuración y los ajustes del punto de conexión

Los ajustes de configuración del punto final se encuentran en varios lugares, como las variables de entorno del sistema o del usuario, los archivos de AWS configuración locales, o se declaran explícitamente en la línea de comandos como parámetros. Los ajustes de configuración del punto de conexión de la AWS CLI tienen prioridad en el siguiente orden:

1. La opción de línea de comandos [--endpoint-url](#).
2. Si está habilitada, la variable de entorno del punto de conexión [AWS\\_IGNORE\\_CONFIGURED\\_ENDPOINT\\_URLS](#) global o la configuración del perfil [ignore\\_configure\\_endpoint\\_urls](#) para ignorar los puntos de conexión personalizados.
3. El valor proporcionado por una variable de entorno específica del servicio [AWS\\_ENDPOINT\\_URL\\_<SERVICE>](#), como `AWS_ENDPOINT_URL_DYNAMODB`.



4. Los valores proporcionados por las variables de entorno [AWS\\_USE\\_DUALSTACK\\_ENDPOINT](#), [AWS\\_USE\\_FIPS\\_ENDPOINT](#) y [AWS\\_ENDPOINT\\_URL](#).
5. El valor de punto de conexión específico del servicio proporcionado por la configuración [endpoint\\_url](#) de una sección de services del archivo compartido config.
6. El valor proporcionado por la configuración [endpoint\\_url](#) en un profile de un archivo compartido config.
7. Configuración de [use\\_dualstack\\_endpoint](#), [use\\_fips\\_endpoint](#) y [endpoint\\_url](#).
8. En último lugar se utiliza cualquier punto final predeterminado URL para el respectivo Servicio de AWS . Para obtener una lista de los puntos de enlace de servicio estándar disponibles en cada región, consulte [Regiones y puntos de enlace de AWS](#) en la Referencia general de Amazon Web Services.

## Credenciales de autenticación y acceso

Debe establecer cómo se AWS CLI autentica AWS cuando desarrolla con AWS los servicios. Para configurar las credenciales de acceso programático a AWS CLI, elija una de las siguientes opciones. Las opciones están en orden de recomendación.

¿Qué usuario necesita acceso programático?	Finalidad	Instrucciones
IAM	Usar a corto plazo.	<a href="#">the section called “Credenciales a corto plazo”</a>
IAM	Usar roles como credenciales.	<a href="#">the section called “Roles de IAM”</a>
IAM	(No recomendado) Usar credenciales a largo plazo.	<a href="#">the section called “IAMusuarios”</a>

## Prioridad de configuración y credenciales

Las credenciales y los valores de configuración se encuentran en varios lugares, como las variables de entorno del sistema o del usuario o los archivos de AWS configuración locales, o se declaran explícitamente en la línea de comandos como parámetros. Ciertas ubicaciones tienen prioridad sobre otras. Las credenciales de la AWS CLI y las opciones de configuración tienen prioridad en el siguiente orden:

1. [Opciones de línea de comandos](#): anula la configuración en cualquier otra ubicación, como los parámetros `--region`, `--output` y `--profile`.
2. [Variables de entorno](#): puede almacenar valores en las variables de entorno de su sistema.
3. [Asumir rol](#): asuma los permisos de un rol de IAM mediante la configuración o el comando `aws sts assume-role`.
4. [Asumir rol con identidad web](#): asuma los permisos de un rol de IAM con identidad web mediante la configuración o el comando `aws sts assume-role`.

5. [Archivo de credenciales](#): los archivos `credentials` y `config` se actualizan al ejecutar el comando `aws configure`. El archivo `credentials` se encuentra en `~/.aws/credentials` en Linux o macOS, o en `C:\Users\USERNAME\.aws\credentials` en Windows.
6. [Proceso personalizado](#): obtenga sus credenciales de un origen externo.
7. [Archivo de configuración](#): los archivos `credentials` y `config` se actualizan al ejecutar el comando `aws configure`. El archivo `config` se encuentra en `~/.aws/config` en Linux o macOS, o en `C:\Users\USERNAME\.aws\config` en Windows.
8. [Credenciales contenedor](#): puede asociar un rol de IAM con cada una de las definiciones de tareas de Amazon Elastic Container Service (Amazon ECS). Las credenciales temporales de ese rol estarán disponibles para los contenedores de esa tarea. Para obtener más información, consulte [Roles de IAM para tareas](#) en la Guía para desarrolladores de Amazon Elastic Container Service.
9. [Credenciales de perfil de instancia de Amazon EC2](#): puede asociar un rol de IAM con cada una de las instancias de Amazon Elastic Compute Cloud (Amazon EC2). Las credenciales temporales de ese rol estarán disponibles para el código que se ejecute en la instancia. Las credenciales se entregan a través del servicio de metadatos de Amazon EC2. Para obtener más información, consulte [Funciones de IAM para Amazon EC2](#) en la Guía del usuario de Amazon EC2 [y Uso de perfiles de instancia](#) en la Guía del usuario de IAM.

## Temas adicionales en esta sección

- [the section called “Credenciales a corto plazo”](#)
- [the section called “Roles de IAM”](#)
- [the section called “IAMusuarios”](#)
- [the section called “Usa metadatos de EC2 instancias de Amazon para AWS CLI las credenciales”](#)
- [the section called “Credenciales externas”](#)

## Autenticar con credenciales a corto plazo

Le recomendamos configurar su herramienta SDK o su herramienta para que utilice la [autenticación del Centro de IAM Identidad](#) con opciones de duración de sesión prolongada. Sin embargo, puede copiar y utilizar las credenciales temporales que están disponibles en el portal de AWS acceso. Las credenciales nuevas deberán copiarse cuando caduquen. Puede utilizar las credenciales temporales en un perfil o como valores para las propiedades del sistema y las variables de entorno.



## Utilice un IAM rol en el AWS CLI

Un [rol AWS Identity and Access Management \(IAM\)](#) es una herramienta de autorización que permite a un usuario obtener permisos adicionales (o diferentes) u obtener permisos para realizar acciones en una AWS cuenta diferente.

### Temas

- [Requisitos previos](#)
- [Descripción general del uso de IAM los roles](#)
- [Configuración y uso de un rol](#)
- [Uso de la autenticación multifactor](#)
- [Roles entre cuentas e ID externo](#)
- [Especificación de un nombre de sesión de rol para simplificar las auditorías](#)
- [Adopción de un rol con una identidad web](#)
- [Borrado de las credenciales almacenadas en memoria caché](#)

## Requisitos previos

Para utilizar los comandos de `iam`, debe instalar y configurar AWS CLI. Para obtener más información, consulte [Instalar la AWS CLI](#).

## Descripción general del uso de IAM los roles

Puede configurar el AWS Command Line Interface (AWS CLI) para que utilice un IAM rol definiendo un perfil para el rol en el `~/.aws/config` archivo.

En el siguiente ejemplo se muestra un perfil de rol llamado `marketingadmin`. Si ejecuta comandos con `--profile marketingadmin` (o los especifica con la [variable de PROFILE entorno AWS \\_](#)), AWS CLI utiliza las credenciales definidas en un perfil independiente `user1` para asumir el rol con el nombre de recurso de Amazon (ARN)`arn:aws:iam::123456789012:role/marketingadminrole`. Puede ejecutar cualquier operación permitida por los permisos asignados a ese rol.

```
[profile marketingadmin]  
role_arn = arn:aws:iam::123456789012:role/marketingadminrole
```

```
source_profile = user1
```

Debe especificar un `source_profile` que apunte a un perfil con nombre independiente que contenga credenciales de usuario con permiso para utilizar el rol. En el ejemplo anterior, el perfil `marketingadmin` utiliza las credenciales en el perfil `user1`. Al especificar que un AWS CLI comando debe utilizar el perfil `marketingadmin`, busca AWS CLI automáticamente las credenciales del `user1` perfil vinculado y las utiliza para solicitar credenciales temporales para el IAM rol especificado. Para ello, CLI utiliza la `AssumeRole` operación [sts](#): en segundo plano. Estas credenciales temporales se utilizan para ejecutar el comando de la AWS CLI solicitado. El rol especificado debe tener políticas de IAM permisos adjuntas que permitan ejecutar el AWS CLI comando solicitado.

Para ejecutar un AWS CLI comando desde una instancia de Amazon Elastic Compute Cloud (AmazonEC2) o un contenedor de Amazon Elastic Container Service (AmazonECS), puedes usar un IAM rol adjunto al perfil de la instancia o al contenedor. Si no especifica ningún perfil o no establece ninguna variable de entorno, ese rol se utiliza directamente. De este modo, puede evitar el almacenamiento de las claves de acceso de larga duración en las instancias. También puede utilizar esos roles de instancia o contenedor solo para obtener credenciales para otro rol. Para ello, utilice `credential_source` (en lugar de `source_profile`) para especificar cómo encontrar las credenciales. El atributo `credential_source` es compatible con los siguientes valores:

- `Environment`: recupera las credenciales fuente de las variables de entorno.
- `Ec2InstanceMetadata`— Utiliza el IAM rol adjunto al perfil de EC2 instancia de Amazon.
- `EcsContainer`— Utiliza la IAM función asociada al ECS contenedor de Amazon.

El siguiente ejemplo muestra la misma `marketingadminrole` función que se utiliza al hacer referencia a un perfil de EC2 instancia de Amazon.

```
[profile marketingadmin]
role_arn = arn:aws:iam::123456789012:role/marketingadminrole
credential_source = Ec2InstanceMetadata
```

Cuando invoca un rol, tiene otras opciones que puede exigir; por ejemplo, el uso de la autenticación multifactor y de un ID externo (que compañías de terceros utilizan para acceder a los recursos de sus clientes). También puede especificar nombres de sesión de rol únicos que se puedan auditar más fácilmente en AWS CloudTrail los registros.

## Configuración y uso de un rol

Cuando ejecutas comandos con un perfil que especifica un IAM rol, AWS CLI utiliza las credenciales del perfil de origen para llamar a AWS Security Token Service (AWS STS) y solicitar credenciales temporales para el rol especificado. El usuario del perfil fuente debe tener permiso para llamar a `sts:assume-role` en el rol del perfil especificado. El rol debe tener una relación de confianza que permita al usuario del perfil de origen utilizarlo. El proceso de recuperación y posterior uso de credenciales temporales en un rol a menudo se denomina asumir el rol.

Puede crear un rol IAM con los permisos que desee que asuman los usuarios siguiendo el procedimiento descrito en la sección [Creación de un rol para delegar permisos a un IAM usuario](#) en la Guía del AWS Identity and Access Management usuario. Si el rol y el usuario de del perfil de origen se encuentran en la misma cuenta, puede introducir su propio ID de cuenta al configurar la relación de confianza del rol.

Después de crear el rol, modifique la relación de confianza para permitir que el usuario de lo asuma.

En el siguiente ejemplo se muestra una política de confianza que puede asociarse a un rol. Esta política permite asumir el rol a cualquier usuario de la cuenta 123456789012, si el administrador de dicha cuenta le concede de forma explícita el permiso `sts:AssumeRole`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:root"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

La política de confianza no concede permisos. El administrador de la cuenta debe delegar el permiso para asumir el rol a usuarios individuales asociando una política a los permisos adecuados. En el siguiente ejemplo, se muestra una política que puede asociarse a un usuario y que le permite asumir únicamente el rol `marketingadminrole`. Para obtener más información sobre cómo conceder a un usuario acceso para que asuma un rol, consulte [Concesión de un permiso de usuario para cambiar de rol](#) en la Guía del IAM usuario.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": "arn:aws:iam::123456789012:role/marketingadminrole"
    }
  ]
}
```

El usuario no necesita tener permisos adicionales para ejecutar los AWS CLI comandos mediante el perfil de rol. En cambio, los permisos para ejecutar el comando proceden de los asociados al rol. Adjunta políticas de permisos al rol para especificar qué acciones se pueden realizar con qué AWS recursos. Para obtener más información sobre cómo asociar permisos a un rol (que funciona de forma idéntica a un usuario), consulte [Cambiar los permisos de un IAM usuario](#) en la Guía del IAM usuario.

Ahora que ya tiene el perfil del rol, los permisos del rol, la relación de confianza del rol y los permisos de usuario configurados correctamente, puede utilizar el rol en la línea de comandos invocando la opción `--profile`. Por ejemplo, el siguiente comando llama al comando Amazon S3 de `ls` utilizando los permisos adjuntos al rol `marketingadmin` que se definieron en el ejemplo que se incluye al principio de este tema.

```
$ aws s3 ls --profile marketingadmin
```

Si desea utilizar el rol para varias llamadas, puede configurar la variable de entorno `AWS_PROFILE` para la sesión actual desde la línea de comandos. Siempre que esa variable de entorno se haya definido, no será necesario especificar la opción `--profile` en cada comando.

Linux o macOS

```
$ export AWS_PROFILE=marketingadmin
```

Windows

```
C:\> setx AWS_PROFILE marketingadmin
```



Para obtener más información sobre la configuración de usuarios y roles, consulte [IAMIdenticidades \(usuarios, grupos de usuarios y roles\)](#) y [IAMroles](#) en la Guía del IAMusuario.

## Uso de la autenticación multifactor

Para mayor seguridad, puede exigir que los usuarios proporcionen una clave de un solo uso generada desde un dispositivo de autenticación multifactor (MFA), un dispositivo U2F o una aplicación móvil cuando intenten realizar una llamada mediante el perfil de rol.

En primer lugar, puede optar por modificar la relación de confianza en el IAM rol que desee. MFA Esto impide que cualquier persona utilice el rol sin autenticarse primero mediante el usoMFA. A modo de ejemplo, vea la línea `Condition` en el siguiente ejemplo. Esta política permite que el usuario nombrado asuma el rol `anika` al que está asociada la política, pero solo si se autentica mediante el uso. MFA

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": { "AWS": "arn:aws:iam::123456789012:user/anika" },
      "Action": "sts:AssumeRole",
      "Condition": { "Bool": { "aws:multipactorAuthPresent": true } }
    }
  ]
}
```

A continuación, añada una línea al perfil del rol que especifique el ARN MFA dispositivo del usuario. El siguiente archivo `config` de ejemplo muestra dos perfiles de rol que utilizan las claves de acceso del usuario `anika` para solicitar credenciales temporales para el rol `cli-role`. El usuario `anika` tiene permisos para asumir el rol, concedidos por la política de confianza del rol.

```
[profile role-without-mfa]
region = us-west-2
role_arn= arn:aws:iam::128716708097:role/cli-role
source_profile=cli-user

[profile role-with-mfa]
region = us-west-2
```

```
role_arn= arn:aws:iam::128716708097:role/cli-role
source_profile = cli-user
mfa_serial = arn:aws:iam::128716708097:mfa/cli-user

[profile cli-user]
region = us-west-2
output = json
```

La `mfa_serial` configuración puede tomar un número de serieARN, como se muestra, de un MFA token de hardware.

El primer perfil `role-without-mfa`, no requiereMFA. Sin embargo, como lo exige el ejemplo anterior de política de confianza adjunta a la funciónMFA, cualquier intento de ejecutar un comando con este perfil fallará.

```
$ aws iam list-users --profile role-without-mfa
```

```
An error occurred (AccessDenied) when calling the AssumeRole operation: Access denied
```

La segunda entrada del perfil, `role-with-mfa`, identifica el MFA dispositivo que se va a utilizar. Cuando el usuario intenta ejecutar un AWS CLI comando con este perfil, AWS CLI le pide que introduzca la contraseña de un solo uso (OTP) que proporciona el MFA dispositivo. Si la MFA autenticación se realiza correctamente, el comando realiza la operación solicitada. No OTP se muestra en la pantalla.

```
$ aws iam list-users --profile role-with-mfa
Enter MFA code for arn:aws:iam::123456789012:mfa/cli-user:
{
  "Users": [
    {
      ...
```

## Roles entre cuentas e ID externo

Puede permitir que los usuarios de utilicen roles que pertenezcan a diferentes cuentas si configura el rol como un rol entre cuentas. Durante la creación del rol, defina el tipo de rol en Otra AWS cuenta, tal y como se describe en [Crear un rol para delegar permisos a un IAM usuario](#). Si lo desea, seleccione Requerir MFA. Require MFA configura la condición adecuada en la relación de confianza, tal y como se describe en [Uso de la autenticación multifactor](#).

Si utiliza un [ID externo](#) para ofrecer mayor control sobre quién puede utilizar un rol entre cuentas, también debe añadir el parámetro `external_id` al perfil del rol. Esto se utiliza normalmente solo cuando la otra cuenta la controla alguien ajeno a su empresa u organización.

```
[profile crossaccountrole]
role_arn = arn:aws:iam::234567890123:role/SomeRole
source_profile = default
mfa_serial = arn:aws:iam::123456789012:mfa/saanvi
external_id = 123456
```

## Especificación de un nombre de sesión de rol para simplificar las auditorías

Cuando muchas personas comparten un rol, la auditoría se convierte en un desafío más. Deseará asociar cada operación invocada con la persona que invocó la acción. Sin embargo, cuando la persona utiliza un rol, la asunción del rol por parte de la persona es una acción independiente de la invocación de una operación y debe relacionar manualmente las dos.

Puede simplificar este proceso especificando nombres de sesión de rol únicos cuando los usuarios asumen un rol. Para ello, añada un parámetro `role_session_name` a cada perfil con nombre en el archivo config que especifica un rol. El `role_session_name` valor se transfiere a la `AssumeRole` operación y pasa a formar parte de ARN la sesión del rol. También se incluye en los AWS CloudTrail registros de todas las operaciones registradas.

Por ejemplo, podría crear un perfil basado en roles de la siguiente manera.

```
[profile namedsessionrole]
role_arn = arn:aws:iam::234567890123:role/SomeRole
source_profile = default
role_session_name = Session_Maria_Garcia
```

Esto hace que la sesión de rol tenga lo siguienteARN.

```
arn:aws:iam::234567890123:assumed-role/SomeRole/Session_Maria_Garcia
```

Además, todos los AWS CloudTrail registros incluyen el nombre de la sesión del rol en la información capturada para cada operación.

## Adopción de un rol con una identidad web

Puede configurar un perfil para indicar que AWS CLI debe asumir un rol mediante la [federación de identidades web y Open ID Connect \(OIDC\)](#). Cuando lo especificas en un perfil, AWS CLI automáticamente realiza la AWS STS `AssumeRoleWithWebIdentity` llamada correspondiente por ti.

### Note

Al especificar un perfil que utiliza un IAM rol, AWS CLI realiza las llamadas adecuadas para recuperar las credenciales temporales. Estas credenciales se guardan en `~/.aws/cli/cache`. AWS CLI Los comandos posteriores que especifican el mismo perfil utilizan las credenciales temporales almacenadas en caché hasta que caduquen. En ese momento, actualiza AWS CLI automáticamente las credenciales.

Para recuperar y utilizar credenciales temporales utilizando identidades federadas web, puede especificar los siguientes valores de configuración en un perfil compartido:

### [role\\_arn](#)

Especifica el ARN rol que se va a asumir.

### `web_identity_token_file`

Especifica la ruta a un archivo que contiene un token de acceso OAuth 2.0 o un token de ID de OpenID Connect proporcionado por el proveedor de identidad. La AWS CLI carga este archivo y pasa su contenido como el argumento `WebIdentityToken` de la operación `AssumeRoleWithWebIdentity`.

### [role\\_session\\_name](#)

Especifica un nombre opcional que se aplica a esta sesión de asunción de rol.

A continuación, se muestra un ejemplo de la configuración mínima necesaria para configurar un rol de adopción con un perfil de identidad web:

```
# In ~/.aws/config

[profile web-identity]
role_arn=arn:aws:iam:123456789012:role/RoLeNameToAssume
```

```
web_identity_token_file=/path/to/a/token
```

También puede establecer esta configuración utilizando [variables de entorno](#):

**AWS\_ROLE\_ARN**

El ARN rol que se va a asumir.

**AWS\_WEB\_IDENTITY\_TOKEN\_FILE**

La ruta al archivo de token de identidad web.

**AWS\_ROLE\_SESSION\_NAME**

El nombre aplicado a esta sesión de asunción de rol.

#### Note

Actualmente, estas variables de entorno solo se aplican al rol que se va a adoptar con un proveedor de identidades web. No se aplican a la configuración general del proveedor de roles de adopción.

## Borrado de las credenciales almacenadas en memoria caché

Cuando se utiliza un rol, se almacenan en AWS CLI caché las credenciales temporales de forma local hasta que caduquen. La próxima vez que intente utilizarlas, AWS CLI intentará renovarlas en su nombre.

Si las credenciales temporales del rol se [revocan](#), no se renuevan automáticamente y se producirá un error al intentar utilizarlas. Sin embargo, puede eliminar la memoria caché para forzar la AWS CLI recuperación de las nuevas credenciales.

Linux o macOS

```
$ rm -r ~/.aws/cli/cache
```

Windows

```
C:\> del /s /q %UserProfile%\aws\cli\cache
```

# Autenticarse con credenciales IAM de usuario

## Warning

Para evitar riesgos de seguridad, no utilices a IAM los usuarios para autenticarte cuando desarrolles software específico o trabajos con datos reales. En cambio, utilice la federación con un proveedor de identidades como [AWS IAM Identity Center](#).

En esta sección se explica cómo configurar los ajustes básicos con un IAM usuario. Entre ellas se incluyen sus credenciales de seguridad con los archivos `config` y `credentials`.

## Temas

- [Paso 1: Crea tu IAM usuario](#)
- [Paso 2: Obtener las claves de acceso](#)
- [Configura el AWS CLI](#)
  - [Uso de aws configure](#)

## Paso 1: Crea tu IAM usuario

Cree su IAM usuario siguiendo el procedimiento de [creación de IAM usuarios \(consola\)](#) de la Guía del IAM usuario.

- Para Opciones de permiso, elija Adjuntar políticas directamente para indicar cómo desea asignar permisos a este usuario.
- La mayoría de SDK los tutoriales de introducción utilizan el servicio Amazon S3 como ejemplo. Para proporcionar a su aplicación acceso completo a Amazon S3, seleccione la política `AmazonS3FullAccess` que desea asociar a este usuario.

## Paso 2: Obtener las claves de acceso

1. Inicie sesión en AWS Management Console y abra la IAM consola en <https://console.aws.amazon.com/iam/>.
2. En el panel de navegación de la IAM consola, seleccione Usuarios y, a continuación, seleccione el **User** `nameusuario` que creó anteriormente.

3. En la página del usuario, selecciona la página Credenciales de seguridad. A continuación, en Claves de acceso, seleccione Crear clave de acceso.
4. En el paso 1 de Crear la clave de acceso, elija Interfaz de línea de comandos (CLI).
5. En el paso 2 de Crear clave de acceso, introduzca una etiqueta opcional y seleccione Siguiente.
6. En el paso 3 de Crear la clave de acceso, selecciona Descargar el archivo.csv para guardar un .csv archivo con la clave de acceso y la clave de acceso secreta del IAM usuario. Necesitará esta información más tarde.
7. Seleccione Done (Listo).

## Configura el AWS CLI

Para uso general, AWS CLI necesita la siguiente información:

- ID de clave de acceso
- Clave de acceso secreta
- AWS Región
- Formato de salida

AWS CLI Almacena esta información en un perfil (una colección de ajustes) con default el nombre del `credentials` archivo. De forma predeterminada, la información de este perfil se utiliza cuando se ejecuta un AWS CLI comando que no especifica explícitamente el perfil que se va a utilizar. Para obtener más información sobre el archivo `credentials`, consulte [Opciones de los archivos de configuración y credenciales](#).

Para configurarlo AWS CLI, utilice uno de los siguientes procedimientos:

Temas

- [Uso de aws configure](#)

## Uso de **aws configure**

Para uso general, el `aws configure` comando es la forma más rápida de configurar AWS CLI la instalación. Este asistente de configuración le solicitará cada uno de los datos que necesita para empezar. A menos que se especifique lo contrario mediante la `--profile` opción, AWS CLI almacena esta información en el default perfil.

En el siguiente ejemplo, se configura un perfil de default con valores de ejemplo. Reemplácelos con sus propios valores, tal y como se describe en las siguientes secciones.

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

En el siguiente ejemplo, se configura un perfil denominado userprod con valores de ejemplo. Reemplácelos con sus propios valores, tal y como se describe en las siguientes secciones.

```
$ aws configure --profile userprod
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

## Usa metadatos de EC2 instancias de Amazon para AWS CLI las credenciales

Al ejecutar AWS CLI desde una instancia de Amazon Elastic Compute Cloud (AmazonEC2), puedes simplificar el suministro de credenciales a tus comandos. Cada EC2 instancia de Amazon contiene metadatos que AWS CLI pueden consultar directamente para obtener credenciales temporales. Cuando se IAM asocia un rol a la instancia, recupera las credenciales de los metadatos de la instancia de forma AWS CLI automática y segura.

Para deshabilitar este servicio, usa la variable de DISABLED entorno [AWSEC2\\_METADATA\\_](#).

Temas

- [Requisitos previos](#)
- [Configuración de un perfil para los EC2 metadatos de Amazon](#)

## Requisitos previos

Para usar EC2 las credenciales de Amazon con AWS CLI, debes completar lo siguiente:



- Instalar y configurar la AWS CLI. Para obtener más información, consulte [Instalar la AWS CLI](#) y [Credenciales de autenticación y acceso](#).
- Comprende los archivos de configuración y los perfiles con nombre. Para obtener más información, consulte [Opciones de los archivos de configuración y credenciales](#).
- Cree un rol AWS Identity and Access Management (IAM) que tiene acceso a los recursos necesarios y lo asociaste a la EC2 instancia de Amazon cuando lo lanzaste. Para obtener más información, consulte [IAM las políticas de Amazon EC2](#) en la Guía del EC2 usuario de Amazon y la [concesión de acceso a los AWS recursos a las aplicaciones que se ejecutan en EC2 instancias de Amazon](#) en la Guía del IAM usuario.

## Configuración de un perfil para los EC2 metadatos de Amazon

Para especificar que quieres usar las credenciales disponibles en el perfil de EC2 instancia de Amazon de alojamiento, usa la siguiente sintaxis en el perfil indicado en tu archivo de configuración. Para obtener instrucciones adicionales, consulte estos pasos a continuación.

```
[profile profilename]  
role_arn = arn:aws:iam::123456789012:role/rolename  
credential_source = Ec2InstanceMetadata  
region = region
```

1. Cree un perfil en su archivo de configuración.

```
[profile profilename]
```

2. Añada su rol de IAM arn que tenga acceso a los recursos necesarios.

```
role_arn = arn:aws:iam::123456789012:role/rolename
```

3. Especifique Ec2InstanceMetadata como fuente de credenciales.

```
credential_source = Ec2InstanceMetadata
```

4. Establezca su región.

```
region = region
```

## Ejemplo

En el siguiente ejemplo se supone que *función de administrador de marketing* rol y usa la *us-west-2* región en un perfil de EC2 instancia de Amazon denominado *marketingadmin*.

```
[profile marketingadmin]
role_arn = arn:aws:iam::123456789012:role/marketingadminrole
credential_source = Ec2InstanceMetadata
region = us-west-2
```

## Obtener credenciales con un proceso externo

### Warning

En el siguiente tema, se explica cómo se obtienen credenciales desde un proceso externo. Esto puede suponer un riesgo para la seguridad si el comando que genera las credenciales llega a estar accesible para usuarios o procesos no aprobados. Le recomendamos que utilice las alternativas seguras y compatibles que ofrece AWS CLI y AWS para reducir el riesgo de comprometer sus credenciales. Asegúrese de proteger el archivo `config` y todos los archivos auxiliares y herramientas para evitar su divulgación.

Asegúrese de que su herramienta de credenciales personalizada no escriba ninguna información secreta en ella, `StdErr` ya que AWS CLI puede capturarla y registrarla, lo que podría exponerla a usuarios no autorizados. SDKs

Si dispone de un método para generar o buscar credenciales que no sea compatible directamente con el AWS CLI, puede configurarlo AWS CLI para que lo utilice configurando los `credential_process` parámetros del `config` archivo.

Por ejemplo, puede incluir una entrada similar a la siguiente en el archivo `config`:

```
[profile developer]
credential_process = /opt/bin/awscreds-custom --username helen
```

## Sintaxis

Para crear esta cadena de forma que sea compatible con cualquier sistema operativo, siga estas reglas:

- Si la ruta de acceso o el nombre del archivo contienen un espacio, rodee la ruta completa y el nombre del archivo con comillas dobles (" "). La ruta y el nombre de archivo solo pueden constar de los caracteres: A-Z a-z 0-9 - \_ . espacio
- Si un nombre de parámetro o un valor de parámetro contienen un espacio, rodee ese elemento con comillas dobles (" "). Incluya solo el nombre o el valor, no el par.
- No incluya ninguna variable de entorno en las cadenas. Por ejemplo, no puede incluir \$HOME ni %USERPROFILE%.
- No especifique la carpeta de inicio como ~. Debe especificar la ruta completa.

### Ejemplo para Windows

```
credential_process = "C:\Path\To\credentials.cmd" parameterWithoutSpaces "parameter with spaces"
```

### Ejemplo para Linux o macOS

```
credential_process = "/Users/Dave/path/to/credentials.sh" parameterWithoutSpaces "parameter with spaces"
```

### Salida prevista del programa Credentials


AWS CLI Ejecuta el comando tal y como se especifica en el perfil y, a continuación, lee los datosSTDOUT. El comando que especifique debe generar un JSON resultado STDOUT que coincida con la siguiente sintaxis.

```
{
  "Version": 1,
  "AccessKeyId": "an AWS access key",
  "SecretAccessKey": "your AWS secret access key",
  "SessionToken": "the AWS session token for temporary credentials",
  "Expiration": "ISO8601 timestamp when the credentials expire"
}
```

#### Note

En la fecha de publicación del presente documento, la clave `Version` debe establecerse en 1. Puede aumentar con el paso del tiempo a medida que la estructura evolucione.

La `Expiration` clave es una marca de tiempo con formato [ISO8601](#). Si la `Expiration` clave no está presente en la salida de la herramienta, se CLI supone que las credenciales son credenciales de larga duración que no se actualizan. De otro modo, las credenciales se consideran credenciales temporales y se actualizan automáticamente volviendo a ejecutar el comando `credential_process` antes de que caduquen.

 Note

AWS CLI No almacena en caché las credenciales de los procesos externos de la misma forma en que lo hace con las credenciales de los roles. Si se requiere el almacenamiento en caché, debe implementarlo en el proceso externo.

El proceso externo puede devolver un código de devolución distinto de cero para indicar que se ha producido un error al intentar recuperar las credenciales.

# Usa el AWS CLI

En esta sección se proporciona una descripción general completa del uso general, las funciones comunes y las opciones disponibles en AWS Command Line Interface (AWS CLI), además de los detalles que se describen en la [the section called “puntos de conexión”](#) sección de configuración.

Esta guía profundiza en los aspectos fundamentales de la escritura de AWS CLI comandos, incluida su estructura básica, su formato y sus capacidades de filtrado. Al comprender estos elementos básicos, podrá crear comandos que se dirijan con precisión a los recursos y las acciones que necesita, sin necesidad de navegar por complejas consolas basadas en la web.

Además, aquí se destacan el contenido de ayuda y la documentación disponibles para el AWS CLI. Desde la ayuda de línea de comandos integrada hasta la completa [AWS CLI guía](#) de , tendrá acceso a información que le ayudará a explorar las características y capacidades del AWS CLI.

Para ver ejemplos y casos de uso Servicio de AWS específicos, consulte [Ejemplos de código](#) la [guía de AWS CLI referencia](#) de la . Estos proporcionan información específica sobre los comandos y muestran ejemplos sobre cómo aprovecharlos AWS CLI para varios Servicios de AWS.

## Note

De forma predeterminada, AWS CLI envía las solicitudes Servicios de AWS a través HTTPS del TCP puerto 443. Para garantizar un uso correcto del AWS CLI, debe poder realizar conexiones salientes en este puerto.

## Temas de esta guía

- [Acceda a la ayuda para el AWS Command Line Interface](#)
- [Estructura de mando en el AWS CLI](#)
- [Especifique los valores de los parámetros para AWS CLI](#)
- [Salida del comando de control desde el AWS CLI](#)
- [Códigos de devolución de la AWS CLI](#)
- [Crear y usar atajos de AWS CLI comandos llamados alias](#)

# Acceda a la ayuda para el AWS Command Line Interface

En este tema se describe cómo acceder al contenido de ayuda de AWS Command Line Interface (AWS CLI).

## Temas

- [El comando de ayuda de AWS CLI integrado](#)
- [AWS CLI guía de referencia](#)
- [API documentación](#)
- [Solución de errores](#)
- [Ayuda adicional](#)

## El comando de ayuda de AWS CLI integrado

Puede obtener ayuda con cualquier comando cuando utilice AWS Command Line Interface (AWS CLI). Para ello, simplemente escriba `help` al final del nombre de un comando.

Por ejemplo, el siguiente comando muestra la ayuda sobre las AWS CLI opciones generales y los comandos de nivel superior disponibles.

```
$ aws help
```

El siguiente comando muestra los comandos específicos de Amazon Elastic Compute Cloud (AmazonEC2) disponibles.

```
$ aws ec2 help
```

En el siguiente ejemplo, se muestra una ayuda detallada para la EC2 `DescribeInstances` operación de Amazon. La ayuda incluye descripciones de los parámetros de entrada, los filtros disponibles y qué se incluye como salida. También incluye ejemplos que muestran cómo escribir variaciones comunes del comando.

```
$ aws ec2 describe-instances help
```

La ayuda de cada comando se divide en seis secciones:

## Nombre

El nombre del comando.

```
NAME
    describe-instances -
```

## Descripción

Descripción de la API operación que invoca el comando.

```
DESCRIPTION
    Describes one or more of your instances.

    If you specify one or more instance IDs, Amazon EC2 returns information
    for those instances. If you do not specify instance IDs, Amazon EC2
    returns information for all relevant instances. If you specify an
    instance ID that is not valid, an error is returned. If you specify an
    instance that you do not own, it is not included in the returned
    results.

    ...
```

## Sinopsis

La sintaxis básica para utilizar el comando y sus opciones. Si una opción aparece entre corchetes, significa que es opcional, que tiene un valor predeterminado o que existe una opción alternativa que puede utilizarse en su lugar.

```
SYNOPSIS
    describe-instances
    [--dry-run | --no-dry-run]
    [--instance-ids <value>]
    [--filters <value>]
    [--cli-input-json <value>]
    [--starting-token <value>]
    [--page-size <value>]
    [--max-items <value>]
    [--generate-cli-skeleton]
```

Por ejemplo, `describe-instances` tiene un comportamiento predeterminado que describe todas las instancias de la cuenta corriente y de la AWS región. Si lo desea, puede especificar

una lista de `instance-ids` para describir una o varias instancias. `dry-run` es un indicador booleano opcional que no adopta ningún valor. Para utilizar un indicador booleano, especifique el valor aparece; en este caso, `--dry-run` o `--no-dry-run`. Del mismo modo, `--generate-cli-skeleton` no adopta un valor. Si existen condiciones para usar una opción, aparecen descritas en la sección `OPTIONS` o se muestran en los ejemplos.

## Opciones

Una descripción de cada una de las opciones que aparecen en el resumen.

### OPTIONS

`--dry-run | --no-dry-run` (boolean)

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

`--instance-ids` (list)

One or more instance IDs.

Default: Describes all your instances.

...

## Ejemplos

Ejemplos que ilustran el uso del comando y sus opciones. Si no hay ningún ejemplo disponible para un comando o caso de uso que necesite, solicite uno mediante el enlace de comentarios de esta página o en la referencia de AWS CLI comandos de la página de ayuda del comando.

### EXAMPLES

#### To describe an Amazon EC2 instance

Command:

```
aws ec2 describe-instances --instance-ids i-5203422c
```

#### To describe all instances with the instance type `m1.small`

Command:

```
aws ec2 describe-instances --filters "Name=instance-type,Values=m1.small"
```

#### To describe all instances with an Owner tag



Command:

```
aws ec2 describe-instances --filters "Name=tag-key,Values=Owner"
```

...

## Salida

Descripciones de cada uno de los campos y tipos de datos incluidos en la respuesta de AWS.

Para `describe-instances`, la salida es una lista de objetos de reserva, cada uno de los cuales contiene varios campos y objetos con información acerca de las instancias asociadas al mismo. Esta información proviene de la [API documentación del tipo de datos de reserva](#) utilizado por AmazonEC2.

### OUTPUT

Reservations -> (list)

One or more reservations.

(structure)

Describes a reservation.

ReservationId -> (string)

The ID of the reservation.

OwnerId -> (string)

The ID of the AWS account that owns the reservation.

RequesterId -> (string)

The ID of the requester that launched the instances on your behalf (for example, AWS Management Console or Auto Scaling).

Groups -> (list)

One or more security groups.

(structure)

Describes a security group.

GroupName -> (string)

The name of the security group.

GroupId -> (string)

The ID of the security group.

```
Instances -> (list)
  One or more instances.

(structure)
  Describes an instance.

  InstanceId -> (string)
    The ID of the instance.

  ImageId -> (string)
    The ID of the AMI used to launch the instance.

  State -> (structure)
    The current state of the instance.

  Code -> (integer)
    The low byte represents the state. The high byte
    is an opaque internal value and should be ignored.

...
```

Cuando AWS CLI renderiza el resultadoJSON, se convierte en una matriz de objetos de reserva, similar al ejemplo siguiente.

```
{
  "Reservations": [
    {
      "OwnerId": "012345678901",
      "ReservationId": "r-4c58f8a0",
      "Groups": [],
      "RequesterId": "012345678901",
      "Instances": [
        {
          "Monitoring": {
            "State": "disabled"
          },
          "PublicDnsName": "ec2-52-74-16-12.us-
west-2.compute.amazonaws.com",
          "State": {
            "Code": 16,
            "Name": "running"
          }
        },
        ...
      ]
    }
  ]
}
```

Cada objeto de reserva tiene campos que describen la reserva y una matriz de objetos de instancia, cada uno de ellos con sus propios campos (por ejemplo, `PublicDnsName`) y objetos (por ejemplo `State`) que los describen.

### Usuarios de Windows

Puede añadir una barra vertical (`|`) a la salida del comando de ayuda al comando `more` para ver las páginas del archivo de ayuda una a una. Pulse la barra espaciadora o `PgDn` para ver más del documento y `q` salir.

```
C:\> aws ec2 describe-instances help | more
```

## AWS CLI guía de referencia

Los archivos de ayuda contienen enlaces que no se pueden ver o visitar desde la línea de comandos. Puede ver estos enlaces e interactuar con ellos utilizando la guía de [referencia en línea de la AWS CLI versión 1, la guía](#) de . La referencia también contiene el contenido de ayuda de todos los AWS CLI comandos. Las descripciones se presentan para facilitar la navegación y la consulta en móviles, tabletas o equipos de escritorio.

## API documentación

Todos los comandos del mismo AWS CLI corresponden a las solicitudes realizadas al público de un AWS servicio API. Cada servicio público API tiene una API referencia que se encuentra en la página de inicio del servicio, en el sitio [web de AWS documentación](#). El contenido de una API referencia varía según la forma en que API se construya y el protocolo que se utilice. Por lo general, una API referencia contiene información detallada sobre las operaciones que admite el servicio API, los datos que se envían y reciben del servicio y cualquier condición de error que el servicio pueda informar.

### API Secciones de documentación

- **Acciones:** información detallada de cada operación y sus parámetros (incluidas las restricciones de longitud o contenido, y valores predeterminados). Muestra los errores que pueden ocurrir con esta operación. Cada operación corresponde a un subcomando de AWS CLI.
- **Tipos de datos:** información detallada sobre las estructuras que un comando puede requerir como parámetro, o devolver en respuesta a una solicitud.

- **Parámetros comunes:** información detallada sobre los parámetros que utilizan todas las acciones de un servicio.
- **Errores comunes:** información detallada sobre los errores que cualquiera de las operaciones del servicio puede devolver.

El nombre y la disponibilidad de cada sección puede variar en función del servicio.

#### Servicio específico CLIs

Algunos servicios tienen una versión independiente CLI que data de antes de que AWS CLI se creara un solo servicio para funcionar con todos los servicios. Estos servicios específicos CLIs tienen documentación independiente que está vinculada desde la página de documentación del servicio. La documentación relativa a un servicio específico CLIs no se aplica al. AWS CLI

## Solución de errores

Si necesita ayuda para diagnosticar y corregir AWS CLI errores, consulte. [Errores de solución de problemas](#)

## Ayuda adicional

Para obtener ayuda adicional con tus AWS CLI problemas, visita la [AWS CLI comunidad](#) en GitHub.

## Estructura de mando en el AWS CLI

En este tema se explica cómo se estructura el comando AWS Command Line Interface (AWS CLI) y cómo utilizar los comandos de espera.

### Temas

- [Estructura de comandos](#)
- [Comandos de espera](#)

## Estructura de comandos

AWS CLI Utiliza una estructura de varias partes en la línea de comandos que debe especificarse en este orden:

1. La llamada base al programa de aws.
2. El comando de nivel superior, que normalmente corresponde a un AWS servicio compatible con AWS CLI
3. El subcomando que especifica qué operación realizar.
4. AWS CLI Opciones o parámetros generales requeridos por la operación. Puede especificarlos en cualquier orden siempre que vayan detrás de las tres primeras partes. Si se especifica un parámetro exclusivo varias veces, solo se aplica el último valor.

```
$ aws <command> <subcommand> [options and parameters]
```

Los parámetros pueden tomar varios tipos de valores de entrada, como números, cadenas, listas, mapas y JSON estructuras. Los valores admitidos dependen del comando y los subcomandos que especifique.

### Ejemplos

#### Amazon S3

En el siguiente ejemplo se enumeran todos los buckets de Amazon S3.

```
$ aws s3 ls
2018-12-11 17:08:50 my-bucket
2018-12-14 14:55:44 my-bucket2
```

Para obtener más información acerca de los comandos de Simple Storage Service (Amazon S3), consulte [aws s3](#) en la Referencia de los comandos de AWS CLI .

#### AWS CloudFormation

El siguiente ejemplo de [create-change-set](#) comando cambia el nombre de la pila de formación de nubes a *my-change-set*.

```
$ aws cloudformation create-change-set --stack-name my-stack --change-set-name my-change-set
```

Para obtener más información sobre los AWS CloudFormation comandos, consulte [aws cloudformation](#) la Referencia de AWS CLI comandos.

## Comandos de espera

Algunos AWS servicios tienen `wait` comandos disponibles. Cualquier comando que use `aws wait` por lo general espera hasta que se completa un comando antes de ir al siguiente paso. Esto es especialmente útil para comandos multiparte o scripting, ya que puede utilizar un comando de espera para evitar pasar a pasos posteriores si el comando de espera falla.

AWS CLI Utiliza una estructura de varias partes en la línea de comandos para el `wait` comando que debe especificarse en este orden:

1. La llamada base al programa de `aws`.
2. El comando de nivel superior, que normalmente corresponde a un AWS servicio compatible con. AWS CLI
3. El comando `wait`
4. El subcomando que especifica qué operación realizar.
5. CLI Opciones o parámetros generales requeridos por la operación. Puede especificarlos en cualquier orden siempre que vayan detrás de las tres primeras partes. Si se especifica un parámetro exclusivo varias veces, solo se aplica el último valor.

```
$ aws <command> wait <subcommand> [options and parameters]
```

Los parámetros pueden tomar varios tipos de valores de entrada, como números, cadenas, listas, mapas y JSON estructuras. Los valores admitidos dependen del comando y los subcomandos que especifique.

### Note

No todos los AWS servicios admiten `wait` comandos. Consulte la [guía de AWS CLI referencia](#) de la para comprobar si su servicio admite `wait` comandos.

## Ejemplos

### AWS CloudFormation

El siguiente ejemplo de [wait change-set-create-complete](#) comandos se detiene y se reanuda solo después de confirmar que el *my-change-set* cambio establecido en el *my-stack* la pila está lista para funcionar.

```
$ aws cloudformation wait change-set-create-complete --stack-name my-stack --change-set-name my-change-set
```

Para obtener más información acerca de los comandos de AWS CloudFormation `wait`, consulte [wait](#) en la Referencia de los comandos de AWS CLI .

## AWS CodeDeploy

El siguiente ejemplo de [wait deployment-successful](#) comandos hace una pausa hasta que *d-A1B2C3111* el despliegue se completa correctamente.

```
$ aws deploy wait deployment-successful --deployment-id d-A1B2C3111
```

Para obtener más información acerca de los comandos de AWS CodeDeploy `wait`, consulte [wait](#) en la Referencia de los comandos de AWS CLI .

## Especifique los valores de los parámetros para AWS CLI

Muchos de los parámetros utilizados en AWS Command Line Interface (AWS CLI) son cadenas simples o valores numéricos, como el nombre del par de claves *my-key-pair* en el siguiente ejemplo de `aws ec2 create-key-pair` comando.

```
$ aws ec2 create-key-pair --key-name my-key-pair
```

El formato del comando puede variar de un terminal a otro. Por ejemplo, la mayoría de los terminales distinguen entre mayúsculas y minúsculas, pero Powershell no distingue entre mayúsculas y minúsculas. Esto significa que los dos ejemplos de comandos siguientes arrojarían resultados diferentes para los terminales que distinguen entre mayúsculas y minúsculas, ya que ven *MyFile\*.txt* y *myfile\*.txt* como parámetros diferentes.

Sin embargo, PowerShell procesaría estas solicitudes de la misma manera que ve *MyFile\*.txt* y con los *myfile\*.txt* mismos parámetros. El siguiente ejemplo de comando muestra estos parámetros mediante el `aws s3 cp` comando:

```
$ aws s3 cp . s3://my-bucket/path --include "MyFile*.txt"
```

```
$ aws s3 cp . s3://my-bucket/path --include "myfile*.txt"
```

Para obtener más información sobre la PowerShell insensibilidad entre mayúsculas y minúsculas, consulte [about\\_CASE-Sensitivity](#) en la documentación. PowerShell

A veces es necesario utilizar comillas o literales en las cadenas que incluyen caracteres especiales o de espacio. Las reglas en torno a este formato también pueden variar entre terminales. Para obtener más información sobre el uso de las comillas alrededor de parámetros complejos, consulte [Comillas y literales con cadenas en AWS CLI](#).

En estos temas se tratan las reglas de formato de terminal más comunes. Si tiene problemas para que su terminal reconozca los valores de sus parámetros, asegúrese de revisar los temas de esta sección y también de consultar la documentación de su terminal para conocer sus reglas de sintaxis específicas.

Temas de parámetros

- [Tipos de parámetros comunes de la AWS CLI](#)
- [Comillas y literales con cadenas en AWS CLI](#)
- [Cargar parámetros de la AWS CLI desde un archivo](#)
- [Esqueletos y archivos de entrada de AWS CLI](#)
- [Usar sintaxis abreviada con la AWS CLI](#)

## Tipos de parámetros comunes de la AWS CLI

En esta sección, se describen algunos de los tipos de parámetros comunes y el formato requerido típico.

Si tiene problemas con el formato de un parámetro para un comando específico, compruebe la ayuda escribiendo **help** después del nombre del comando. La ayuda de cada subcomando incluye el nombre y la descripción de una opción. El tipo de parámetro de la opción se muestra entre paréntesis. Para obtener más información acerca de obtener ayuda, consulte [the section called “Obtención de ayuda”](#).

Los tipos de parámetros incluyen:

- [Cadena](#)
- [Timestamp](#)
- [Enumeración](#)



- [Booleano](#)
- [Entero](#)
- [Binario/blob \(objeto binario grande\) y blob en streaming](#)
- [Mapeo](#)
- [Documento](#)

## Cadena

Los parámetros de cadena pueden contener caracteres alfanuméricos, símbolos y espacios en blanco del conjunto de caracteres [ASCII](#). Las cadenas que contienen espacios en blanco se deben entrecomillar. Le recomendamos que no utilice símbolos ni espacios en blanco diferentes del carácter de espacio estándar, y que observe las [reglas de entrecomillado](#) del terminal para evitar resultados inesperados.

Algunos parámetros de cadena pueden aceptar datos binarios de un archivo. Consulte [Archivos binarios](#) para ver un ejemplo.

## Timestamp

Las marcas temporales tienen un formato conforme a la norma [ISO 8601](#). A menudo se conocen como parámetros "DateTime" o "Date".

```
$ aws ec2 describe-spot-price-history --start-time 2014-10-13T19:00:00Z
```

Los formatos aceptados son:

- *AAAA-MM-DDThh:mm:ss.sssTZD (UTC)*, por ejemplo, 2014-10-01T20:30:00.000Z
- *AAAA-MM-DDThh:mm:ss.sssTZD (con compensación)*, por ejemplo, 2014-10-01T12:30:00.000-08:00
- *AAAA:MM:DD*, por ejemplo, 2014-10-01
- Hora Unix en segundos; por ejemplo, 1412195400. Esto es lo que a veces se denomina [fecha de inicio UNIX](#) y representa el número de segundos desde la medianoche del 1 de enero de 1970 UTC.

Puede establecer el formato de la marca de tiempo con el ajuste de archivo [cli\\_timestamp\\_format](#).

## Enumeración

Una o varias cadenas separadas por espacios. Si cualquiera de los elementos de cadena contiene un espacio, debe escribir dicho elemento entre comillas. Observe las [normas de entrecomillado](#) del terminal para evitar resultados inesperados

```
$ aws ec2 describe-spot-price-history --instance-types m1.xlarge m1.medium
```

## Booleano

Marcador binario que activa o desactiva una opción. Por ejemplo, `ec2 describe-spot-price-history` tiene con un parámetro `--dry-run` booleano que, cuando se especifica, valida la consulta con el servicio sin ejecutar realmente una consulta.

```
$ aws ec2 describe-spot-price-history --dry-run
```

El resultado indica si el comando tenía el formato correcto o no. Este comando también incluye una versión `--no-dry-run` del parámetro que puede utilizarse para indicar de forma explícita que el comando debe ejecutarse con normalidad, aunque no es necesario incluirlo, ya que este es el comportamiento predeterminado.

## Entero

Un número entero sin firma.

```
$ aws ec2 describe-spot-price-history --max-items 5
```

## Binario/blob (objeto binario grande) y blob en streaming

En la AWS CLI, puede pasar un valor binario como una cadena directamente en la línea de comandos. Existen dos tipos de blobs:

- [Blob](#)
- [Blob de streaming](#)

### Blob

Para pasar un valor a un parámetro con el tipo `blob`, debe especificar una ruta a un archivo local que contenga los datos binarios mediante el prefijo `fileb://`. Los archivos a los que se hace

referencia mediante el prefijo `fileb://` siempre se tratan como binarios sin codificar. La ruta especificada se interpreta como una ruta relativa al directorio de trabajo actual. Por ejemplo, el parámetro `--plaintext` para `aws kms encrypt` es un blob.

```
$ aws kms encrypt \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --plaintext fileb://ExamplePlaintextFile \  
  --output text \  
  --query CiphertextBlob | base64 \  
  --decode > ExampleEncryptedFile
```

## Blob de streaming

Los blobs de streaming como `aws cloudsearchdomain upload-documents` no utilizan prefijos. En cambio, los parámetros de blob de streaming se formatean mediante la ruta de archivo directa. En el siguiente ejemplo se utiliza la ruta de archivo directa `document-batch.json` para el comando `aws cloudsearchdomain upload-documents`:

```
$ aws cloudsearchdomain upload-documents \  
  --endpoint-url https://doc-my-domain.us-west-1.cloudsearch.amazonaws.com \  
  --content-type application/json \  
  --documents document-batch.json
```

## Mapeo

Un conjunto de pares de clave-valor especificado en JSON o mediante la [sintaxis abreviada](#) de la CLI. En el siguiente ejemplo de JSON, se lee un elemento de una tabla de Amazon DynamoDB denominada `my-table` con un parámetro de mapa, `--key`. El parámetro especifica la clave principal denominada `id` con un valor numérico de 1 en una estructura JSON anidada.

Para conocer un uso más avanzado de JSON en una línea de comandos, considere usar un procesador JSON de línea de comandos, como `jq`, para crear cadenas JSON. Para obtener más información sobre `jq`, consulte el [repositorio jq](#) en GitHub.

```
$ aws dynamodb get-item --table-name my-table --key '{"id": {"N": "1"}}'  
  
{  
  "Item": {  
    "name": {  
      "S": "John"    }  
  }  
}
```

```
    },
    "id": {
      "N": "1"
    }
  }
}
```

## Documento

### Note

La [sintaxis abreviada](#) no es compatible con los tipos de documento.

Los tipos de documento se utilizan para enviar datos sin necesidad de integrar JSON dentro de cadenas. El tipo de documento permite a los servicios proporcionar esquemas arbitrarios para que utilice tipos de datos más flexibles.

Esto permite enviar datos JSON sin necesidad de valores de escape. Por ejemplo, en lugar de utilizar la siguiente entrada JSON de escape:

```
{"document": "{\"key\":true}"}
```

Puede utilizar el tipo de documento que se muestra a continuación:

```
{"document": {"key": true}}
```

## Valores válidos para tipos de documentos

Debido a la naturaleza flexible de los tipos de documentos, existen varios tipos de valores válidos. Entre los valores válidos se incluyen los siguientes:

### Cadena

```
--option "value"
```

### Número

```
--option 123  
--option 123.456
```

## Booleano

```
--option true
```

## Null

```
--option null
```

## Array (Matriz)

```
--option '["value1", "value2", "value3"]'  
--option '["value", 1, true, null, ["key1", 2.34], {"key2": "value2"}]'
```

## Objeto

```
--option '{"key": "value}"'  
--option '{"key1": "value1", "key2": 123, "key3": true, "key4": null, "key5":  
["value3", "value4"], "key6": {"value5": "value6"}'}
```

## Comillas y literales con cadenas en AWS CLI

Hay dos maneras principales en las que se usan comillas simples y dobles en el AWS CLI.

- [Usar comillas alrededor de cadenas que contienen espacios en blanco](#)
- [Uso de comillas dentro de cadenas](#)

### Usar comillas alrededor de cadenas que contienen espacios en blanco

Los nombres de los parámetros y sus valores están separados por espacios en la línea de comandos. Si el valor de una cadena contiene un espacio incrustado, debe escribir toda la cadena entre comillas para evitar que se malinterprete el espacio como un separador entre el valor y el nombre del siguiente parámetro. AWS CLI El tipo de comillas que utilice depende del sistema operativo en el que se ejecute. AWS CLI

### Linux and macOS

Uso de comillas simples ' '

```
$ aws ec2 create-key-pair --key-name 'my key pair'
```

Para obtener más información sobre el uso de las comillas, consulte documentación del usuario de su shell preferido.

## PowerShell

### Comillas simples (recomendadas)

Las marcas de comillas simples ' ' se llaman cadenas verbatim. La cadena se pasa al comando exactamente como se escribe, lo que significa que PowerShell las variables no se transferirán.

```
PS C:\> aws ec2 create-key-pair --key-name 'my key pair'
```

### Comillas dobles

Las marcas de comillas dobles " " se llaman cadenas expandible. Las variables se pueden pasar en cadenas expandibles.

```
PS C:\> aws ec2 create-key-pair --key-name "my key pair"
```

Para obtener más información sobre el uso de comillas, consulte [Acerca de las reglas de cotización](#) en Microsoft PowerShell Docs.

## Windows command prompt

Uso de comillas dobles " ".

```
C:\> aws ec2 create-key-pair --key-name "my key pair"
```

Si lo desea, puede separar el nombre del parámetro del valor con un signo de igual = en lugar de un espacio. Normalmente, esto solo es necesario si el valor del parámetro comienza con un guion.

```
$ aws ec2 delete-key-pair --key-name=-mykey
```

## Uso de comillas dentro de cadenas

Las cadenas pueden contener comillas y el shell puede requerir comillas de cierre para que funcionen correctamente. Uno de los tipos de valores de parámetros más comunes es una JSON

cadena. Esto es complejo, ya que incluye espacios y comillas dobles " " alrededor del nombre y el valor de cada elemento de la JSON estructura. La forma de introducir los parámetros JSON con formato en la línea de comandos varía según el sistema operativo.

Para un JSON uso más avanzado de la línea de comandos, considere la posibilidad de utilizar un JSON procesador de línea de comandos, por ejemplo `jq`, para crear JSON cadenas. Para obtener más información `jq`, consulte el [repositorio jq](#) en GitHub.

## Linux and macOS

Para que Linux y macOS interpreten cadenas literalmente, utilice comillas simples ' ' para encerrar la estructura de JSON datos, como en el siguiente ejemplo. No es necesario evitar las comillas dobles incrustadas en la JSON cadena, ya que se tratan literalmente. Como el JSON está encerrado entre comillas simples, será necesario eliminar todas las comillas simples de la cadena, lo que suele hacerse con una barra invertida antes de la comilla simple. \'

```
$ aws ec2 run-instances \  
  --image-id ami-12345678 \  
  --block-device-mappings '[{"DeviceName":"/dev/sdb","Ebs":  
{"VolumeSize":20,"DeleteOnTermination":false,"VolumeType":"standard"}]'
```

Para obtener más información sobre el uso de las comillas, consulte documentación del usuario de su shell preferido.

## PowerShell

Utilice comillas simples ' ' o comillas dobles " ".

### Comillas simples (recomendadas)

Las marcas de comillas simples ' ' se llaman cadenas *verbatim*. La cadena se pasa al comando exactamente como se escribe, lo que significa que PowerShell las variables no se transferirán.

Como las estructuras de JSON datos incluyen comillas dobles, le sugerimos que las cierre entre comillas ' ' simples. Si utiliza comillas simples, no necesita evitar las comillas dobles incrustadas en la JSON cadena. Sin embargo, debe evitar cada comilla simple con una comilla invertida ` dentro de la JSON estructura.

```
PS C:\> aws ec2 run-instances `
```

```
--image-id ami-12345678 `
--block-device-mappings '[{"DeviceName":"/dev/sdb","Ebs":
{"VolumeSize":20,"DeleteOnTermination":false,"VolumeType":"standard"}]`
```

## Comillas dobles

Las marcas de comillas dobles " " se llaman cadenas expandible. Las variables se pueden pasar en cadenas expandibles.

Si utiliza comillas dobles, no necesita evitar las comillas simples incrustadas en la JSON cadena. Sin embargo, debe evitar cada comilla doble con una comilla invertida ` dentro de la JSON estructura, como en el ejemplo siguiente.

```
PS C:\> aws ec2 run-instances `
--image-id ami-12345678 `
--block-device-mappings "[{ `DeviceName `": `"/dev/sdb `", `Ebs `":
{ `VolumeSize `":20, `DeleteOnTermination `":false, `VolumeType `": `standard `"}]`
```

Para obtener más información sobre el uso de comillas, consulte [Acerca de las reglas de cotización](#) en Microsoft PowerShell Docs.

### Warning

Antes PowerShell de enviar un comando a AWS CLI, determina si el comando se interpreta mediante reglas típicas PowerShell o entre CommandLineToArgvW comillas. Cuando PowerShell los procesos se utilizanCommandLineToArgvW, debes evitar los caracteres con una barra \ invertida.

Para obtener más información sobre CommandLineToArgvW in PowerShell, consulte [Qué pasa con el extraño tratamiento de las comillas y las barras invertidas por CommandLineToArgv W](#) en Microsoft DevBlogs, [Todo el mundo cita los argumentos de la línea de comandos de manera incorrecta en el](#) blog de Microsoft Docs y la [función CommandLineToArgv W](#) en Microsoft Docs.

### Comillas simples

Las marcas de comillas simples ' ' se llaman cadenas verbatim. La cadena se pasa al comando exactamente como se escribe, lo que significa que PowerShell las variables no se transferirán. Los caracteres se escapan con una barra invertida\.

```
PS C:\> aws ec2 run-instances `
--image-id ami-12345678 `
```



```
--block-device-mappings '[{"DeviceName":"/dev/sdb","Ebs":
{"VolumeSize":20,"DeleteOnTermination":false,"VolumeType":
"standard"}}]'
```

### Comillas dobles

Las marcas de comillas dobles " " se llaman cadenas expandable. Las variables se pueden pasar en cadenas expandable. En el caso de las cadenas entre comillas dobles, hay que escapar dos veces usando \" para cada cita en lugar de usar solo una comilla invertida. La barra invertida escapa la barra invertida y, a continuación, la barra invertida se utiliza como un carácter de escape para proceso CommandLineToArgvW.

```
PS C:\> aws ec2 run-instances `
  --image-id ami-12345678 `
  --block-device-mappings "[{"DeviceName":"/dev/sdb","Ebs":
{"VolumeSize":20,"DeleteOnTermination":false,"VolumeType":
"standard"}}]"
```

### Blobs (recomendado)

Para evitar las reglas PowerShell de cotización para la entrada de JSON datos, usa Blobs para pasar tus JSON datos directamente a. AWS CLI Para obtener más información acerca de Blobs, consulte o [Blob](#).

## Windows command prompt

La línea de comandos de Windows requiere comillas dobles " " para cerrar la JSON estructura de datos. Además, para evitar que el procesador de comandos malinterprete las comillas dobles incrustadas en laJSON, también debe evitar (preceder de una barra invertida\) todas las comillas dobles de la propia estructura de JSON datos, como " en el ejemplo siguiente.

```
C:\> aws ec2 run-instances ^
  --image-id ami-12345678 ^
  --block-device-mappings [{"DeviceName":"/dev/sdb","Ebs":
{"VolumeSize":20,"DeleteOnTermination":false,"VolumeType":
"standard"}}]"
```

El único caso en el que no se utiliza el carácter de cierre es con las comillas dobles de los extremos.

## Cargar parámetros de la AWS CLI desde un archivo

Algunos parámetros esperan nombres de archivo como argumentos, desde los que la AWS CLI carga los datos. Otros parámetros le permiten especificar el valor del parámetro como texto escrito en la línea de comandos o leído de un archivo. Tanto si un archivo es obligatorio como si es opcional, debe codificarlo correctamente para que la AWS CLI pueda entenderlo. La codificación del archivo debe coincidir con la configuración regional predeterminada del sistema de lectura. Esto se puede determinar utilizando el método Python `locale.getpreferredencoding()`.

### Note

De forma predeterminada, Windows PowerShell genera texto como UTF-16, lo que está en conflicto con la codificación UTF-8 utilizada por archivos JSON y muchos sistemas Linux. Le recomendamos que utilice `-Encoding ascii` con los comandos `Out-File` de PowerShell para asegurarse de que la AWS CLI puede leer el archivo resultante.

### Temas

- [Cómo cargar parámetros de un archivo](#)
- [Archivos binarios](#)
- [Archivos remotos](#)

## Cómo cargar parámetros de un archivo

En ocasiones, resulta conveniente cargar el valor de un parámetro desde un archivo en lugar de intentar escribirlo como un valor de parámetro de la línea de comandos, como, por ejemplo, cuando el parámetro es una cadena JSON compleja. Para especificar un archivo que contenga el valor, especifique la URL del archivo con el siguiente formato.

```
file://complete/path/to/file
```

- Los dos primeros caracteres de barra inclinada "/" forman parte de la especificación. Si la ruta requerida comienza con '/', el resultado será tres caracteres de barra inclinada: `file:///`  
*folder/file*.
- La dirección URL proporciona la ruta al archivo que incluye el contenido del parámetro real.

- Cuando se utilizan archivos con espacios o caracteres especiales, siguiendo las [normas de escape y entrecomillado](#) para el terminal.

### Note

Este comportamiento se deshabilita de forma automática en los parámetros que ya esperan una URL, como el parámetro que identifica una URL de plantilla de AWS CloudFormation. También puede desactivar este comportamiento desactivando la configuración de [cli\\_follow\\_urlparam](#) en el archivo de configuración de AWS CLI.

Las rutas de archivo en los siguientes ejemplos se interpretan como rutas relativas al directorio de trabajo actual.

### Linux or macOS

```
// Read from a file in the current directory
$ aws ec2 describe-instances --filters file://filter.json

// Read from a file in /tmp
$ aws ec2 describe-instances --filters file:///tmp/filter.json

// Read from a file with a filename with whitespaces
$ aws ec2 describe-instances --filters 'file://filter content.json'
```

### Windows command prompt

```
// Read from a file in C:\temp
C:\> aws ec2 describe-instances --filters file://C:\temp\filter.json

// Read from a file with a filename with whitespaces
C:\> aws ec2 describe-instances --filters "file://C:\temp\filter content.json"
```

La opción de prefijo `file://` admite ampliaciones tipo Unix, incluidas `"~/`", `"./"` y `"../"`. En Windows, la expresión `"~/`" se amplía a su directorio de usuarios, guardado en la variable de entorno `%USERPROFILE%`. Por ejemplo, en Windows 10 normalmente tendría un directorio de usuarios en `C:\Users\UserName\`.

Debe añadir el carácter de escape a los documentos JSON que están integrados como valor de otro documento JSON.

```
$ aws sqs create-queue --queue-name my-queue --attributes file://attributes.json
```

attributes.json

```
{
  "RedrivePolicy": "{\\"deadLetterTargetArn\\":\\"arn:aws:sqs:us-west-2:0123456789012:deadletter\\", \\"maxReceiveCount\\":\\"5\\"}"
}
```

## Archivos binarios

Para los comandos que toman datos binarios como un parámetro, especifique que los datos son contenido binario utilizando el prefijo `fileb://`. Los comandos que aceptan datos binarios son:

- **aws ec2 run-instances**: parámetro `--user-data`.
- **aws s3api put-object**: parámetro `--sse-customer-key`.
- **aws kms decrypt**: parámetro `--ciphertext-blob`.

En el siguiente ejemplo, se genera una clave AES binaria de 256 bits utilizando una herramienta de línea de comandos de Linux y después se proporciona dicha clave a Amazon S3 para cifrar un archivo cargado en el lado del servidor.

```
$ dd if=/dev/urandom bs=1 count=32 > sse.key
32+0 records in
32+0 records out
32 bytes (32 B) copied, 0.000164441 s, 195 kB/s
$ aws s3api put-object \
  --bucket my-bucket \
  --key test.txt \
  --body test.txt \
  --sse-customer-key fileb://sse.key \
  --sse-customer-algorithm AES256
{
  "SSECustomerKeyMD5": "iVg8oWa8sy714+FjtesrJg==",
  "SSECustomerAlgorithm": "AES256",
  "ETag": "\"a6118e84b76cf98bf04bbe14b6045c6c\""
```

```
}
```

## Archivos remotos

La AWS CLI también permite cargar parámetros desde un archivo ubicado en Internet con una URL `http://` o `https://`. En el siguiente ejemplo, se hace referencia a un archivo almacenado en un bucket Amazon S3. Esto le permite obtener acceso a archivos de parámetros desde cualquier equipo, pero requiere que el contenedor sea de acceso público.

```
$ aws ec2 run-instances \  
  --image-id ami-12345678 \  
  --block-device-mappings http://my-bucket.s3.amazonaws.com/filename.json
```

En el ejemplo anterior se presupone que el archivo `filename.json` contiene los siguientes datos JSON.

```
[  
  {  
    "DeviceName": "/dev/sdb",  
    "Ebs": {  
      "VolumeSize": 20,  
      "DeleteOnTermination": false,  
      "VolumeType": "standard"  
    }  
  }  
]
```

Por ver otro ejemplo que haga referencia a un archivo que contenga parámetros con formato JSON, consulte [Adjunte una política IAM gestionada a un usuario](#).

## Esqueletos y archivos de entrada de AWS CLI

La mayoría de los comandos de AWS CLI aceptan todas las entradas de parámetros de un archivo. Estas plantillas se pueden generar mediante opción `generate-cli-skeleton`.

### Temas

- [Sobre esqueletos y archivos de entrada de AWS CLI](#)
- [Generación de un esqueleto de comandos](#)

## Sobre esqueletos y archivos de entrada de AWS CLI

La mayoría de los comandos de AWS Command Line Interface (AWS CLI) pueden aceptar todas las entradas de parámetros desde un archivo con los parámetros `--cli-input-json`.

Esos mismos comandos proporcionan de forma efectiva el parámetro `--generate-cli-skeleton` para generar un archivo en formato JSON con todos los parámetros que pueda editar y completar. A continuación, puede ejecutar el comando con el parámetro `--cli-input-json` relevante y apuntar al archivo cumplimentado.

### Important

Varios comandos de AWS CLI no se corresponden directamente con operaciones individuales de la API de AWS, como los [aws s3 comandos de](#). Estos comandos no son compatibles con los parámetros `--generate-cli-skeleton` o `--cli-input-json` descritos en este tema. Si no sabe si un determinado comando admite estos parámetros, ejecute el siguiente comando sustituyendo los nombres del *servicio* y el *comando* con los que quiera usar.

```
$ aws service command help
```

El resultado incluye una sección `Synopsis` que muestra los parámetros que admite el comando especificado.

```
$ aws iam list-users help
...
SYNOPSIS
    list-users
    ...
    [--cli-input-json]
    ...
    [--generate-cli-skeleton <value>]
...
```

El parámetro `--generate-cli-skeleton` hace que el comando no se ejecute, sino que genere y muestre una plantilla de parámetros que se puede personalizar y utilizar como entrada de otro comando posterior. La plantilla generada incluye todos los parámetros que admite el comando.

El parámetro `--generate-cli-skeleton` acepta uno de los siguientes valores:

- `input`: la plantilla generada incluye todos los parámetros de entrada formateados como JSON. Este es el valor predeterminado.
- `output`: la plantilla generada incluye todos los parámetros de salida formateados como JSON.

Como la AWS CLI es esencialmente un "encapsulador" de la API del servicio, el archivo de esqueleto espera que el usuario haga referencia a todos los parámetros utilizando los nombres de parámetro de la API subyacente. Es probable que este nombre sea distinto al nombre del parámetro de la AWS CLI. Por ejemplo, un parámetro de la AWS CLI llamado `user-name` podría corresponderse con un parámetro de la API de servicio de AWS llamado `UserName` (observe cómo se usan las mayúsculas y que falta un guion). Le recomendamos que utilice la opción `--generate-cli-skeleton` para generar la plantilla con los nombres de parámetros "correctos" con el fin de evitar errores. También puede consultar la Guía de referencia de la API del servicio para ver los nombres de los parámetros esperados. Puede eliminar cualquier parámetro de la plantilla que no sea necesario y para el que no desee proporcionar un valor.

Por ejemplo, si ejecuta el siguiente comando, se genera la plantilla de parámetros para el comando Amazon Elastic Compute Cloud (Amazon EC2) `run-instances`.

## JSON

En el siguiente ejemplo, se muestra cómo se genera una plantilla con formato JSON utilizando el valor predeterminado (`input`) en el parámetro `--generate-cli-skeleton`.

```
$ aws ec2 run-instances --generate-cli-skeleton
```

```
{
  "DryRun": true,
  "ImageId": "",
  "MinCount": 0,
  "MaxCount": 0,
  "KeyName": "",
  "SecurityGroups": [
    ""
  ],
  "SecurityGroupIds": [
    ""
  ],
  "UserData": "",
```

```
"InstanceType": "",
"Placement": {
  "AvailabilityZone": "",
  "GroupName": "",
  "Tenancy": ""
},
"KernelId": "",
"RamdiskId": "",
"BlockDeviceMappings": [
  {
    "VirtualName": "",
    "DeviceName": "",
    "Ebs": {
      "SnapshotId": "",
      "VolumeSize": 0,
      "DeleteOnTermination": true,
      "VolumeType": "",
      "Iops": 0,
      "Encrypted": true
    },
    "NoDevice": ""
  }
],
"Monitoring": {
  "Enabled": true
},
"SubnetId": "",
"DisableApiTermination": true,
"InstanceInitiatedShutdownBehavior": "",
"PrivateIpAddress": "",
"ClientToken": "",
"AdditionalInfo": "",
"NetworkInterfaces": [
  {
    "NetworkInterfaceId": "",
    "DeviceIndex": 0,
    "SubnetId": "",
    "Description": "",
    "PrivateIpAddress": "",
    "Groups": [
      ""
    ],
    "DeleteOnTermination": true,
    "PrivateIpAddresses": [
```



```

        {
            "PrivateIpAddress": "",
            "Primary": true
        }
    ],
    "SecondaryPrivateIpAddressCount": 0,
    "AssociatePublicIpAddress": true
}
],
"IamInstanceProfile": {
    "Arn": "",
    "Name": ""
},
"EbsOptimized": true
}

```

## Generación de un esqueleto de comandos

Para generar y utilizar un archivo de esqueleto de parámetros

1. Ejecute el comando con el parámetro `--generate-cli-skeleton` para producir JSON y dirígala a un archivo para guardarla.

JSON

```
$ aws ec2 run-instances --generate-cli-skeleton input > ec2runinst.json
```

2. Abra el archivo de esqueleto de parámetros en el editor de texto y quite todos los parámetros que no necesite. Por ejemplo, puede reducir la plantilla hasta dejarla tal y como se muestra a continuación. Asegúrese de que el código JSON del archivo sigue siendo válido después de eliminar los elementos que no necesita.

JSON

```

{
    "DryRun": true,
    "ImageId": "",
    "KeyName": "",
    "SecurityGroups": [
        ""
    ],

```

```
"InstanceType": "",
"Monitoring": {
  "Enabled": true
}
}
```

En este ejemplo, dejamos el parámetro `DryRun` establecido en `true` para utilizar la característica “dry run” (simulacro) de Amazon EC2. Esta característica le permite probar el comando de forma segura sin crear ni modificar ningún recurso.

3. Rellene los valores restantes con valores adecuados para su escenario. En este ejemplo, proporcionamos el tipo de instancia, el nombre de la clave, el grupo de seguridad y el identificador de Amazon Machine Image (AMI) que vamos a utilizar. En este ejemplo, se presupone que la región apropiada es la región de AWS predeterminada. La AMI `ami-dfc39aef` es una imagen de Amazon Linux de 64 bits alojada en la región `us-west-2`. Si utiliza una región diferente, debe [encontrar el ID apropiado de la AMI que desea utilizar](#).

## JSON

```
{
  "DryRun": true,
  "ImageId": "ami-dfc39aef",
  "KeyName": "mykey",
  "SecurityGroups": [
    "my-sg"
  ],
  "InstanceType": "t2.micro",
  "Monitoring": {
    "Enabled": true
  }
}
```

4. Ejecute el comando con los parámetros completados pasando el archivo de plantilla relleno al parámetro `--cli-input-json` al utilizar el prefijo `file://`. La AWS CLI interpreta la ruta como una ruta relativa al directorio de trabajo actual, por lo que el siguiente ejemplo, en el que solo aparece el nombre de archivo y no la ruta, buscará el archivo directamente en el directorio de trabajo actual.

## JSON

```
$ aws ec2 run-instances --cli-input-json file://ec2runinst.json
```

```
A client error (DryRunOperation) occurred when calling the RunInstances operation: Request would have succeeded, but DryRun flag is set.
```

El error del simulacro indica que el código JSON está formado correctamente y que los valores de los parámetros son válidos. Si se informa de cualquier otro problema en la salida, corríjalo y repita el paso anterior hasta que aparezca el mensaje "Request would have succeeded".

5. Ahora puede establecer el parámetro `DryRun` en `false` para deshabilitar el simulacro.

## JSON

```
{
  "DryRun": false,
  "ImageId": "ami-dfc39aef",
  "KeyName": "mykey",
  "SecurityGroups": [
    "my-sg"
  ],
  "InstanceType": "t2.micro",
  "Monitoring": {
    "Enabled": true
  }
}
```

6. Ejecute el comando para que `run-instances` lance una instancia de Amazon EC2 y muestre los detalles que se generan cuando el lanzamiento se realiza correctamente. El parámetro `--output` controla el formato de salida, independientemente del formato de la plantilla de parámetros de entrada.

## JSON

```
$ aws ec2 run-instances --cli-input-json file://ec2runinst.json --output json
```

```
{
  "OwnerId": "123456789012",
```

```
"ReservationId": "r-d94a2b1",  
"Groups": [],  
"Instances": [  
...
```

## Usar sintaxis abreviada con la AWS CLI

La AWS Command Line Interface (AWS CLI) puede aceptar muchos de sus parámetros de opciones en formato JSON. Sin embargo, resultaría tedioso escribir largas listas o estructuras de JSON en la línea de comandos. Para simplificarlo, la AWS CLI también admite una sintaxis abreviada que permite una representación de los parámetros de opciones más sencilla que si se usara el formato JSON íntegro.

### Temas

- [Parámetros estructurales](#)
- [Uso de sintaxis abreviada con la AWS Command Line Interface](#)

## Parámetros estructurales

La sintaxis abreviada de la AWS CLI facilita a los usuarios la introducción de parámetros planos (con estructuras no anidadas). El formato consiste en una lista de pares de clave-valor separados por comas. Asegúrese de utilizar las reglas de [comillas](#) y caracteres de escape apropiadas para el terminal como sintaxis abreviada son cadenas.

### Linux or macOS

```
--option key1=value1,key2=value2,key3=value3
```

### PowerShell

```
--option "key1=value1,key2=value2,key3=value3"
```

Ambos equivalen al siguiente ejemplo con formato JSON.

```
--option '{"key1":"value1","key2":"value2","key3":"value3"}'
```

No debe haber espacios en blanco entre cada uno de los pares de clave-valor separados por comas. A continuación, se muestra un ejemplo del comando de `update-table` de Amazon DynamoDB con la opción `--provisioned-throughput` especificada en modo abreviado.

```
$ aws dynamodb update-table \  
  --provisioned-throughput ReadCapacityUnits=15,WriteCapacityUnits=10 \  
  --table-name MyDDBTable
```

Esto equivale al siguiente ejemplo con formato JSON.

```
$ aws dynamodb update-table \  
  --provisioned-throughput '{"ReadCapacityUnits":15,"WriteCapacityUnits":10}' \  
  --table-name MyDDBTable
```

## Uso de sintaxis abreviada con la AWS Command Line Interface

Puede especificar los parámetros de entrada en una lista de dos formas: en JSON o en sintaxis abreviada. La sintaxis abreviada de la AWS CLI ha sido diseñada para que su inserción en listas con números, cadenas o estructuras no anidadas resulte más sencilla.

A continuación se muestra el formato básico, en el que los valores de la lista están separados por un único espacio.

```
--option value1 value2 value3
```

Esto equivale al siguiente ejemplo con formato JSON.

```
--option '[value1,value2,value3]'
```

Como ya hemos mencionado anteriormente, puede especificar una lista de números, una lista de cadenas o una lista de estructuras no anidadas en formato abreviado. A continuación, se muestra un ejemplo del comando `stop-instances` para Amazon Elastic Compute Cloud (Amazon EC2), en el que el parámetro de entrada (lista de cadenas) de la opción `--instance-ids` se especifica en forma abreviada.

```
$ aws ec2 stop-instances \  
  --instance-ids i-1486157a i-1286157c i-ec3a7e87
```

Esto equivale al siguiente ejemplo con formato JSON.

```
$ aws ec2 stop-instances \  
  --instance-ids ['i-1486157a","i-1286157c","i-ec3a7e87"]'
```

En el siguiente ejemplo se muestra el comando de `create-tags` de Amazon EC2, que toma una lista de estructuras no anidadas para la opción `--tags`. La opción `--resources` especifica el identificador de la instancia que debe etiquetarse.

```
$ aws ec2 create-tags \  
  --resources i-1286157c \  
  --tags Key=My1stTag,Value=Value1 Key=My2ndTag,Value=Value2  
  Key=My3rdTag,Value=Value3
```

Esto equivale al siguiente ejemplo con formato JSON. El parámetro JSON está escrito en varias líneas para facilitar su lectura.

```
$ aws ec2 create-tags \  
  --resources i-1286157c \  
  --tags '['  
    {"Key": "My1stTag", "Value": "Value1"},  
    {"Key": "My2ndTag", "Value": "Value2"},  
    {"Key": "My3rdTag", "Value": "Value3"}  
  ]'
```

## Salida del comando de control desde el AWS CLI

En esta sección se describen las diferentes formas de controlar la salida de AWS Command Line Interface (AWS CLI). La personalización de la AWS CLI salida de su terminal puede mejorar la legibilidad, agilizar la automatización de las secuencias de comandos y facilitar la navegación a través de conjuntos de datos más grandes.

AWS CLI Algunos servicios tienen [paginación](#) del lado del servidor para sus datos para opciones de paginación adicionales.

Por último, AWS CLI tiene filtros tanto del lado [del servidor como del lado del cliente que puedes usar individualmente o en conjunto para filtrar](#) tu salida. AWS CLI

### Temas

- [Salida sensible](#)
- [Opciones de salida del servidor frente a las del cliente](#)

- [Configurar el formato de salida de la AWS CLI](#)
- [Usa las opciones de AWS CLI paginación](#)
- [AWS CLI Salida de filtro](#)

## Salida sensible

Algunas operaciones del AWS CLI pueden devolver información que podría considerarse confidencial, incluida la información de las variables de entorno. La exposición de esta información puede representar un riesgo de seguridad en ciertos escenarios; por ejemplo, la información podría incluirse en los registros de integración y despliegue continuos (CI/CD). Por lo tanto, es importante que revise cuándo incluye dichos resultados como parte de sus registros y los suprima cuando no los necesite.

Para obtener información adicional sobre la protección de datos confidenciales, consulte [the section called “Protección de los datos”](#).

Tenga en cuenta las siguientes prácticas recomendadas:

- Considere la posibilidad de recuperar sus secretos mediante programación de un almacén de secretos, como AWS Secrets Manager
- Revisa el contenido de tus registros de compilación para asegurarte de que no contienen información confidencial. Considera enfoques como canalizar `/dev/null` o capturar el resultado como un bash o una PowerShell variable para suprimir los resultados de los comandos.

El siguiente es un ejemplo de bash para redirigir los resultados, pero no los errores, a: `/dev/null`

```
$ aws s3 ls > /dev/null
```

Para obtener información específica sobre cómo suprimir la salida de su terminal, consulte la documentación del usuario del terminal que utilice.

- Tenga en cuenta el acceso a sus registros y determine el acceso de forma adecuada a su caso de uso.

## Opciones de salida del servidor frente a las del cliente

AWS CLI Tiene filtros tanto [del lado del servidor como del lado del cliente que puede usar individualmente o en conjunto para filtrar](#) los resultados. AWS CLI El filtrado del lado del servidor se

procesa primero y devuelve su salida para el filtrado del lado del cliente. La API del servicio admite el filtrado del servidor. El cliente admite el filtrado del lado del cliente mediante el parámetro. AWS CLI `--query`

Las opciones de salida del lado del servidor son funciones compatibles directamente con la API. Servicio de AWS Los datos filtrados o paginados no se envían al cliente, lo que puede acelerar los tiempos de respuesta HTTP y mejorar el ancho de banda para conjuntos de datos más grandes.

Las opciones de salida del cliente son características creadas por la AWS CLI. Todos los datos se envían al cliente y, a continuación, este AWS CLI filtra o archiva el contenido que se muestra. Las operaciones del cliente no ahorran velocidad ni ancho de banda en el caso de conjuntos de datos más grandes.

Cuando las opciones del servidor y del cliente se utilizan juntas, las operaciones del servidor se completan primero y, después, se envían al cliente para las operaciones del cliente. De este modo se aprovechan los posibles ahorros de velocidad y ancho de banda de las opciones del servidor y, al mismo tiempo, se utilizan características adicionales de la AWS CLI para obtener el resultado deseado.

## Configurar el formato de salida de la AWS CLI

Este tema describe los diferentes formatos de salida para la AWS Command Line Interface (AWS CLI). La AWS CLI admite los siguientes formatos de salida:

- **[json](#)**: la salida se formatea como una cadena [JSON](#).
- **[text](#)**: la salida tiene el formato de varias líneas de valores de cadena separados por tabuladores. Esto puede ser útil para pasar la salida a un procesador de texto, como `grep`, `sed` o `awk`.
- **[table](#)**: el resultado tiene el formato de una tabla en la que se usan los caracteres `+|-` para los bordes de celda. Normalmente, la información se presenta en un formato que es más fácil de leer que los demás formatos, pero que no es útil para programar.

## Cómo seleccionar el formato de salida

Tal como se explica en el tema de [configuración](#), el formato de salida se puede especificar de tres formas diferentes:

- Uso de la opción **`output`** en un perfil con nombre en el archivo **`config`**: el siguiente ejemplo establece el formato de la salida predeterminado como `text`.



```
[default]
output=text
```

- Uso de la variable de entorno de **AWS\_DEFAULT\_OUTPUT**: la siguiente salida establece el formato como `table` para los comandos de esta sesión de línea de comandos hasta que se cambie la variable o finalice la sesión. Con esta variable de entorno, se invalida cualquier valor establecido en el archivo `config`.

```
$ export AWS_DEFAULT_OUTPUT="table"
```

- Uso de la opción **--output** en la línea de comandos: el siguiente ejemplo establece la salida únicamente de este comando como `json`. El uso de esta opción en el comando anula cualquier variable de entorno definida actualmente o el valor en el archivo `config`.

```
$ aws swf list-domains --registration-status REGISTERED --output json
```

### Important

El tipo de salida que especifique cambia cómo opera la opción `--query`:

- Si especifica `--output text`, la salida se pagina antes de que se aplique el filtro `--query` y AWS CLI ejecuta la consulta una vez en cada página de la salida. Debido a esto, la consulta incluye el primer elemento que coincide en cada página, lo que puede dar lugar a una salida adicional inesperada. Para filtrar más la salida, puede usar otras herramientas de la línea de comandos como `head` o `tail`.
- Si especifica `--output json`, la salida se procesará por completo como una única estructura nativa antes de aplicar el filtro `--query`. La AWS CLI ejecuta la consulta solo una vez en toda la estructura y produce un resultado filtrado que luego se ofrece como salida.

## Formato de salida JSON

[JSON](#) es el formato de salida predeterminado de la AWS CLI. La mayoría de lenguajes de programación pueden decodificar fácilmente cadenas JSON con funciones integradas o con

bibliotecas disponibles públicamente. Puede combinar la salida JSON con la [opción `--query`](#) de una forma eficaz para aplicar filtros y formato a la salida JSON de la AWS CLI.

Para aplicar un filtrado más avanzado que el que permite `--query`, considere la posibilidad de utilizar `jq`, un procesador JSON de línea de comandos. Puede descargarlo y consultar el tutorial oficial en <http://stedolan.github.io/jq/>.

A continuación, se muestra un ejemplo de una salida JSON:

```
$ aws iam list-users --output json
```

```
{
  "Users": [
    {
      "Path": "/",
      "UserName": "Admin",
      "UserId": "AIDA111111111111EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:user/Admin",
      "CreateDate": "2014-10-16T16:03:09+00:00",
      "PasswordLastUsed": "2016-06-03T18:37:29+00:00"
    },
    {
      "Path": "/backup/",
      "UserName": "backup-user",
      "UserId": "AIDA222222222222EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:user/backup/backup-user",
      "CreateDate": "2019-09-17T19:30:40+00:00"
    },
    {
      "Path": "/",
      "UserName": "cli-user",
      "UserId": "AIDA333333333333EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:user/cli-user",
      "CreateDate": "2019-09-17T19:11:39+00:00"
    }
  ]
}
```

## Formato de salida de texto

El formato `text` organiza la salida de la AWS CLI en líneas delimitadas por tabulaciones. Este formato funciona bien con herramientas de texto tradicionales de Unix, como `grep`, `sed` y `awk`, así como con el procesamiento de texto que realiza PowerShell.

El formato de salida de `text` sigue la estructura básica que se muestran a continuación. Las columnas se ordenan alfabéticamente según los nombres de las claves correspondientes del objeto JSON subyacente.

```
IDENTIFIER sorted-column1 sorted-column2
IDENTIFIER2 sorted-column1 sorted-column2
```

A continuación se muestra un ejemplo de la salida de `text`. Cada campo está separado por tabulaciones de los demás, con una tabulación adicional si hay un campo vacío.

```
$ aws iam list-users --output text
```

```
USERS   arn:aws:iam::123456789012:user/Admin           2014-10-16T16:03:09+00:00
        2016-06-03T18:37:29+00:00 /                AIDA111111111111EXAMPLE Admin
USERS   arn:aws:iam::123456789012:user/backup/backup-user 2019-09-17T19:30:40+00:00
        /backup/ AIDA222222222222EXAMPLE backup-user
USERS   arn:aws:iam::123456789012:user/cli-user           2019-09-17T19:11:39+00:00
        /                AIDA333333333333EXAMPLE cli-user
```

La cuarta columna es el campo `PasswordLastUsed` y está vacía en las dos últimas entradas, ya que esos usuarios nunca inician sesión en la AWS Management Console.

### Important

Le recomendamos encarecidamente que si especifica una salida `text`, utilice siempre la opción `--query` para garantizar un comportamiento coherente.

Esto se debe a que el formato de texto ordena alfabéticamente las columnas de salida por el nombre de la clave del objeto JSON subyacente que el servicio de AWS devuelve y podría haber recursos similares con distintos nombres de clave. Por ejemplo, la representación JSON de una instancia de Amazon EC2 basada en Linux podría tener elementos que no están presentes en la representación JSON de una instancia basada en Windows, o viceversa. Además, los recursos podrían tener elementos de valor de clave añadidos o eliminados en futuras actualizaciones, alterando el orden de las columnas. Aquí es donde

`--query` incrementa la funcionalidad de la salida `text` para proporcionarle un control total sobre el formato de salida.

En el ejemplo siguiente, el comando especifica qué elementos se van a mostrar y define el orden de las columnas con la notación en lista `[key1, key2, ...]`. Esto le ofrece la plena confianza de que los valores de clave correctos se mostrarán siempre en la columna prevista. Por último, tenga en cuenta que la AWS CLI devuelve el valor `None` para las claves que no existen.

```
$ aws iam list-users --output text --query 'Users[*].
[UserName,Arn,CreateDate,PasswordLastUsed,UserId]'
```

```
Admin          arn:aws:iam::123456789012:user/Admin
2014-10-16T16:03:09+00:00  2016-06-03T18:37:29+00:00  AIDA111111111111EXAMPLE
backup-user    arn:aws:iam::123456789012:user/backup-user
2019-09-17T19:30:40+00:00  None                        AIDA222222222222EXAMPLE
cli-user       arn:aws:iam::123456789012:user/cli-backup
2019-09-17T19:11:39+00:00  None                        AIDA333333333333EXAMPLE
```

El siguiente ejemplo muestra cómo se puede usar `grep` y `awk` con la salida `text` del comando `aws ec2 describe-instances`. El primer comando muestra la zona de disponibilidad, el estado actual y el ID de cada instancia en la salida `text`. El segundo comando procesa esa salida para mostrar únicamente los ID de instancia de todas las instancias en ejecución en la zona de disponibilidad `us-west-2a`.

```
$ aws ec2 describe-instances --query 'Reservations[*].Instances[*].
[Placement.AvailabilityZone, State.Name, InstanceId]' --output text
```

```
us-west-2a    running i-4b41a37c
us-west-2a    stopped i-a071c394
us-west-2b    stopped i-97a217a0
us-west-2a    running i-3045b007
us-west-2a    running i-6fc67758
```

```
$ aws ec2 describe-instances --query 'Reservations[*].Instances[*].
[Placement.AvailabilityZone, State.Name, InstanceId]' --output text | grep us-west-2a |
grep running | awk '{print $3}'
```

```
i-4b41a37c
i-3045b007
i-6fc67758
```

El siguiente ejemplo va un paso más allá y muestra no solo cómo filtrar el resultado, sino cómo utilizar esa salida para automatizar los tipos de instancias cambiantes para cada instancia detenida.

```
$ aws ec2 describe-instances --query 'Reservations[*].Instances[*].[State.Name,
  InstanceId]' --output text |
> grep stopped |
> awk '{print $2}' |
> while read line;
> do aws ec2 modify-instance-attribute --instance-id $line --instance-type '{"Value":
  "m1.medium"}';
> done
```

La salida `text` también puede resultar útil en PowerShell. Como las columnas de la salida `text` están delimitadas por tabulaciones, puede dividir fácilmente la salida en una matriz utilizando el delimitador ``t` de PowerShell. El siguiente comando muestra el valor de la tercera columna (`InstanceId`) si la primera columna (`AvailabilityZone`) coincide con la cadena `us-west-2a`.

```
PS C:\>aws ec2 describe-instances --query 'Reservations[*].Instances[*].
[Placement.AvailabilityZone, State.Name, InstanceId]' --output text |
%{if ($_.split("`t")[0] -match "us-west-2a") { $_.split("`t")[2]; } }
```

```
-4b41a37c
i-a071c394
i-3045b007
i-6fc67758
```

Tenga en cuenta que, aunque en el ejemplo anterior se muestra cómo utilizar el parámetro `--query` para analizar los objetos JSON subyacentes y extraer la columna deseada, PowerShell dispone de una funcionalidad propia para gestionar JSON si la compatibilidad entre plataformas no es un problema. En lugar de tratar la salida como texto, como requieren la mayoría de los shells de comandos, PowerShell le permite utilizar el cmdlet `ConvertFrom-JSON` para producir un objeto estructurado jerárquicamente. A continuación, podrá acceder directamente al miembro que desee desde ese objeto.

```
(aws ec2 describe-instances --output json | ConvertFrom-
Json).Reservations.Instances.InstanceId
```

### Tip

Si obtiene un texto en la salida y lo filtra con un solo campo utilizando el parámetro `--query`, la salida será una sola línea de valores separados por tabuladores. Para obtener cada valor en una línea independiente, puede colocar el campo de salida entre corchetes, tal y como se muestra en los siguientes ejemplos.

Salida de una sola línea separada por tabuladores:

```
$ aws iam list-groups-for-user --user-name susan --output text --query
"Groups[].GroupName"
```

```
HRDepartment    Developers    SpreadsheetUsers    LocalAdmins
```

Cada valor en su propia línea incluyendo `[GroupName]` entre corchetes:

```
$ aws iam list-groups-for-user --user-name susan --output text --query
"Groups[].[GroupName]"
```

```
HRDepartment
Developers
SpreadsheetUsers
LocalAdmins
```

## Formato de salida de tabla

El formato `table` produce representaciones legibles para los humanos de la salida de la AWS CLI compleja en forma de tabla.

```
$ aws iam list-users --output table
```

```
-----
|
| ListUsers |
```

```

+-----+
+
||
Users                                                     ||
|+-----+-----+-----+-----+
+-----+-----+-----+-----+
||           Arn           |           CreateDate           |
PasswordLastUsed | Path |           UserId           |           UserName           ||
|+-----+-----+-----+-----+
+-----+-----+-----+-----+
|| arn:aws:iam::123456789012:user/Admin           | 2014-10-16T16:03:09+00:00 |
2016-06-03T18:37:29+00:00 | /           | AIDA111111111111EXAMPLE | Admin           ||
|| arn:aws:iam::123456789012:user/backup/backup-user | 2019-09-17T19:30:40+00:00 | |
           | /backup/ | AIDA222222222222EXAMPLE | backup-user ||
|| arn:aws:iam::123456789012:user/cli-user           | 2019-09-17T19:11:39+00:00 |
           | /           | AIDA333333333333EXAMPLE | cli-user           ||
+-----+-----+-----+-----+
+

```

Puede combinar la opción `--query` con el formato `table` para mostrar un conjunto de elementos preseleccionados a partir de la salida en bruto. Observe las diferencias de la salida entre las notaciones de diccionario y lista: en el primer ejemplo los nombres de columna están ordenados alfabéticamente, y en el segundo ejemplo las columnas sin nombre se ordenan según lo definido por el usuario. Para obtener más información acerca de la opción `--query`, consulte [AWS CLI Salida de filtro](#).

```

$ aws ec2 describe-volumes --query 'Volumes[*].
{ID:VolumeId,InstanceId:Attachments[0].InstanceId,AZ:AvailabilityZone,Size:Size}' --
output table

```

```

-----
|           DescribeVolumes           |
+-----+-----+-----+-----+
|   AZ   |   ID   | InstanceId | Size |
+-----+-----+-----+-----+
| us-west-2a | vol-e11a5288 | i-a071c394 | 30 |
| us-west-2a | vol-2e410a47 | i-4b41a37c | 8 |
+-----+-----+-----+-----+

```

```

$ aws ec2 describe-volumes --query 'Volumes[*].
[VolumeId,Attachments[0].InstanceId,AvailabilityZone,Size]' --output table

```

```

-----
|                               DescribeVolumes                               |
+-----+-----+-----+-----+
| vol-e11a5288| i-a071c394 | us-west-2a | 30 |
| vol-2e410a47| i-4b41a37c | us-west-2a | 8  |
+-----+-----+-----+-----+

```

## Usa las opciones de AWS CLI paginación

En este tema se describen las diferentes formas de paginar la salida de la AWS CLI.

### Paginación del lado del servidor

En el caso de los comandos que pueden devolver una lista grande de elementos, the AWS Command Line Interface (AWS CLI) tiene varias opciones para controlar la cantidad de elementos que se incluyen en la salida cuando se AWS CLI llama a un servicio API para completar la lista.

Las opciones incluyen:

- [Cómo usar el parámetro `—no-paginate`](#)
- [Cómo usar el parámetro `—page-size`](#)
- [Cómo usar el parámetro `—max-items`](#)
- [Cómo usar el parámetro `—starting-token`](#)

De forma predeterminada, AWS CLI utiliza un tamaño de página determinado por el servicio individual y recupera todos los elementos disponibles. Por ejemplo, Simple Storage Service (Amazon S3) tiene un tamaño de página predeterminado de 1000. Si ejecuta `aws s3api list-objects` en un bucket de Simple Storage Service (Amazon S3) que contiene 3500 objetos, AWS CLI realiza cuatro llamadas a Simple Storage Service (Amazon S3) y se encarga de administrar por usted la lógica de paginación específica del servicio en segundo plano y de devolver los 3500 objetos en la salida final.

### Cómo usar el parámetro `—no-paginate`

La opción `--no-paginate` desactiva los siguientes tokens de paginación en el lado del cliente. Cuando se utiliza un comando, de forma predeterminada realiza varias llamadas AWS CLI automáticamente para obtener todos los resultados posibles y crear una paginación. Una llamada por cada página. Al deshabilitar la paginación, AWS CLI solo se llama una vez a la primera página de resultados del comando.



Por ejemplo, si `aws s3api list-objects` ejecuta un bucket de Amazon S3 que contiene 3500 objetos, AWS CLI solo realizará la primera llamada a Amazon S3 y devolverá solo los primeros 1000 objetos del resultado final.

```
$ aws s3api list-objects \  
  --bucket my-bucket \  
  --no-paginate  
{  
  "Contents": [  
  ...
```

### Cómo usar el parámetro `—page-size`

Si observa algún problema al ejecutar los comandos de la lista en un gran número de recursos, el tamaño de página predeterminado puede ser demasiado grande. Esto puede provocar que las llamadas a los AWS servicios superen el tiempo máximo permitido y generar un error de «tiempo de espera agotado». Puede usar la `--page-size` opción para especificar que se AWS CLI solicite un número menor de elementos por cada llamada al AWS servicio. AWS CLI Sigue recuperando la lista completa, pero realiza un mayor número de API llamadas de servicio en segundo plano y recupera un número menor de elementos con cada llamada. Esto ofrece a las llamadas individuales una mayor posibilidad de que tengan éxito sin que se agote el tiempo de espera. Cambiar el tamaño de la página no afecta al resultado; solo afecta al número de API llamadas que hay que realizar para generar el resultado.

```
$ aws s3api list-objects \  
  --bucket my-bucket \  
  --page-size 100  
{  
  "Contents": [  
  ...
```

### Cómo usar el parámetro `—max-items`

Para incluir menos elementos a la vez en la AWS CLI salida, utilice la `--max-items` opción. AWS CLI Sigue gestionando la paginación con el servicio tal y como se ha descrito anteriormente, pero solo imprime el número de elementos que especifique cada vez.

```
$ aws s3api list-objects \  
  --bucket my-bucket \  
  --max-items 10
```

```

--max-items 100
{
  "NextToken": "eyJNYXJrZXIiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxfQ==",
  "Contents": [
    ...
  ]
}

```

### Cómo usar el parámetro `--starting-token`

Si el número de elementos de output (`--max-items`) es inferior al número total de elementos devueltos por las API llamadas subyacentes, el resultado incluye un elemento `NextToken` que puede pasar a un comando posterior para recuperar el siguiente conjunto de elementos. En el siguiente ejemplo se muestra cómo usar el valor `NextToken` devuelto por el ejemplo anterior y le permite recuperar los segundos 100 elementos.

#### Note

El parámetro `--starting-token` no puede ser nulo ni está vacío. Si el comando anterior no devuelve un valor `NextToken`, no hay más elementos para devolver y no es necesario llamar al comando de nuevo.

```

$ aws s3api list-objects \
  --bucket my-bucket \
  --max-items 100 \
  --starting-token eyJNYXJrZXIiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxfQ==
{
  "Contents": [
    ...
  ]
}

```

Es posible que el AWS servicio especificado no devuelva los artículos en el mismo orden cada vez que llames. Si especifica diferentes valores para `--page-size` y `--max-items`, puede obtener resultados inesperados en el que faltan elementos o están duplicados. Para evitar esto, utilice el mismo número con `--page-size` y `--max-items` para sincronizar la paginación de la AWS CLI con la del servicio subyacente. También puede recuperar la lista completa y realizar las operaciones de paginación necesarias de forma local.

## AWS CLI Salida de filtro

El AWS Command Line Interface (AWS CLI) tiene filtros tanto del lado del servidor como del lado del cliente que puede usar individualmente o en conjunto para filtrar la salida. AWS CLI El filtrado del lado del servidor se procesa primero y devuelve su salida para el filtrado del lado del cliente.

- El filtrado del lado del servidor es compatible con el yAPI, por lo general, se implementa con un parámetro. `--filter` El servicio solo devuelve resultados coincidentes, lo que puede acelerar los tiempos de HTTP respuesta para conjuntos de datos de gran tamaño.
- El cliente que utiliza el parámetro admite el filtrado del lado del AWS CLI cliente. `--query` Este parámetro tiene capacidades que el filtrado del servidor puede no tener.

### Temas

- [Filtrado en el servidor](#)
- [Filtrado del lado del cliente](#)
- [Combinación del filtrado del lado del cliente y del servidor](#)
- [Recursos adicionales de](#)

### Filtrado en el servidor

El filtrado del lado del servidor lo proporciona AWS CLI el servicio. AWS API El AWS servicio solo devuelve los registros de la HTTP respuesta que coinciden con su filtro, lo que puede acelerar los tiempos de HTTP respuesta para conjuntos de datos de gran tamaño. Como el servicio define el filtrado del lado del servidorAPI, los nombres y funciones de los parámetros varían de un servicio a otro. Algunos nombres de parámetros comunes utilizados para filtrar son:

- `--filter` tales como [ses](#) y [ce](#).
- `--filters` tales como [ec2](#), [autoescalado](#) y [rds](#).
- Nombres que empiezan por la palabra `filter`, por ejemplo `--filter-expression` para el comando [aws dynamodb scan](#).

Para obtener información sobre si un comando específico tiene filtrado del lado del servidor y cuáles son las reglas de filtrado, consulte la guía de referencia de la 2 de la [guía de AWS CLI referencia](#).

## Filtrado del lado del cliente

AWS CLI Proporciona funciones de filtrado integradas JSON en el lado del cliente con el parámetro. `--query` El parámetro `--query` es una potente herramienta que se puede utilizar para personalizar el contenido y el estilo de la salida. El `--query` parámetro toma la HTTP respuesta que llega del servidor y filtra los resultados antes de mostrarlos. Como la HTTP respuesta completa se envía al cliente antes de filtrarla, el filtrado del lado del cliente puede ser más lento que el filtrado del lado del servidor para conjuntos de datos grandes.

Las consultas utilizan la [JMESPathsyntaxis](#) para crear expresiones con las que filtrar los resultados. Para aprender la JMESPath sintaxis, consulte el [tutorial](#) en el JMESPathsitio web.

### Important

El tipo de salida que especifique cambia cómo opera la opción `--query`:

- Si lo especifica `--output text`, el resultado se pagina antes de aplicar el `--query` filtro y AWS CLI ejecuta la consulta una vez en cada página del resultado. Debido a esto, la consulta incluye el primer elemento que coincide en cada página, lo que puede dar lugar a una salida adicional inesperada. Para filtrar más la salida, puede usar otras herramientas de la línea de comandos como `head` o `tail`.
- Si especifica `--output json`, la salida se procesará por completo como una única estructura nativa antes de aplicar el filtro `--query`. AWS CLI Ejecuta la consulta solo una vez en toda la estructura, lo que produce un resultado filtrado que, a continuación, se muestra.

### Temas de filtrado del lado del cliente

- [Antes de comenzar](#)
- [Identificadores](#)
- [Cómo seleccionar de una lista](#)
- [Filtrado de datos anidados](#)
- [Aplanamiento de los resultados](#)
- [Filtrado para valores específicos](#)
- [Expresiones de canalización](#)
- [Filtrado para varios valores de identificador](#)

- [Agregado de etiquetas a valores de identificador](#)
- [Funciones](#)
- [Ejemplos --query avanzados](#)

Antes de comenzar

Cuando utilice las expresiones de filtro utilizadas en estos ejemplos, asegúrese de utilizar las reglas de entrecomillado correctas para el shell de terminal. Para obtener más información, consulte [the section called “Comillas con cadenas”](#).

El siguiente JSON resultado muestra un ejemplo de lo que puede producir el `--query` parámetro. El resultado describe tres EBS volúmenes de Amazon adjuntos a distintas EC2 instancias de Amazon.

Ejemplo de resultado

```
$ aws ec2 describe-volumes
{
  "Volumes": [
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2013-09-17T00:55:03.000Z",
          "InstanceId": "i-a071c394",
          "VolumeId": "vol-e11a5288",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
        }
      ],
      "VolumeType": "standard",
      "VolumeId": "vol-e11a5288",
      "State": "in-use",
      "SnapshotId": "snap-f23ec1c8",
      "CreateTime": "2013-09-17T00:55:03.000Z",
      "Size": 30
    },
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2013-09-18T20:26:16.000Z",
```

```
    "InstanceId": "i-4b41a37c",
    "VolumeId": "vol-2e410a47",
    "State": "attached",
    "DeleteOnTermination": true,
    "Device": "/dev/sda1"
  }
],
"VolumeType": "standard",
"VolumeId": "vol-2e410a47",
"State": "in-use",
"SnapshotId": "snap-708e8348",
"CreateTime": "2013-09-18T20:26:15.000Z",
"Size": 8
},
{
  "AvailabilityZone": "us-west-2a",
  "Attachments": [
    {
      "AttachTime": "2020-11-20T19:54:06.000Z",
      "InstanceId": "i-1jd73kv8",
      "VolumeId": "vol-a1b3c7nd",
      "State": "attached",
      "DeleteOnTermination": true,
      "Device": "/dev/sda1"
    }
  ],
  "VolumeType": "standard",
  "VolumeId": "vol-a1b3c7nd",
  "State": "in-use",
  "SnapshotId": "snap-234087fb",
  "CreateTime": "2020-11-20T19:54:05.000Z",
  "Size": 15
}
]
}
```

## Identificadores

Los identificadores son las etiquetas de los valores de salida. Al crear filtros, se utilizan identificadores para reducir los resultados de la consulta. En el siguiente ejemplo de salida, se resaltan todos los identificadores como `Volumes`, `AvailabilityZone` y `AttachTime`.

```
$ aws ec2 describe-volumes
```

```
{
  "Volumes": [
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2013-09-17T00:55:03.000Z",
          "InstanceId": "i-a071c394",
          "VolumeId": "vol-e11a5288",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
        }
      ],
      "VolumeType": "standard",
      "VolumeId": "vol-e11a5288",
      "State": "in-use",
      "SnapshotId": "snap-f23ec1c8",
      "CreateTime": "2013-09-17T00:55:03.000Z",
      "Size": 30
    },
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2013-09-18T20:26:16.000Z",
          "InstanceId": "i-4b41a37c",
          "VolumeId": "vol-2e410a47",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
        }
      ],
      "VolumeType": "standard",
      "VolumeId": "vol-2e410a47",
      "State": "in-use",
      "SnapshotId": "snap-708e8348",
      "CreateTime": "2013-09-18T20:26:15.000Z",
      "Size": 8
    },
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
```

```

    "AttachTime": "2020-11-20T19:54:06.000Z",
    "InstanceId": "i-1jd73kv8",
    "VolumeId": "vol-a1b3c7nd",
    "State": "attached",
    "DeleteOnTermination": true,
    "Device": "/dev/sda1"
  }
],
"VolumeType": "standard",
"VolumeId": "vol-a1b3c7nd",
"State": "in-use",
"SnapshotId": "snap-234087fb",
"CreateTime": "2020-11-20T19:54:05.000Z",
"Size": 15
}
]
}

```

Para obtener más información, consulte [Identificadores](#) en el sitio JMESPathweb.

### Cómo seleccionar de una lista

Una lista o matriz es un identificador que va seguido de un corchete “[” como `Volumes` y `Attachments` en la [the section called “Antes de comenzar”](#).

### Sintaxis

```
<listName>[ ]
```

Para filtrar todos los resultados de una matriz, puede usar la notación comodín. Las expresiones [comodín](#) son expresiones utilizadas para devolver elementos utilizando la notación `*`.

En el siguiente ejemplo se consulta todo el contenido de `Volumes`.

```

$ aws ec2 describe-volumes \
  --query 'Volumes[*]'
[
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-17T00:55:03.000Z",
        "InstanceId": "i-a071c394",

```



```

    "VolumeId": "vol-e11a5288",
    "State": "attached",
    "DeleteOnTermination": true,
    "Device": "/dev/sda1"
  }
],
"VolumeType": "standard",
"VolumeId": "vol-e11a5288",
"State": "in-use",
"SnapshotId": "snap-f23ec1c8",
"CreateTime": "2013-09-17T00:55:03.000Z",
"Size": 30
},
{
  "AvailabilityZone": "us-west-2a",
  "Attachments": [
    {
      "AttachTime": "2020-11-20T19:54:06.000Z",
      "InstanceId": "i-1jd73kv8",
      "VolumeId": "vol-a1b3c7nd",
      "State": "attached",
      "DeleteOnTermination": true,
      "Device": "/dev/sda1"
    }
  ],
  "VolumeType": "standard",
  "VolumeId": "vol-a1b3c7nd",
  "State": "in-use",
  "SnapshotId": "snap-234087fb",
  "CreateTime": "2020-11-20T19:54:05.000Z",
  "Size": 15
}
]

```

Para ver un volumen específico en la matriz por índice, se llama al índice de matriz. Por ejemplo, el primer elemento de la matriz `Volumes` tiene un índice de 0, lo que resulta en la consulta `Volumes[0]`. Para obtener más información sobre los índices de matriz, consulte las [expresiones de índice](#) en el JMESPath sitio web.

```

$ aws ec2 describe-volumes \
  --query 'Volumes[0]'
{
  "AvailabilityZone": "us-west-2a",

```

```

"Attachments": [
  {
    "AttachTime": "2013-09-17T00:55:03.000Z",
    "InstanceId": "i-a071c394",
    "VolumeId": "vol-e11a5288",
    "State": "attached",
    "DeleteOnTermination": true,
    "Device": "/dev/sda1"
  }
],
"VolumeType": "standard",
"VolumeId": "vol-e11a5288",
"State": "in-use",
"SnapshotId": "snap-f23ec1c8",
"CreateTime": "2013-09-17T00:55:03.000Z",
"Size": 30
}

```

Para ver un rango específico de volúmenes por índice, utilice `slice` con la siguiente sintaxis, donde `iniciar` es el índice de matriz inicial, `detener` es el índice en el que el filtro deja de procesarse y `paso` es el intervalo de omisión.

### Sintaxis

```
<arrayName>[<start>:<stop>:<step>]
```

Si se omite cualquiera de estos elementos de la expresión de sector, utilizan los siguientes valores predeterminados:

- Iniciar: el primer índice de la lista, 0.
- Detener: el último índice de la lista.
- Paso: sin saltar paso, donde el valor es 1.

Para devolver solo los dos primeros volúmenes, utilice un valor inicial de 0, un valor de detención de 2 y un valor de paso de 1, como se muestra en el siguiente ejemplo.

```

$ aws ec2 describe-volumes \
  --query 'Volumes[0:2:1]'
[
  {
    "AvailabilityZone": "us-west-2a",

```

```

"Attachments": [
  {
    "AttachTime": "2013-09-17T00:55:03.000Z",
    "InstanceId": "i-a071c394",
    "VolumeId": "vol-e11a5288",
    "State": "attached",
    "DeleteOnTermination": true,
    "Device": "/dev/sda1"
  }
],
"VolumeType": "standard",
"VolumeId": "vol-e11a5288",
"State": "in-use",
"SnapshotId": "snap-f23ec1c8",
"CreateTime": "2013-09-17T00:55:03.000Z",
"Size": 30
},
{
  "AvailabilityZone": "us-west-2a",
  "Attachments": [
    {
      "AttachTime": "2013-09-18T20:26:16.000Z",
      "InstanceId": "i-4b41a37c",
      "VolumeId": "vol-2e410a47",
      "State": "attached",
      "DeleteOnTermination": true,
      "Device": "/dev/sda1"
    }
  ],
  "VolumeType": "standard",
  "VolumeId": "vol-2e410a47",
  "State": "in-use",
  "SnapshotId": "snap-708e8348",
  "CreateTime": "2013-09-18T20:26:15.000Z",
  "Size": 8
}
]

```

Dado que este ejemplo contiene valores predeterminados, puede acortar el segmento de `Volumes[0:2:1]` a `Volumes[:2]`.

En el ejemplo siguiente se omiten los valores predeterminados y se devuelve cada dos volúmenes de toda la matriz.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[:2]'
[
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-17T00:55:03.000Z",
        "InstanceId": "i-a071c394",
        "VolumeId": "vol-e11a5288",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-e11a5288",
    "State": "in-use",
    "SnapshotId": "snap-f23ec1c8",
    "CreateTime": "2013-09-17T00:55:03.000Z",
    "Size": 30
  },
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2020-11-20T19:54:06.000Z",
        "InstanceId": "i-1jd73kv8",
        "VolumeId": "vol-a1b3c7nd",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-a1b3c7nd",
    "State": "in-use",
    "SnapshotId": "snap-234087fb",
    "CreateTime": "2020-11-20T19:54:05.000Z",
    "Size": 15
  }
]
```

Los pasos también pueden utilizar números negativos para filtrar en el orden inverso de una matriz como se muestra en el siguiente ejemplo.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[::-2]'
[
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2020-11-20T19:54:06.000Z",
        "InstanceId": "i-1jd73kv8",
        "VolumeId": "vol-a1b3c7nd",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-a1b3c7nd",
    "State": "in-use",
    "SnapshotId": "snap-234087fb",
    "CreateTime": "2020-11-20T19:54:05.000Z",
    "Size": 15
  },
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-17T00:55:03.000Z",
        "InstanceId": "i-a071c394",
        "VolumeId": "vol-e11a5288",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-e11a5288",
    "State": "in-use",
    "SnapshotId": "snap-f23ec1c8",
    "CreateTime": "2013-09-17T00:55:03.000Z",
    "Size": 30
  }
]
```

```
}
]
```

Para obtener más información, consulte [Slices](#) en el JMESPathsitio web.

## Filtrado de datos anidados

Para restringir el filtrado de `Volumes[*]` para los valores anidados, se utilizan subexpresiones agregando un punto y sus criterios de filtro.

## Sintaxis

```
<expression>.<expression>
```

En el siguiente ejemplo se muestra toda la información de `Attachments` para todos los volúmenes.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[*].Attachments'
[
  [
    {
      "AttachTime": "2013-09-17T00:55:03.000Z",
      "InstanceId": "i-a071c394",
      "VolumeId": "vol-e11a5288",
      "State": "attached",
      "DeleteOnTermination": true,
      "Device": "/dev/sda1"
    }
  ],
  [
    {
      "AttachTime": "2013-09-18T20:26:16.000Z",
      "InstanceId": "i-4b41a37c",
      "VolumeId": "vol-2e410a47",
      "State": "attached",
      "DeleteOnTermination": true,
      "Device": "/dev/sda1"
    }
  ],
  [
    {
      "AttachTime": "2020-11-20T19:54:06.000Z",
      "InstanceId": "i-1jd73kv8",
```

```

    "VolumeId": "vol-a1b3c7nd",
    "State": "attached",
    "DeleteOnTermination": true,
    "Device": "/dev/sda1"
  }
]
]
```

Para filtrar más los valores anidados, agregue la expresión para cada indentificador anidado. En el ejemplo siguiente, se muestran los State para todos los Volumes.

```

$ aws ec2 describe-volumes \
  --query 'Volumes[*].Attachments[*].State'
[
  [
    "attached"
  ],
  [
    "attached"
  ],
  [
    "attached"
  ]
]
```

### Aplanamiento de los resultados

Para obtener más información, consulte [SubExpressions](#) en el JMESPath sitio web.

Puede aplanar los resultados de `Volumes[*].Attachments[*].State` eliminando la notación comodín que resulta en la consulta `Volumes[*].Attachments[].State`. El aplanamiento a menudo es útil para mejorar la legibilidad de los resultados.

```

$ aws ec2 describe-volumes \
  --query 'Volumes[*].Attachments[].State'
[
  "attached",
  "attached",
  "attached"
]
```

Para obtener más información, consulte [Flatten](#) en el sitio JMESPath web.

## Filtrado para valores específicos

Para filtrar valores específicos de una lista, utilice una expresión de filtro como se muestra en la siguiente sintaxis.

### Sintaxis

```
? <expression> <comparator> <expression>]
```

Los comparadores de expresiones incluyen ==, !=, <, <=, > y >=. En el ejemplo siguiente se filtra para el VolumeIds, para todos los Volumes en un AttachedState.

```
$ aws ec2 describe-volumes \  
  --query 'Volumes[*].Attachments[?State==`attached`].VolumeId'  
[  
  [  
    "vol-e11a5288"  
  ],  
  [  
    "vol-2e410a47"  
  ],  
  [  
    "vol-a1b3c7nd"  
  ]  
]
```

Esto se puede aplanar, lo que resulta en el siguiente ejemplo.

```
$ aws ec2 describe-volumes \  
  --query 'Volumes[*].Attachments[?State==`attached`].VolumeId[]'  
[  
  "vol-e11a5288",  
  "vol-2e410a47",  
  "vol-a1b3c7nd"  
]
```

En el ejemplo siguiente se filtra para el VolumeIds, de todos los Volumes que tienen un tamaño inferior a 20.

```
$ aws ec2 describe-volumes \  
  --query 'Volumes[?Size < `20`].VolumeId'  
[
```



```
"vol-2e410a47",
"vol-a1b3c7nd"
]
```

Para obtener más información, consulte [Filtrar expresiones](#) en el JMESPathsitio web.

## Expresiones de canalización

Puede canalizar los resultados de un filtro a una nueva lista y, a continuación, filtrar el resultado con otra expresión utilizando la siguiente sintaxis:

### Sintaxis

```
<expression> | <expression>]
```

En el siguiente ejemplo se toman los resultados del filtro de la expresión `Volumes[*].Attachments[].InstanceId` y genera el primer resultado en la matriz.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[*].Attachments[].InstanceId | [0]'
"i-a071c394"
```

En este ejemplo, se crea primero la matriz a partir de la siguiente expresión.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[*].Attachments[].InstanceId'
"i-a071c394",
"i-4b41a37c",
"i-1jd73kv8"
```

Y luego devuelve el primer elemento de esa matriz.

```
"i-a071c394"
```

Para obtener más información, consulte [Pipe Expressions](#) en el JMESPathsitio web.

## Filtrado para varios valores de identificador

Para filtrar por varios identificadores, utilice una lista de selección múltiple mediante la siguiente sintaxis:

### Sintaxis

```
<listName>[].[<expression>, <expression>]
```

En el siguiente ejemplo, VolumeId y VolumeType se filtran en la lista Volumes lo que resulta en la siguiente expresión.

```
$ aws ec2 describe-volumes \  
  --query 'Volumes[][VolumeId, VolumeType]'  
[  
  [  
    "vol-e11a5288",  
    "standard"  
  ],  
  [  
    "vol-2e410a47",  
    "standard"  
  ],  
  [  
    "vol-a1b3c7nd",  
    "standard"  
  ]  
]
```

Para agregar datos anidados a la lista, agregue otra lista de selección múltiple. El siguiente ejemplo amplía el ejemplo anterior al filtrar también InstanceId y State en la lista de Attachments anidada. Esto da como resultado la siguiente expresión.

```
$ aws ec2 describe-volumes \  
  --query 'Volumes[][VolumeId, VolumeType, Attachments[][InstanceId, State]]'  
[  
  [  
    "vol-e11a5288",  
    "standard",  
    [  
      [  
        "i-a071c394",  
        "attached"  
      ]  
    ]  
  ],  
  [  
    "vol-2e410a47",  
    "standard",  
    []  
  ]  
]
```

```
[
  [
    "i-4b41a37c",
    "attached"
  ]
],
[
  "vol-a1b3c7nd",
  "standard",
  [
    [
      "i-1jd73kv8",
      "attached"
    ]
  ]
]
]
```

Para que sea más legible, aplane la expresión como se muestra en el siguiente ejemplo.

```
$ aws ec2 describe-volumes \
  --query 'Volumes[][VolumeId, VolumeType, Attachments[][InstanceId, State]][[]]'
```

```
[
  "vol-e11a5288",
  "standard",
  [
    "i-a071c394",
    "attached"
  ],
  "vol-2e410a47",
  "standard",
  [
    "i-4b41a37c",
    "attached"
  ],
  "vol-a1b3c7nd",
  "standard",
  [
    "i-1jd73kv8",
    "attached"
  ]
]
```

Para obtener más información, consulte la [lista de selección múltiple](#) en el JMESPathsitio web.

## Agregado de etiquetas a valores de identificador

Para facilitar la lectura de este resultado, utilice un hash de selección múltiple con la siguiente sintaxis.

### Sintaxis

```
<listName>[].{<label>: <expression>, <label>: <expression>}
```

No es necesario que la etiqueta de identificación sea la misma que el nombre del identificador. En el ejemplo siguiente se utiliza la etiqueta `VolumeType` para los valores `VolumeType`.

```
$ aws ec2 describe-volumes \  
  --query 'Volumes[].{VolumeType: VolumeType}'  
[  
  {  
    "VolumeType": "standard",  
  },  
  {  
    "VolumeType": "standard",  
  },  
  {  
    "VolumeType": "standard",  
  }  
]
```

Para mayor simplicidad, en el siguiente ejemplo se conservan los nombres de los identificadores de cada etiqueta y se muestra el `VolumeId`, `VolumeType`, `InstanceId` y `State` de todos los volúmenes:

```
$ aws ec2 describe-volumes \  
  --query 'Volumes[].{VolumeId: VolumeId, VolumeType: VolumeType, InstanceId:  
  Attachments[0].InstanceId, State: Attachments[0].State}'  
[  
  {  
    "VolumeId": "vol-e11a5288",  
    "VolumeType": "standard",  
    "InstanceId": "i-a071c394",  
    "State": "attached"
```

```

},
{
  "VolumeId": "vol-2e410a47",
  "VolumeType": "standard",
  "InstanceId": "i-4b41a37c",
  "State": "attached"
},
{
  "VolumeId": "vol-a1b3c7nd",
  "VolumeType": "standard",
  "InstanceId": "i-1jd73kv8",
  "State": "attached"
}
]

```

Para obtener más información, consulte el [hash de selección múltiple](#) en el JMESPath sitio web.

## Funciones

La JMESPath sintaxis contiene muchas funciones que puede utilizar para sus consultas. Para obtener información sobre JMESPath las funciones, consulte [Funciones integradas](#) en el JMESPath sitio web.

Para demostrar cómo puede incorporar una función a sus consultas, en el siguiente ejemplo se utiliza la función `sort_by`. La función `sort_by` ordena una matriz utilizando una expresión como clave de ordenación utilizando la siguiente sintaxis:

## Sintaxis

```
sort_by(<listName>, <sort expression>)[].<expression>
```

En el ejemplo siguiente se utiliza el [ejemplo de hash de selección múltiple](#) y ordena la salida por `VolumeId`.

```

$ aws ec2 describe-volumes \
  --query 'sort_by(Volumes, &VolumeId)[].{VolumeId: VolumeId, VolumeType: VolumeType,
InstanceId: Attachments[0].InstanceId, State: Attachments[0].State}'
[
  {
    "VolumeId": "vol-2e410a47",
    "VolumeType": "standard",

```

```

    "InstanceId": "i-4b41a37c",
    "State": "attached"
  },
  {
    "VolumeId": "vol-a1b3c7nd",
    "VolumeType": "standard",
    "InstanceId": "i-1jd73kv8",
    "State": "attached"
  },
  {
    "VolumeId": "vol-e11a5288",
    "VolumeType": "standard",
    "InstanceId": "i-a071c394",
    "State": "attached"
  }
]

```

Para obtener más información, consulte [sort\\_by en el sitio web](#). JMESPath

## Ejemplos **--query** avanzados

Para extraer información de un elemento específico

En el siguiente ejemplo, se usa el parámetro `--query` para encontrar un elemento específico en una lista y, a continuación, extrae la información de ese elemento. En el ejemplo, se muestran todas las `AvailabilityZones` asociadas con el punto de enlace de servicio especificado. Extrae el elemento de la lista `ServiceDetails` que tiene el `ServiceName` especificado; a continuación, genera el campo `AvailabilityZones` a partir de ese elemento seleccionado.

```

$ aws --region us-east-1 ec2 describe-vpc-endpoint-services \
  --query 'ServiceDetails[?ServiceName==`com.amazonaws.us-
east-1.ecs`].AvailabilityZones'
[
  [
    "us-east-1a",
    "us-east-1b",
    "us-east-1c",
    "us-east-1d",
    "us-east-1e",
    "us-east-1f"
  ]
]

```

Para mostrar instantáneas después de la fecha de creación especificada

En el siguiente ejemplo se muestra cómo enumerar todas las instantáneas que se han creado después de una fecha especificada, incluidos solo algunos de los campos disponibles en la salida.

```
$ aws ec2 describe-snapshots --owner self \
  --output json \
  --query 'Snapshots[?StartTime>=`2018-02-07`].
{Id:SnapshotId,VID:VolumeId,Size:VolumeSize}'
[
  {
    "id": "snap-0effb42b7a1b2c3d4",
    "vid": "vol-0be9bb0bf12345678",
    "Size": 8
  }
]
```

Para mostrar las más recientes AMIs

En el siguiente ejemplo, se muestran las cinco imágenes de máquina de Amazon (AMIs) más recientes que ha creado, ordenadas de la más reciente a la más antigua.

```
$ aws ec2 describe-images \
  --owners self \
  --query 'reverse(sort_by(Images,&CreationDate))[:5].{id:ImageId,date:CreationDate}'
[
  {
    "id": "ami-0a1b2c3d4e5f60001",
    "date": "2018-11-28T17:16:38.000Z"
  },
  {
    "id": "ami-0a1b2c3d4e5f60002",
    "date": "2018-09-15T13:51:22.000Z"
  },
  {
    "id": "ami-0a1b2c3d4e5f60003",
    "date": "2018-08-19T10:22:45.000Z"
  },
  {
    "id": "ami-0a1b2c3d4e5f60004",
    "date": "2018-05-03T12:04:02.000Z"
  },
  {
```

```

    "id": "ami-0a1b2c3d4e5f60005",
    "date": "2017-12-13T17:16:38.000Z"
  }
]

```

Para mostrar instancias de escalado automático en mal estado

El siguiente ejemplo muestra solo la InstanceId para las instancias en mal estado en el grupo de Auto Scaling especificado.

```

$ aws autoscaling describe-auto-scaling-groups \
  --auto-scaling-group-name My-AutoScaling-Group-Name \
  --output text \
  --query 'AutoScalingGroups[*].Instances[?HealthStatus==`Unhealthy`].InstanceId'

```

Para incluir volúmenes con la etiqueta especificada

En el siguiente ejemplo se describen todas las instancias con una etiqueta test. Siempre y cuando haya otra etiqueta al lado de test adjunta al volumen, el volumen se sigue devolviendo en los resultados.

La siguiente expresión para devolver todas las etiquetas con la etiqueta test en una matriz. Cualquier etiqueta de que no sea la etiqueta test contiene un valor null.

```

$ aws ec2 describe-volumes \
  --query 'Volumes[*].Tags[?Value == `test`]'

```

Para excluir volúmenes con la etiqueta especificada

El siguiente ejemplo describe todas las instancias sin una etiqueta test. El uso de una expresión ?Value != `test` simple no funciona para excluir un volumen ya que los volúmenes pueden tener varias etiquetas. Siempre y cuando haya otra etiqueta al lado de test adjunta al volumen, el volumen se sigue devolviendo en los resultados.

Para excluir todos los volúmenes con la etiqueta test, comience con la siguiente expresión para devolver todas las etiquetas con la etiqueta test en una matriz. Cualquier etiqueta de que no sea la etiqueta test contiene un valor null.

```

$ aws ec2 describe-volumes \
  --query 'Volumes[*].Tags[?Value == `test`]'

```



Luego, filtre todos los resultados positivos `test` con la función `not_null`.

```
$ aws ec2 describe-volumes \  
  --query 'Volumes[!not_null(Tags[?Value == `test`].Value)]'
```

Canalice los resultados para aplanar los que resultan en la siguiente consulta.

```
$ aws ec2 describe-volumes \  
  --query 'Volumes[!not_null(Tags[?Value == `test`].Value)] | []'
```

## Combinación del filtrado del lado del cliente y del servidor

Se puede utilizar el filtrado del lado del cliente y del servidor juntos. El filtrado del lado del servidor se completa primero y envía los datos al cliente que luego el parámetro `--query` filtra. Si utiliza conjuntos de datos de gran tamaño, utilizar primero el filtrado del lado del servidor puede reducir la cantidad de datos que se envían al cliente por cada AWS CLI llamada y, al mismo tiempo, conservar la potente personalización que proporciona el filtrado del lado del cliente.

En el siguiente ejemplo, se enumeran los EC2 volúmenes de Amazon que utilizan filtros tanto del lado del servidor como del lado del cliente. El servicio filtra una lista de todos los volúmenes asociados de la zona de disponibilidad `us-west-2a`. El parámetro `--query` limita aún más la salida a solo los volúmenes con un valor de `Size` mayor que 50 y muestra solo los campos especificados con nombres definidos por el usuario.

```
$ aws ec2 describe-volumes \  
  --filters "Name=availability-zone,Values=us-west-2a" "Name=status,Values=attached" \  
  \  
  --query 'Volumes[?Size > `50`].{Id:VolumeId,Size:Size,Type:VolumeType}' \  
  [ \  
    { \  
      "Id": "vol-0be9bb0bf12345678", \  
      "Size": 80, \  
      "VolumeType": "gp2" \  
    } \  
  ]
```

El siguiente ejemplo recupera una lista de imágenes que cumplen varios criterios. A continuación, utiliza el parámetro `--query` para ordenar la salida por `CreationDate`, que selecciona solo las más recientes. Finalmente, muestra el valor de `ImageId` de esa imagen.

```
$ aws ec2 describe-images \  
  --owners amazon \  
  --filters "Name=name,Values=amzn*gp2" "Name=virtualization-type,Values=hvm" \  
  "Name=root-device-type,Values=ebs" \  
  --query "sort_by(Images, &CreationDate)[-1].ImageId" \  
  --output text  
ami-00ced3122871a4921
```

El siguiente ejemplo muestra el número de volúmenes disponibles que son superiores a 1000 IOPS y se utiliza `length` para contar cuántos hay en una lista.

```
$ aws ec2 describe-volumes \  
  --filters "Name=status,Values=available" \  
  --query 'length(Volumes[?Iops > `1000`])'  
3
```

## Recursos adicionales de

### JMESPathTerminal

JMESPathTerminal es un comando de terminal interactivo para experimentar con JMESPath expresiones que se utilizan para el filtrado del lado del cliente. al usar el comando `jp term`, el terminal muestra los resultados inmediatos de la consulta a medida que usted escribe. Puede canalizar la AWS CLI salida directamente a la terminal, lo que permite experimentar con consultas avanzadas.

El siguiente ejemplo canaliza la `aws ec2 describe-volumes` salida directamente a la JMESPath Terminal.

```
$ aws ec2 describe-volumes | jp term
```

Para obtener más información sobre la JMESPath Terminal y las instrucciones de instalación, consulte [JMESPathTerminal](#) on GitHub.

### Utilidad jq

La utilidad `jq` le proporciona una forma de transformar su salida en el lado del cliente a un formato de salida que desee. Para obtener más información `jq` e instrucciones de instalación, consulte [jq](#) on GitHub.

## Códigos de devolución de la AWS CLI

El código de retorno suele ser un código oculto enviado después de ejecutar un comando de la AWS Command Line Interface (AWS CLI) que describe el estado del comando. Puede utilizar el comando `echo` para mostrar el código enviado desde el último comando de la AWS CLI y utilice estos códigos para determinar si un comando se ha realizado correctamente y por qué un comando puede tener un error. Además de los códigos de retorno, puede ver más detalles sobre un error ejecutando sus comandos con el conmutador `--debug`. Este conmutador produce un informe detallado de los pasos que usa la AWS CLI para procesar el comando e indica cuál fue el resultado de cada paso.

Para determinar el código de devolución de un comando de la AWS CLI, ejecute uno de los siguientes comandos inmediatamente después de ejecutar el comando de la CLI.

### Linux and macOS

```
$ echo $?  
0
```

### Windows PowerShell

```
PS> echo $lastexitcode  
0
```

### Windows Command Prompt

```
C:\> echo %errorlevel%  
0
```

A continuación se indican los valores código de devolución que se pueden devolver al finalizar la ejecución de un comando de la AWS Command Line Interface (AWS CLI).

Código	Significado
0	El servicio respondió con un código de estado de respuesta HTTP de 200 que indica que AWS CLI no generó errores y el servicio de AWS al que se envía la solicitud.
1	Error en una o varias operaciones de transferencia de Amazon S3. Limitado a comandos de S3.

Códig	Significado
2	<p>El significado de este código de devolución depende del comando:</p> <ul style="list-style-type: none"><li>• Aplicable a todos los comandos de AWS CLI: no se pudo analizar el comando introducido. Errores de análisis que pueden deberse, entre otras cosas, a falta de subcomandos o argumentos necesarios o al uso de comandos o argumentos desconocidos.</li><li>• Limitado a comandos S3: uno o más archivos marcados para transferencia durante el proceso de transferencia. Sin embargo, el resto de los archivos marcados para transferencia se han transferido correctamente. Los archivos que se omiten durante el proceso de transferencia pueden ser: archivos que no existen; archivos de dispositivos orientados a caracteres, dispositivos orientados a bloques, colas FIFO o sockets, y archivos para los que el usuario no tiene permiso de lectura.</li></ul>
130	<p>El comando se ha visto interrumpido por SIGINT. Esta es la señal que usted envió para cancelar un comando con <code>Ctrl+C</code>.</p>
255	<p>Error del comando. La AWS CLI o el servicio de AWS al que se había enviado la solicitud produjeron errores.</p>

## Crear y usar atajos de AWS CLI comandos llamados alias

Los alias son atajos que puedes crear en AWS Command Line Interface (AWS CLI) para acortar los comandos o scripts que utilizas con frecuencia. Puede crear alias en el campo `alias` ubicado en la carpeta de configuración.

### Temas

- [Requisitos previos](#)
- [Paso 1: Creación del archivo de alias](#)
- [Paso 2: Creación de un alias](#)
- [Paso 3: Cómo llamar a un alias](#)
- [Ejemplos de repositorio de alias](#)
- [Recursos](#)

## Requisitos previos

Para utilizar los comandos de alias, debe seguir estos pasos:

- Instalar y configurar la AWS CLI. Para obtener más información, consulte [Instalar la AWS CLI](#) y [Credenciales de autenticación y acceso](#).
- Usa una AWS CLI versión mínima de 1.11.24 o 2.0.0.
- (Opcional) Para usar scripts bash de AWS CLI alias, debes usar un terminal compatible con bash.

## Paso 1: Creación del archivo de alias

Para crear el `alias` archivo, puede usar la navegación de archivos y un editor de texto, o usar la terminal que prefiera mediante el procedimiento. step-by-step Para crear rápidamente el archivo de alias, utilice el siguiente bloque de comandos.

Linux and macOS

```
$ mkdir -p ~/.aws/cli  
$ echo '[toplevel]' > ~/.aws/cli/alias
```

Windows

```
C:\> md %USERPROFILE%\.aws\cli  
C:\> echo [toplevel] > %USERPROFILE%\.aws\cli\alias
```

Para crear el archivo de alias

1. Cree una carpeta con un nombre `cli` en su carpeta AWS CLI de configuración. De forma predeterminada, la carpeta de configuración es `~/.aws/` en Linux o macOS y `%USERPROFILE%\.aws\` en Windows. Puede crearla a través de la navegación de archivos o mediante el siguiente comando.

Linux and macOS

```
$ mkdir -p ~/.aws/cli
```

## Windows

```
C:\> md %USERPROFILE%\aws\cli
```

La ruta predeterminada resultante de la carpeta `cli` es `~/.aws/cli/` en Linux o macOS y `%USERPROFILE%\aws\cli` en Windows.

2. En la carpeta `cli`, cree un archivo de texto con el nombre `alias` sin extensión y agréguele `[toplevel]` a la primera línea. Puede crear este archivo a través de su editor de texto preferido o utilizar el siguiente comando.

## Linux and macOS

```
$ echo '[toplevel]' > ~/.aws/cli/alias
```

## Windows

```
C:\> echo [toplevel] > %USERPROFILE%\aws\cli\alias
```

## Paso 2: Creación de un alias

Puede crear un alias mediante comandos básicos o scripting de bash.

### Creación de un alias de comando básico

Puede crear su alias agregando un comando con la siguiente sintaxis en el archivo `alias` que creó en el paso anterior.

### Sintaxis

```
aliasname = command [--options]
```

La **aliasname** es como llamas a tu alias. La **command** es el comando al que desea llamar, que puede incluir otros alias. Puede incluir opciones o parámetros en su alias, o agregarlos al llamar a su alias.

En el siguiente ejemplo se muestra cómo crear un alias llamado `aws whoami` con el comando [aws sts get-caller-identity](#). Dado que este alias llama a un comando AWS CLI existente, puede escribir el comando sin el prefijo `aws`.

```
whoami = sts get-caller-identity
```

En el siguiente ejemplo se toma el ejemplo anterior `whoami` y se agregan las opciones de filtro `Account` y texto `output`.

```
whoami2 = sts get-caller-identity --query Account --output text
```

## Creación de un alias de subcomando

### Note

La función de alias de los subcomandos requiere una AWS CLI versión mínima de 1.11.24 o 2.0.0

Puede crear un alias para subcomandos si agrega un comando mediante la siguiente sintaxis en el archivo `alias` que creó en el paso anterior.

### Sintaxis

```
[command commandGroup]  
aliasname = command [--options]
```

La *commandGroup* es el espacio de nombres de comandos, por ejemplo, el comando está en el grupo de comandos `aws ec2 describe-regions`. La *aliasname* es lo que llamas a tu alias. La *command* es el comando al que desea llamar, que puede incluir otros alias. Puede incluir opciones o parámetros en su alias, o agregarlos al llamar a su alias.

En el siguiente ejemplo se muestra cómo crear un alias llamado `aws ec2 regions` con el comando [aws ec2 describe-regions](#). Dado que este alias llama a un comando AWS CLI existente en el espacio de nombres de comando `ec2`, puede escribir el comando sin el prefijo `aws ec2`.

```
[command ec2]  
regions = describe-regions --query Regions[].RegionName
```

Para crear alias de comandos fuera del espacio de nombres del comando, anteponga al comando completo un signo de exclamación. En el siguiente ejemplo se muestra cómo crear un alias llamado `aws ec2 instance-profiles` con el comando [aws iam list-instance-profiles](#).

```
[command ec2]
instance-profiles = !aws iam list-instance-profiles
```

### Note

Los alias solo utilizan espacios de nombres de comando existentes y no puede crear nuevos. Por ejemplo, no puede crear un alias con la sección `[command johnsmith]` ya que el espacio de nombres de comando `johnsmith` no existe.

## Creación de un alias de scripting bash

### Warning

Para usar scripts bash de AWS CLI alias, debes usar un terminal compatible con bash

Puede crear un alias utilizando scripts bash para procesos más avanzados utilizando la siguiente sintaxis.

### Sintaxis

```
aliasname =
    !f() {
        script content
    }; f
```

La *aliasname* es como llamas a tu alias y *script content* es el script que desea ejecutar cuando llame al alias.

El siguiente ejemplo utiliza `opendns` para generar la dirección IP actual. Dado que puede usar alias en otros alias, el siguiente alias `myip` es útil para permitir o revocar el acceso a su dirección IP desde otros alias.

```
myip =
```



```
!f() {
  dig +short myip.opendns.com @resolver1.opendns.com
}; f
```

El siguiente ejemplo de script llama al `aws myip` alias anterior para autorizar la entrada de un grupo de EC2 seguridad de Amazon a su dirección IP.

```
authorize-my-ip =
!f() {
  ip=$(aws myip)
  aws ec2 authorize-security-group-ingress --group-id ${1} --cidr $ip/32 --protocol
tcp --port 22
}; f
```

Cuando llama a alias que usan scripting de Bash, las variables siempre se pasan en el orden en que las ingresó. En el scripting de Bash, los nombres de las variables no se tienen en cuenta, solo el orden en que aparecen. En los siguientes ejemplos de alias `textalert`, la variable para la opción `--message` es la primera y la opción `--phone-number` es la segunda.

```
textalert =
!f() {
  aws sns publish --message "${1}" --phone-number ${2}
}; f
```

### Paso 3: Cómo llamar a un alias

Para ejecutar el alias que creó en su archivo `alias` utilice la siguiente sintaxis. Puede agregar opciones adicionales al llamar a su alias.

Sintaxis

```
$ aws aliasname
```

En el siguiente ejemplo se utiliza el alias de comando `aws whoami`.

```
$ aws
whoami
{
  "UserId": "A12BCD34E5FGHI6JKLM",
```

```
"Account": "1234567890987",
"Arn": "arn:aws:iam::1234567890987:user/userName"
}
```

El siguiente ejemplo utiliza el alias `aws whoami` con opciones adicionales para devolver solo el número Account en la salida `text`.

```
$ aws whoami --query Account --output
text
1234567890987
```

En el siguiente ejemplo se utiliza el [alias de subcomando](#) `aws ec2 regions`.

```
$ aws ec2
regions
[
  "ap-south-1",
  "eu-north-1",
  "eu-west-3",
  "eu-west-2",
  ...
```

## Cómo llamar a un alias usando variables de scripting de Bash

Cuando se llama a alias que usan scripting de Bash, las variables se pasan en el orden en que se introducen. En el scripting de Bash, el nombre de las variables no se tiene en cuenta, solo el orden en que aparecen. Por ejemplo, en el siguiente alias `textalert`, la variable de la opción `--message` está primera y `--phone-number` está segunda.

```
textalert =
!f() {
  aws sns publish --message "${1}" --phone-number ${2}
}; f
```

Cuando llama al alias `textalert`, debe pasar variables en el mismo orden en que se ejecutan en el alias. En el siguiente ejemplo utilizamos las variables `$message` y `$phone`. La variable `$message` se pasa como `${1}` para la opción `--message` y la variable `$phone` se pasa como `${2}` para la opción `--phone-number`. Esto da como resultado una llamada correcta al alias `textalert` para enviar un mensaje.

```
$ aws textalert $message
  $phone
{
  "MessageId": "1ab2cd3e4-fg56-7h89-i01j-2klmn34567"
}
```

En el siguiente ejemplo, el orden se cambia al llamar al alias para \$phone y \$message. La variable \$phone se pasa como \${1} para la opción --message y la variable \$message se pasa como \${2} para la opción --phone-number. Dado que las variables están fuera de servicio, el alias pasa las variables de manera incorrecta. Esto provoca un error porque el contenido de \$message no coinciden con los requisitos de formato de número de teléfono para la opción --phone-number.

```
$ aws textalert $phone
  $message
usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:

aws help
aws <command> help
aws <command> <subcommand> help

Unknown options: text
```

## Ejemplos de repositorio de alias

El [repositorio de AWS CLI alias](#) de GitHub contiene ejemplos de AWS CLI alias creados por el equipo de AWS CLI desarrolladores y la comunidad. Puede usar todo el ejemplo del archivo `alias` o tomar alias individuales para su propio uso.

### Warning

Al ejecutar los comandos de esta sección, se elimina su archivo `alias` existente. Para evitar sobrescribir el archivo de alias existente, cambie la ubicación de descarga.

Para utilizar alias del repositorio

1. Instale Git Para obtener instrucciones de instalación, consulte [Introducción - Instalación de Git](#) en la Documentación de Git.

2. Instale el comando `jp`. El comando `jp` se utiliza en el alias `tostring`. Para obtener instrucciones de instalación, consulte el [JMESPatharchivo \(jp\) README .md](#) en GitHub.
3. Instale el comando `jq`. El comando `jq` se utiliza en el alias `tostring-with-jq`. Para obtener instrucciones de instalación, consulte el [JSONprocesador \(jq\)](#) en GitHub.
4. Descargue el archivo `alias` de la siguiente manera:
  - Ejecute los siguientes comandos que se descargan del repositorio y se copian el archivo `alias` en la carpeta de configuración.

#### Linux and macOS

```
$ git clone https://github.com/awslabs/awscli-aliases.git
$ mkdir -p ~/.aws/cli
$ cp awscli-aliases/alias ~/.aws/cli/alias
```

#### Windows

```
C:\> git clone https://github.com/awslabs/awscli-aliases.git
C:\> md %USERPROFILE%\aws\cli
C:\> copy awscli-aliases\alias %USERPROFILE%\aws\cli
```

- Descárguelo directamente del repositorio y guárdelo en la `cli` carpeta de su carpeta AWS CLI de configuración. De forma predeterminada, la carpeta de configuración es `~/.aws/` en Linux o macOS y `%USERPROFILE%\aws\` en Windows.
5. Para verificar que los alias funcionan, ejecute el siguiente alias.

```
$ aws whoami
```

Esto muestra la misma respuesta que el comando `aws sts get-caller-identity`:

```
{
  "Account": "012345678901",
  "UserId": "AIUAINBADX2VEG2TC6HD6",
  "Arn": "arn:aws:iam::012345678901:user/myuser"
}
```

## Recursos

- El [repositorio de AWS CLI alias GitHub](#) contiene ejemplos de AWS CLI alias creados por el equipo de AWS CLI desarrolladores y la contribución de la AWS CLI comunidad.
- El anuncio de la función alias de [AWS re:Invent 2016: The Effective AWS CLI User ya está publicado](#). YouTube
- [aws sts get-caller-identity](#)
- [aws ec2 describe-instances](#)
- [aws sns publish](#)

# Ejemplos de código

En este capítulo se proporciona una colección de ejemplos que muestran cómo utilizar the AWS Command Line Interface (AWS CLI) con Servicios de AWS.

En esta guía AWS CLI se incluyen los siguientes tipos de ejemplos:

- Ejemplos de [comandos guiados: ejemplos](#) de comandos guiados para la Guía del AWS CLI usuario sobre cómo AWS CLI utilizarlos con algunos Servicios de AWS. Suele ser ejemplos más detallados que los ejemplos de la [guía de AWS CLI referencia](#) de la .
- [AWS CLI ejemplos de comandos: ejemplos](#) de comandos de código abierto que también están disponibles en la [guía de AWS CLI referencia](#) de la . Los ejemplos de comandos están alojados en el [AWS CLI](#) repositorio de GitHub.
- [AWS CLI uso de ejemplos de código de secuencias de comandos en Bash: ejemplos](#) de secuencias de comandos en bash de código abierto. [Los ejemplos de secuencias de comandos en Bash están alojados en el repositorio de ejemplos de código en.AWSGitHub](#)

## Comentarios sobre los ejemplos

¿No encuentra lo que necesita? Solicite un ejemplo de comando utilizando el enlace Enviar comentarios al final de esta página o en la página de comandos correspondiente de la guía de [AWS CLI referencia de la AWS CLI versión 2 de la guía](#) .

¿Quiere contribuir? Contribuya con ejemplos de AWS CLI comandos en el [repositorio AWS de ejemplos de código](#) en GitHub. Para obtener más información sobre cómo contribuir, consulte los [ejemplos AWS CLI de código de los pasos rápidos para contribuir](#) en GitHub las páginas.

## Ejemplos de AWS CLI comandos guiados

The AWS Command Line Interface (AWS CLI) es una herramienta de código abierto que le permite interactuar con una amplia gama de comandos de Servicios de AWS uso en su shell de línea de comandos. En esta sección se proporcionan ejemplos guiados que muestran cómo aprovecharlos para acceder AWS CLI a algunos de los. Servicios de AWS Esto incluye algunos de los AWS CLI comandos personalizados, como los `aws s3` comandos de alto nivel. Estos ejemplos de

comandos muestran las acciones más comunes que se utilizan en algunos casos Servicios de AWS y proporcionan recursos adicionales para obtener más información.

Tanto si es un AWS usuario experimentado como si es nuevo en el mundo AWS CLI, estos ejemplos guiados le servirán de recurso para agilizar sus AWS operaciones.

Para obtener una referencia completa de todos los comandos disponibles para cada uno de ellos Servicio de AWS, consulte la [guía de AWS CLI referencia](#) de la . Además, puede utilizar la [ayuda de línea de comandos integrada](#) para explorar la variedad de comandos Servicios de AWS, opciones y funciones del AWS CLI.

Para ver más ejemplos de comandos que no están disponibles en esta sección, consulte la [AWS CLI ejemplos de comandos](#) sección. Estos son ejemplos de comandos de código abierto que también están disponibles en la [guía de AWS CLI referencia](#) de la . Los ejemplos de comandos están alojados en el [AWS CLI](#) repositorio de GitHub.

Para ver ejemplos de secuencias de comandos bash de código abierto, consulte. [the section called “Ejemplos de script Bash”](#) Los ejemplos de secuencias de comandos en Bash están alojados en el repositorio de ejemplos de [AWS código](#) en. GitHub

## Servicios

- [Utilice Amazon DynamoDB con AWS CLI](#)
- [Usa Amazon EC2 con AWS CLI](#)
- [Utilice Amazon S3 Glacier con AWS CLI](#)
- [AWS Identity and Access Management Úselo desde AWS CLI](#)
- [Utilice Amazon S3 con AWS CLI](#)
- [Usar Amazon SNS con la AWS CLI](#)

## Utilice Amazon DynamoDB con AWS CLI

### Introducción a Amazon DynamoDB

#### [¿Qué es Amazon DynamoDB?](#)

El AWS Command Line Interface (AWS CLI) proporciona soporte para todos los servicios de AWS bases de datos, incluido Amazon DynamoDB. Puede usarlo AWS CLI para operaciones

improvisadas, como crear una tabla. También puede usarla para incluir operaciones de DynamoDB en scripts de utilidades.

Para obtener más información sobre el uso AWS CLI de DynamoDB, [dynamodb](#) consulte la Referencia de comandos.AWS CLI

Para ver una lista de los AWS CLI comandos de DynamoDB, utilice el siguiente comando.

```
$ aws dynamodb help
```

## Temas

- [Requisitos previos](#)
- [Creación y uso de tablas de DynamoDB](#)
- [Uso de DynamoDB Local](#)
- [Recursos](#)

## Requisitos previos

Para ejecutar los comandos de dynamodb, necesitará:

- Instalar y configurar la AWS CLI. Para obtener más información, consulte [Instalar la AWS CLI y Credenciales de autenticación y acceso](#).

## Creación y uso de tablas de DynamoDB

El formato de la línea de comandos se compone de un nombre de comando de DynamoDB, seguido de los parámetros de dicho comando. AWS CLI Admite la [sintaxis CLI abreviada de los valores de los parámetros y](#) es completa. JSON

En el siguiente ejemplo, se crea una tabla llamada MusicCollection.

```
$ aws dynamodb create-table \  
  --table-name MusicCollection \  
  --attribute-definitions AttributeName=Artist,AttributeType=S  
  AttributeName=SongTitle,AttributeType=S \  
  --key-schema AttributeName=Artist,KeyType=HASH  
  AttributeName=SongTitle,KeyType=RANGE \  
  
```



```
--provisioned-throughput ReadCapacityUnits=1,WriteCapacityUnits=1
```

A continuación, puede añadir nuevas líneas a la tabla con comandos similares a los que se muestran en el siguiente ejemplo. En estos ejemplos se utiliza una combinación de sintaxis abreviada y JSON

```
$ aws dynamodb put-item \  
  --table-name MusicCollection \  
  --item '{  
    "Artist": {"S": "No One You Know"},  
    "SongTitle": {"S": "Call Me Today"} ,  
    "AlbumTitle": {"S": "Somewhat Famous"}  
  }' \  
  --return-consumed-capacity TOTAL  
{  
  "ConsumedCapacity": {  
    "CapacityUnits": 1.0,  
    "TableName": "MusicCollection"  
  }  
}
```

```
$ aws dynamodb put-item \  
  --table-name MusicCollection \  
  --item '{  
    "Artist": {"S": "Acme Band"},  
    "SongTitle": {"S": "Happy Day"} ,  
    "AlbumTitle": {"S": "Songs About Life"}  
  }' \  
  --return-consumed-capacity TOTAL  
{  
  "ConsumedCapacity": {  
    "CapacityUnits": 1.0,  
    "TableName": "MusicCollection"  
  }  
}
```

Puede resultar difícil redactar de forma válida JSON en un comando de una sola línea. Para hacerlo más fácil, AWS CLI pueden leer JSON archivos. Por ejemplo, considere el siguiente JSON fragmento, que está almacenado en un archivo denominado `expression-attributes.json`

```
{  
  ":v1": {"S": "No One You Know"},  
  ":v2": {"S": "Call Me Today"}  
}
```

```
}
```

Puede utilizar ese archivo para emitir una solicitud de query mediante la AWS CLI. En el siguiente ejemplo, se usa el contenido del archivo `expression-attributes.json` como valor del parámetro `--expression-attribute-values`.

```
$ aws dynamodb query --table-name MusicCollection \  
  --key-condition-expression "Artist = :v1 AND SongTitle = :v2" \  
  --expression-attribute-values file://expression-attributes.json  
{  
  "Count": 1,  
  "Items": [  
    {  
      "AlbumTitle": {  
        "S": "Somewhat Famous"  
      },  
      "SongTitle": {  
        "S": "Call Me Today"  
      },  
      "Artist": {  
        "S": "No One You Know"  
      }  
    }  
  ],  
  "ScannedCount": 1,  
  "ConsumedCapacity": null  
}
```

## Uso de DynamoDB Local

Además de DynamoDB, puede usarlo AWS CLI con DynamoDB Local. DynamoDB Local es un pequeño servidor y base de datos del lado del cliente que imita el servicio de DynamoDB. DynamoDB Local le permite escribir aplicaciones que utilizan DynamoDB sin manipular ninguna tabla o dato del API servicio web de DynamoDB. En su lugar, todas las API acciones se redirigen a una base de datos local. lo que le permite ahorrar en rendimiento aprovisionado, almacenamiento de datos y tarifas de transferencia de datos.

Para obtener más información sobre DynamoDB Local y cómo usarlo con, consulte las siguientes secciones de AWS CLI la Guía para desarrolladores de Amazon [DynamoDB](#):

- [DynamoDB Local](#)

- [Uso de la AWS CLI con DynamoDB Local](#)

## Recursos

AWS CLI referencia:

- [aws dynamodb](#)
- [aws dynamodb create-table](#)
- [aws dynamodb put-item](#)
- [aws dynamodb query](#)

Referencia de servicio:

- [DynamoDB Local](#) en la Guía para desarrolladores de Amazon DynamoDB
- [Uso de la AWS CLI con DynamoDB Local](#) en la Guía para desarrolladores de Amazon DynamoDB

## Usa Amazon EC2 con AWS CLI

Una introducción a Amazon Elastic Compute Cloud (EC2)

[Introducción a AmazonEC2: servidor y alojamiento de Elastic Cloud con AWS](#)

Amazon Elastic Compute Cloud (AmazonEC2) proporciona entornos de computación virtual altamente escalables y flexibles. Amazon EC2 le permite aprovisionar y gestionar servidores virtuales, conocidos como EC2 instancias de Amazon, para satisfacer una amplia gama de necesidades informáticas.

Las EC2 instancias de Amazon son máquinas virtuales que se pueden personalizar con diversas configuraciones de memoriaCPU, almacenamiento y capacidades de red. Puede elegir entre una amplia selección de tipos de instancias, que van desde opciones livianas y rentables hasta instancias potentes y de alto rendimiento, según los requisitos de su aplicación. Esta flexibilidad le permite adaptarse a sus necesidades informáticas para optimizar el rendimiento y la rentabilidad.

Además, Amazon EC2 ofrece un conjunto de funciones que le permiten gestionar sus recursos informáticos de forma eficaz. Estas incluyen la capacidad de lanzar nuevas instancias rápidamente,

crear imágenes de máquinas personalizadas (AMIs) para una implementación rápida y ampliar o reducir la capacidad informática según sea necesario.

Puedes acceder a las funciones de Amazon EC2 mediante AWS Command Line Interface (AWS CLI). Para ver una lista de los AWS CLI comandos de AmazonEC2, usa el siguiente comando.

```
aws ec2 help
```

Antes de ejecutar los comandos, defina sus credenciales predeterminadas. Para obtener más información, consulte [Configure el AWS CLI](#).

En este tema se muestran ejemplos abreviados de AWS CLI comandos que realizan tareas habituales en AmazonEC2.

Para ver ejemplos extensos de AWS CLI comandos, consulta el [repositorio de ejemplos AWS CLI de código](#) en GitHub.

## Temas

- [Crear, mostrar y eliminar pares de EC2 claves de Amazon en AWS CLI](#)
- [Crear, configurar y eliminar grupos de seguridad para Amazon EC2](#)
- [Lance, publique y cierre EC2 instancias de Amazon para AWS CLI](#)
- [Cambiar un tipo de instancia de Amazon EC2 con un script de Bash](#)

## Crear, mostrar y eliminar pares de EC2 claves de Amazon en AWS CLI

Puede usar AWS Command Line Interface (AWS CLI) para crear, mostrar y eliminar sus pares de claves para Amazon Elastic Compute Cloud (AmazonEC2). Usas pares de claves para conectarte a una EC2 instancia de Amazon.

Debes proporcionar el par de claves a Amazon EC2 al crear la instancia y, a continuación, usar ese par de claves para autenticarte cuando te conectes a la instancia.

### Note

Para ver ejemplos de comandos adicionales, consulta la [AWS CLI guía](#) de .

## Temas

- [Requisitos previos](#)

- [Crear un par de claves](#)
- [Mostrar un par de claves](#)
- [Eliminar un par de claves](#)
- [Referencias](#)

## Requisitos previos

Para ejecutar los comandos de `ec2`, necesitará:

- Instalar y configurar la AWS CLI. Para obtener más información, consulte [Instalar la AWS CLI](#) y [Credenciales de autenticación y acceso](#).
- Configura tus IAM permisos para permitir el EC2 acceso a Amazon. Para obtener más información sobre IAM los permisos de AmazonEC2, consulta [IAM las políticas de Amazon EC2](#) en la Guía del EC2 usuario de Amazon.

## Crear un par de claves

Para crear un par de claves, utilice el comando `aws ec2 create-key-pair` con la opción `--query` y la opción `--output text` para transferir su clave privada directamente a un archivo.

```
$ aws ec2 create-key-pair --key-name MyKeyPair --query 'KeyMaterial' --output text  
> MyKeyPair.pem
```

Para PowerShell ello, la `> file` redirección tiene por defecto la codificación UTF -8, que no se puede utilizar con algunos SSH clientes. Por lo tanto, debe convertir la salida transfiriéndola al comando `out-file` y establecer de forma explícita la codificación en `ascii`.

```
PS C:\>aws ec2 create-key-pair --key-name MyKeyPair --query 'KeyMaterial' --output text  
| out-file -encoding ascii -filepath MyKeyPair.pem
```

El archivo `MyKeyPair.pem` resultante tiene este aspecto.

```
-----BEGIN RSA PRIVATE KEY-----  
EXAMPLEKEYKCAQEAY7WZhaDsrA1W3mR1QtvhwyORRX8gnxgDAfRt/gx42kWXsT4rXE/b5CpSgie/  
vBoU7jLxx92pNHofnByP+Dc21eyyz6CvjTmWA0JwfWiW5/akH7i05dSrvC7dQkW2duV5QuUdE0QW  
Z/aNxMniGQE6XAgfwlnXVBwrerrQo+ZWQeqiUwwMkuEbLeJFLhMCvYURpUMSC1oehm449i1x9X1F  
G50TCFe0zf18dqqCP6GzbPaIjiU19xX/az0R9V+tpU0zEL+wmXnZt3/nHPQ5xvD20JH67km6SuPW  
oPzev/D8V+x4+bHthfSjR9Y7DvQFjFBVwHXigBdtZcU2/wei8D/HYwIDAQABAoIBAGZ1kaEvnrrqu
```

```

/uler7vgIn5m71N5LKw4hJLAIW6tUT/fzvtcHK0SkbQCQXuriHmQ2MQyJX/0kn2NfjLV/ufGxbL1
mb5qwMGUnEpJaZD6QSSs3kICLwWUYUiGfc0uiSbmJoap/GTLU0W5Mfcv36PaBUNy5p53V6G7hXb2
bahyWyJNfjLe4M86yd2YK3V2CmK+X/B0sShnJ36+hjrXPPWmV3N9zEmCdJjA+K15DYmhm/tJWSD9
81oGk9TopEp7CkIfatEATyyZiVqoRq6k64iuM9JkA30zdXzMQexXVJ1TLZVEH0E7bhLY9d801ozR
oQs/FiZNAx2iijCwyv01pjE73+kCgYEA9mZtyhkHkFDpwrSM1APaL8oNAbbjwEy7Z5Mqfq1+1Ip1
YkriL0DbLXLvRAH+yHPRit2hH0jtUNZh4Axv+cpq09qbUI3+43eEy24B7G/Uh+GTfbjsXs0xQx/x
p9otyVwc7hsQ5TA5PZb+mvkJ50BEKzet9XcKw0NBYELGhnEPe7cCgYEA06Vgov6YH1eHui9kHuws
ayav0elc5zKxjF9nfHFJRry21R1trw2Vdpn+9g481URrpzWV0Eihvm+xTtmaZlSp//lkq75XDwnU
WA8gkn603QE3fq2yN98BURsAKdJfJ5RL1HvGQvTe10HLYYXpJnEkHv+Unl2ajLivWUt5pbBrKbUC
gYBjb0+0Zk0sCcpZ29sbzjYjpIddErySIyRX5gV2uNQwAjLdp9PfN295yQ+BxMBXiIycWVQiw0bH
oMo7yykABY70zd5wQewBQ4AdS1WSX4nGDtsiFxiWiI5sKuAAe0CbTosy1s8w8fxoJ5Tz1sdoxNeGs
Arq6Wv/G16zQuAE9zK9vwwKBgF+09VI/1wJBirsDGz9whVwFFPrTkJNvJZzYt69qezx1sjgFKshy
WBhd4xHZtmCqpBP1AymEjr/T01bxyARmXMnIOWIANXMG4KGSy11mzSVAoQ+fqR+cJ3d0dyP11j
jjb0Ed/NY8frlNDxAVHE8BSkdsx2f6ELEyBKJSRr9snRAoGAMrTwYneXzvTskF/S5Fyu0i0egLDa
NWUH38v/nDCgEpIXD5Hn3qAEcju1IjmbwlvT+nY2jVhv7UGd8MjwUTNGItbdb6nsYqM2asrnF3qS
VRkAKKKYeGjKpUfVTTrW0YFjXkfcR/V+QFL50ndHAKJXjW7a4ejJLncTzmZSpYzwApc=
-----END RSA PRIVATE KEY-----

```

La clave privada no está almacenada en AWS y solo se puede recuperar cuando se crea. No puede recuperarla más adelante. En vez de ello, si pierde la clave privada, debe crear un par de claves nuevo.

Si se está conectando a su instancia desde un equipo Linux, le recomendamos que utilice el siguiente comando para establecer los permisos de su archivo de clave privada de modo que solo usted pueda leerlo.

```
$ chmod 400 MyKeyPair.pem
```

Mostrar un par de claves

Se genera una "huella digital" a partir del par de claves, que puede utilizar para verificar que la clave privada que tiene en su equipo local coincide con la clave pública almacenada en AWS.

La huella digital es un SHA1 hash que se extrae de una copia DER codificada de la clave privada. Este valor se captura cuando se crea el par de claves y se almacena en AWS junto con la clave pública. Puedes ver la huella digital en la EC2 consola de Amazon o ejecutando el AWS CLI comando [aws ec2 describe-key-pairs](#).

En el siguiente ejemplo se muestra la huella digital de MyKeyPair.

```
$ aws ec2 describe-key-pairs --key-name MyKeyPair
{
```

```
"KeyPairs": [  
  {  
    "KeyName": "MyKeyPair",  
    "KeyFingerprint":  
    "1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f"  
  }  
]  
}
```

Para obtener más información sobre las claves y las huellas dactilares, consulta [Amazon EC2 Key Pairs](#) en la Guía del EC2 usuario de Amazon.

### Eliminar un par de claves

Para eliminar un par de claves, ejecute el [aws ec2 delete-key-pair](#) comando substituyendo *MyKeyPair* con el nombre del par que se va a eliminar.

```
$ aws ec2 delete-key-pair --key-name MyKeyPair
```

### Referencias

AWS CLI referencia:

- [aws ec2](#)
- [aws ec2 create-key-pair](#)
- [aws ec2 delete-key-pair](#)
- [aws ec2 describe-key-pairs](#)

Otra referencia:

- [Documentación de Amazon Elastic Compute Cloud](#)
- Para ver ejemplos de AWS CLI código AWS SDK y contribuir a ellos, consulte el [Repositorio AWS de ejemplos de código](#) en GitHub.

## Crear, configurar y eliminar grupos de seguridad para Amazon EC2

Puede crear un grupo de seguridad para las instancias Amazon Elastic Compute Cloud (Amazon EC2) que básicamente funciona como un firewall, con reglas que determinan el tráfico de red que puede entrar y salir.

Use el AWS Command Line Interface (AWS CLI) para crear un grupo de seguridad, agregar reglas a los grupos de seguridad existentes y eliminar los grupos de seguridad.

### Note

Para ver ejemplos de comandos adicionales, consulte la [AWS CLI guía](#) de .

## Temas

- [Requisitos previos](#)
- [Creación de un grupo de seguridad](#)
- [Agregar reglas al grupo de seguridad](#)
- [Eliminación de un grupo de seguridad](#)
- [Referencias](#)

## Requisitos previos

Para ejecutar los comandos de ec2, necesitará:

- Instalar y configurar la AWS CLI. Para obtener más información, consulte [Instalar la AWS CLI](#) y [Credenciales de autenticación y acceso](#).
- Establezca sus permisos de IAM para permitir el acceso a Amazon EC2. Para obtener más información sobre los permisos de IAM para Amazon EC2, [consulte las políticas de IAM para Amazon EC2 en la Guía del usuario de Amazon](#) EC2.

## Creación de un grupo de seguridad

Puede crear grupos de seguridad asociados a nubes privadas virtuales (VPC).

En el siguiente ejemplo de [aws ec2 create-security-group](#) se muestra cómo crear un grupo de seguridad para una VPC especificada.

```
$ aws ec2 create-security-group --group-name my-sg --description "My security group" --  
vpc-id vpc-1a2b3c4d  
{  
  "GroupId": "sg-903004f8"  
}
```



Para ver la información inicial para un grupo de seguridad, ejecute el comando de [aws ec2 describe-security-groups](#). Solo puede hacer referencia a un grupo de seguridad de EC2-VPC por su `vpc-id`, no por su nombre.

```
$ aws ec2 describe-security-groups --group-ids sg-903004f8
{
  "SecurityGroups": [
    {
      "IpPermissionsEgress": [
        {
          "IpProtocol": "-1",
          "IpRanges": [
            {
              "CidrIp": "0.0.0.0/0"
            }
          ],
          "UserIdGroupPairs": []
        }
      ],
      "Description": "My security group"
      "IpPermissions": [],
      "GroupName": "my-sg",
      "VpcId": "vpc-1a2b3c4d",
      "OwnerId": "123456789012",
      "GroupId": "sg-903004f8"
    }
  ]
}
```

## Agregar reglas al grupo de seguridad

Cuando ejecute una instancia de Amazon EC2, debe habilitar reglas en el grupo de seguridad para permitir el tráfico de red entrante en el medio que utilice para conectarse a la imagen.

Por ejemplo, si lanza una instancia de Windows, normalmente debe añadir una regla para permitir el tráfico entrante en el puerto TCP 3389 (RDP) para dar soporte al protocolo de escritorio remoto (RDP). Si lanza una instancia de Linux, normalmente debe añadir una regla para permitir el tráfico entrante en el puerto TCP 22 (SSH) para dar soporte a las conexiones de SSH.

Use el comando [aws ec2 authorize-security-group-ingress](#) para añadir una regla a un grupo de seguridad. Uno de los parámetros obligatorios de este comando es la dirección IP pública

de su equipo, o la red (en forma de un intervalo de direcciones) al que su equipo está asociada, en notación [CIDR](#).

**Note**

Proporcionamos el siguiente servicio, <https://checkip.amazonaws.com/>, para determinar su dirección IP pública. Para encontrar otros servicios que puedan ayudarle a identificar su dirección IP, utilice el navegador para buscar "cuál es mi dirección IP". Si se conecta a través de un ISP o protegido por su firewall mediante una dirección IP dinámica (a través de una gateway NAT desde una red privada), su dirección puede cambiar de forma periódica. En ese caso, debe averiguar el rango de direcciones IP que utilizan los equipos cliente.

En el siguiente ejemplo se muestra cómo añadir una regla para RDP (puerto TCP 3389) a un grupo de seguridad EC2-VPC con el ID `sg-903004f8` utilizando su dirección IP.

Para empezar, busque su dirección IP.

```
$ curl https://checkip.amazonaws.com
x.x.x.x
```

Luego, puede agregar la dirección IP a su grupo de seguridad mediante la ejecución del comando [aws ec2 authorize-security-group-ingress](#).

```
$ aws ec2 authorize-security-group-ingress --group-id sg-903004f8 --protocol tcp --port 3389 --cidr x.x.x.x/x
```

El siguiente comando añade otra regla para habilitar SSH en instancias en el mismo grupo de seguridad.

```
$ aws ec2 authorize-security-group-ingress --group-id sg-903004f8 --protocol tcp --port 22 --cidr x.x.x.x/x
```

Para ver los cambios realizados en el grupo de seguridad, ejecute el comando [aws ec2 describe-security-groups](#).

```
$ aws ec2 describe-security-groups --group-ids sg-903004f8
{
  "SecurityGroups": [
```

```

    {
      "IpPermissionsEgress": [
        {
          "IpProtocol": "-1",
          "IpRanges": [
            {
              "CidrIp": "0.0.0.0/0"
            }
          ],
          "UserIdGroupPairs": []
        }
      ],
      "Description": "My security group"
    },
    {
      "IpPermissions": [
        {
          "ToPort": 22,
          "IpProtocol": "tcp",
          "IpRanges": [
            {
              "CidrIp": "x.x.x.x/x"
            }
          ],
          "UserIdGroupPairs": [],
          "FromPort": 22
        }
      ],
      "GroupName": "my-sg",
      "OwnerId": "123456789012",
      "GroupId": "sg-903004f8"
    }
  ]
}

```

## Eliminación de un grupo de seguridad

Para eliminar un grupo de seguridad, ejecute el comando [aws ec2 delete-security-group](#).

### Note

No puede eliminar un grupo de seguridad que esté conectado actualmente a un entorno.

El siguiente ejemplo de comando elimina un grupo de seguridad EC2-VPC.

```
$ aws ec2 delete-security-group --group-id sg-903004f8
```

## Referencias

AWS CLI referencia:

- [aws ec2](#)
- [aws ec2 authorize-security-group-ingress](#)
- [aws ec2 create-security-group](#)
- [aws ec2 delete-security-group](#)
- [aws ec2 describe-security-groups](#)

Otra referencia:

- [Documentación de Amazon Elastic Compute Cloud](#)
- Para ver ejemplos de AWS CLI código y AWS del SDK y contribuir a ellos, consulta el [repositorio AWS de ejemplos de código](#) en GitHub.

## Lance, publique y cierre EC2 instancias de Amazon para AWS CLI

Puede usar AWS Command Line Interface (AWS CLI) para lanzar, enumerar y finalizar instancias de Amazon Elastic Compute Cloud (AmazonEC2). Si lanza una instancia que no está incluida en la capa AWS gratuita, se le facturará después del lanzamiento de la instancia y se le cobrará por el tiempo que la instancia esté en ejecución, incluso si permanece inactiva.

### Note

Para ver ejemplos de comandos adicionales, consulta la guía de [AWS CLI referencia de la guía](#) de .

## Temas

- [Requisitos previos](#)
- [iniciar la instancia](#)
- [Agregar un dispositivo de bloques a la instancia](#)
- [Agregar una etiqueta a la instancia](#)

- [Conéctese a su instancia](#)
- [Enumerar las instancias](#)
- [Terminar una instancia](#)
- [Referencias](#)

## Requisitos previos

Para ejecutar los comandos `ec2` en este tema, debe hacer lo siguiente:

- Instalar y configurar la AWS CLI. Para obtener más información, consulte [Instalar la AWS CLI](#) y [Credenciales de autenticación y acceso](#).
- Configura tus IAM permisos para permitir el EC2 acceso a Amazon. Para obtener más información sobre IAM los permisos de AmazonEC2, consulta [IAM las políticas de Amazon EC2](#) en la Guía del EC2 usuario de Amazon.
- Cree un [par de claves](#) y un [grupo de seguridad](#).
- Selecciona una imagen de máquina de Amazon (AMI) y anota el AMI ID. Para obtener más información, consulta [Cómo encontrar un producto adecuado AMI](#) en la Guía del EC2 usuario de Amazon.

## iniciar la instancia

Para lanzar una EC2 instancia de Amazon con la AMI que seleccionaste, usa el [aws ec2 run-instances](#) comando. Puede lanzar la instancia en una nube privada virtual (VPC).

Al principio, la instancia aparece en el estado `pending`, pero cambia al estado `running` después de unos minutos.

En el siguiente ejemplo, se muestra cómo lanzar una `t2.micro` instancia en la subred especificada de un VPC. Sustituya el *italicized* los valores de los parámetros por los suyos propios.

```
$ aws ec2 run-instances --image-id ami-xxxxxxx --count 1 --instance-type t2.micro --  
key-name MyKeyPair --security-group-ids sg-903004f8 --subnet-id subnet-6e7f829e  
{  
  "OwnerId": "123456789012",  
  "ReservationId": "r-5875ca20",  
  "Groups": [  
    {  
      "GroupName": "my-sg",
```

```
    "GroupId": "sg-903004f8"
  }
],
"Instances": [
  {
    "Monitoring": {
      "State": "disabled"
    },
    "PublicDnsName": null,
    "Platform": "windows",
    "State": {
      "Code": 0,
      "Name": "pending"
    },
    "EbsOptimized": false,
    "LaunchTime": "2013-07-19T02:42:39.000Z",
    "PrivateIpAddress": "10.0.1.114",
    "ProductCodes": [],
    "VpcId": "vpc-1a2b3c4d",
    "InstanceId": "i-5203422c",
    "ImageId": "ami-173d747e",
    "PrivateDnsName": "ip-10-0-1-114.ec2.internal",
    "KeyName": "MyKeyPair",
    "SecurityGroups": [
      {
        "GroupName": "my-sg",
        "GroupId": "sg-903004f8"
      }
    ],
    "ClientToken": null,
    "SubnetId": "subnet-6e7f829e",
    "InstanceType": "t2.micro",
    "NetworkInterfaces": [
      {
        "Status": "in-use",
        "SourceDestCheck": true,
        "VpcId": "vpc-1a2b3c4d",
        "Description": "Primary network interface",
        "NetworkInterfaceId": "eni-a7edb1c9",
        "PrivateIpAddresses": [
          {
            "PrivateDnsName": "ip-10-0-1-114.ec2.internal",
            "Primary": true,
            "PrivateIpAddress": "10.0.1.114"
          }
        ]
      }
    ]
  }
]
```

```
    }
  ],
  "PrivateDnsName": "ip-10-0-1-114.ec2.internal",
  "Attachment": {
    "Status": "attached",
    "DeviceIndex": 0,
    "DeleteOnTermination": true,
    "AttachmentId": "eni-attach-52193138",
    "AttachTime": "2013-07-19T02:42:39.000Z"
  },
  "Groups": [
    {
      "GroupName": "my-sg",
      "GroupId": "sg-903004f8"
    }
  ],
  "SubnetId": "subnet-6e7f829e",
  "OwnerId": "123456789012",
  "PrivateIpAddress": "10.0.1.114"
}
],
"SourceDestCheck": true,
"Placement": {
  "Tenancy": "default",
  "GroupName": null,
  "AvailabilityZone": "us-west-2b"
},
"Hypervisor": "xen",
"BlockDeviceMappings": [
  {
    "DeviceName": "/dev/sda1",
    "Ebs": {
      "Status": "attached",
      "DeleteOnTermination": true,
      "VolumeId": "vol-877166c8",
      "AttachTime": "2013-07-19T02:42:39.000Z"
    }
  }
],
"Architecture": "x86_64",
"StateReason": {
  "Message": "pending",
  "Code": "pending"
},
```

```

    "RootDeviceName": "/dev/sda1",
    "VirtualizationType": "hvm",
    "RootDeviceType": "ebs",
    "Tags": [
      {
        "Value": "MyInstance",
        "Key": "Name"
      }
    ],
    "AmiLaunchIndex": 0
  }
]
}

```

### Agregar un dispositivo de bloques a la instancia

Cada instancia que lance tiene un volumen de dispositivo raíz asociado. Puedes usar el mapeo de dispositivos de bloques para especificar volúmenes adicionales de Amazon Elastic Block Store (AmazonEBS) o volúmenes de almacenes de instancias para adjuntarlos a una instancia cuando se lance.

Para añadir un dispositivo de bloque a la instancia, especifique la opción `--block-device-mappings` cuando utilice `run-instances`.

El siguiente parámetro de ejemplo proporciona un EBS volumen de Amazon estándar de 20 GB y lo asigna a la instancia mediante el identificador `/dev/sdf`.

```
--block-device-mappings "[{"DeviceName":"/dev/sdf","Ebs":{"VolumeSize":20,
"DeleteOnTermination":false}]"]
```

En el siguiente ejemplo, se añade un EBS volumen de Amazon, asignado a `/dev/sdf`, en función de una instantánea existente. Una instantánea representa una imagen que se carga en el volumen por usted. Cuando se especifica una instantánea, no es necesario especificar un tamaño de volumen; será lo suficientemente grande como para almacenar la imagen. Sin embargo, si especifica un tamaño, este deberá ser igual o mayor que el tamaño de la instantánea.

```
--block-device-mappings [{"DeviceName":"/dev/sdf","Ebs":{"SnapshotId":"snap-
a1b2c3d4"}}]"]
```

En el siguiente ejemplo se añaden dos volúmenes a la instancia. El número de volúmenes disponibles para su instancia depende del tipo de instancia.



```
--block-device-mappings "[{\"DeviceName\":\"/dev/sdf\",\"VirtualName\":\"ephemeral0\"},  
{\"DeviceName\":\"/dev/sdg\",\"VirtualName\":\"ephemeral1\"}]"
```

En el siguiente ejemplo se crea el mapeo (/dev/sdj), pero no se aprovisiona un volumen para la instancia.

```
--block-device-mappings "[{\"DeviceName\":\"/dev/sdj\",\"NoDevice\":\"\"}]"
```

Para obtener más información, consulta [Block Device Mapping](#) en la Guía del EC2 usuario de Amazon.

### Agregar una etiqueta a la instancia

Una etiqueta es una etiqueta que se asigna a un AWS recurso. Le permite añadir metadatos a los recursos que puede utilizar para distintos fines. Para obtener más información, consulta [Cómo etiquetar tus recursos](#) en la Guía del EC2 usuario de Amazon.

En el siguiente ejemplo se muestra cómo añadir una etiqueta con el nombre de clave "Name" y el valor "MyInstance" a la instancia especificada, mediante el comando [aws ec2 create-tags](#).

```
$ aws ec2 create-tags --resources i-5203422c --tags Key=Name,Value=MyInstance
```

### Conéctese a su instancia

Puede conectarse a su instancia mientras se esté ejecutando y utilizarla como si fuera un equipo que tiene delante. Para obtener más información, consulte [Connect to Your Amazon EC2 Instance](#) en la Guía del EC2 usuario de Amazon.

### Enumerar las instancias

Puede utilizarla AWS CLI para enumerar sus instancias y ver información sobre ellas. Puede incluir todas sus instancias en la lista o filtrar los resultados en función de las instancias que le interesen.

Los siguientes ejemplos muestran cómo utilizar el comando [aws ec2 describe-instances](#).

El siguiente comando muestra todas sus instancias.

```
$ aws ec2 describe-instances
```

El siguiente comando filtra únicamente las instancias `t2.micro` de la lista y devuelve únicamente los valores `InstanceId` para cada coincidencia.

```
$ aws ec2 describe-instances --filters "Name=instance-type,Values=t2.micro" --query
"Reservations[].Instances[].InstanceId"
[
  "i-05e998023d9c69f9a"
]
```

El siguiente comando muestra todas las instancias que tienen la etiqueta `Name=MyInstance`.

```
$ aws ec2 describe-instances --filters "Name=tag:Name,Values=MyInstance"
```

El siguiente comando muestra las instancias que se lanzaron con alguno de los siguientes comandos AMIs: `ami-x0123456`, `ami-y0123456`, `yami-z0123456`.

```
$ aws ec2 describe-instances --filters "Name=image-id,Values=ami-x0123456,ami-
y0123456,ami-z0123456"
```

## Terminar una instancia

Al terminar una instancia, esta se elimina. Una vez se ha terminado la instancia, no es posible volver a conectarse a ella.

En cuanto el estado de la instancia cambie a `shutting-down` o a `terminated`, dejará de incurrir en costos por ella. Si desea volver a conectarse a una instancia más adelante, utilice [stop-instances](#) en lugar de `terminate-instances`. Para obtener más información, consulte [Finalizar su instancia](#) en la Guía del EC2 usuario de Amazon.

Para eliminar una instancia, utilice el comando [aws ec2 terminate-instances](#) para eliminarla.

```
$ aws ec2 terminate-instances --instance-ids i-5203422c
{
  "TerminatingInstances": [
    {
      "InstanceId": "i-5203422c",
      "CurrentState": {
        "Code": 32,
        "Name": "shutting-down"
      },
      "PreviousState": {
```

```
        "Code": 16,  
        "Name": "running"  
    }  
  ]  
}
```

## Referencias

AWS CLI referencia:


- [aws ec2](#)
- [aws ec2 create-tags](#)
- [aws ec2 describe-instances](#)
- [aws ec2 run-instances](#)
- [aws ec2 terminate-instances](#)

Otra referencia:

- [Documentación de Amazon Elastic Compute Cloud](#)
- Para ver ejemplos de AWS CLI código AWS SDK y contribuir a ellos, consulte el [Repositorio AWS de ejemplos de código](#) en GitHub.

## Cambiar un tipo de instancia de Amazon EC2 con un script de Bash

En este ejemplo de secuencias de comandos bash para Amazon EC2 se cambia el tipo de instancia de una instancia de Amazon EC2 mediante (). AWS Command Line Interface AWS CLI Detiene la instancia si se está ejecutando, cambia el tipo de instancia y, a continuación, si se solicita, reinicia la instancia. Los scripts de shell son programas diseñados para ejecutarse en una interfaz de línea de comandos.

 Note

## Temas

- [Antes de comenzar](#)

- [Acerca de este ejemplo](#)
- [Parámetros](#)
- [Archivos](#)
- [Referencias](#)

Antes de comenzar

Antes de que pueda ejecutar cualquiera de los siguientes ejemplos, se debe completar lo siguiente.

- Instalar y configurar la AWS CLI. Para más información, consulte [Instalar la AWS CLI](#) y [Credenciales de autenticación y acceso](#).
- El perfil que utilice debe tener permisos que permitan realizar las AWS operaciones de los ejemplos.
- Una instancia de Amazon EC2 en ejecución en la cuenta para la que tiene permiso para detener y modificar. Si ejecuta el script de prueba, inicia una instancia, prueba el cambio del tipo y, a continuación, termina la instancia.
- Como práctica AWS recomendada, conceda a este código los privilegios mínimos o solo los permisos necesarios para realizar una tarea. Para obtener más información, consulte [Otorgar privilegios mínimos](#) en la Guía del usuario de AWS Identity and Access Management (IAM).
- Este código no se ha probado en todas AWS las regiones. Algunos AWS servicios solo están disponibles en regiones específicas. Para obtener más información, consulte [Puntos de enlace de servicio y cuotas](#) en la Guía de referencia general de AWS .
- La ejecución de este código puede generar cargos en su AWS cuenta. Es su responsabilidad asegurarse de que los recursos creados por este script se eliminen cuando haya terminado con ellos.

Acerca de este ejemplo

Este ejemplo se escribe como una función en el archivo de script de shell `change_ec2_instance_type.sh` que puede `source` desde otro script o desde la línea de comandos. Cada archivo de script contiene comentarios que describen cada una de las funciones. Una vez la función esté en la memoria, puede invocarla desde la línea de comandos. Por ejemplo, los siguientes comandos cambian el tipo de instancia especificada a `t2.nano`:

```
$ source ./change_ec2_instance_type.sh
```

```
$ ./change_ec2_instance_type -i *instance-id* -t new-type
```

Para ver el ejemplo completo y los archivos de script descargables, consulte [Cambiar el tipo de instancia de Amazon EC2](#) en el repositorio de ejemplos de AWS código en GitHub

## Parámetros

**-i**(string) Especifica el ID de instancia que se va a modificar.

**-t**(string)Especifica el tipo de instancia de Amazon EC2 al que se va a cambiar.

**-r**(switch) De forma predeterminada, no se establece. Si se configura **-r**, reinicia la instancia después del modificador de tipo.

**-f**(switch) De forma predeterminada, el script solicita al usuario que confirme el cierre de la instancia antes de realizar el cambio. Si se configura **-f**, la función no le pide al usuario antes de apagar la instancia para hacer el cambio de tipo

**-v**(switch) De forma predeterminada, el script funciona silenciosamente y muestra la salida solo en caso de error. Si se configura **-v**, la función muestra el estado a lo largo de su operación.

## Archivos

### **change\_ec2\_instance\_type.sh**

El archivo de script principal contiene la función `change_ec2_instance_type()` que realiza las siguientes tareas:

- Comprueba que existe la instancia de Amazon EC2 especificada.
- A menos que se seleccione **-f**, advierte al usuario antes de detener la instancia.
- Cambia el tipo de instancia
- Si configura **-r**, reinicia la instancia y confirma que la instancia se está ejecutando

Vea el código de [change\\_ec2\\_instance\\_type.sh](#) on GitHub

### **test\_change\_ec2\_instance\_type.sh**

El script del archivo `test_change_ec2_instance_type.sh` prueba las diversas rutas de código para la función de `change_ec2_instance_type`. Si todos los pasos del script de prueba funcionan correctamente, este elimina todos los recursos que ha creado.

Puede ejecutar el script de prueba con los siguientes parámetros:

- `-v`-(switch) Cada prueba muestra un estado de superación/fallo a medida que se ejecutan. De forma predeterminada, las pruebas se ejecutan silenciosamente y la salida incluye solo el estado final de aprobado o error.
- `-i`-(switch) El script se detiene después de cada prueba para permitirle examinar los resultados intermedios de cada paso. Le permite examinar el estado actual de la instancia mediante la consola de Amazon EC2. El script pasa al siguiente paso después de presionar ENTER (INTRO) en el símbolo del sistema.

Ver el código de [test\\_change\\_ec2\\_instance\\_type.sh](#) on GitHub.

## **awsdocs\_general.sh**

El archivo de script `awsdocs_general.sh` contiene funciones de uso general utilizadas en ejemplos avanzados para el AWS CLI.

Ver el código de [awsdocs\\_general.sh](#) on GitHub.

## Referencias

AWS CLI referencia:

- [aws ec2](#)
- [aws ec2 describe-instances](#)
- [aws ec2 modify-instance-attribute](#)
- [aws ec2 start-instances](#)
- [aws ec2 stop-instances](#)
- [aws ec2 wait instance-running](#)
- [aws ec2 wait instance-stopped](#)

Otra referencia:

- [Documentación de Amazon Elastic Compute Cloud](#)
- Para ver ejemplos de AWS CLI código y AWS del SDK y contribuir a ellos, consulta el [repositorio AWS de ejemplos de código](#) en GitHub.

# Utilice Amazon S3 Glacier con AWS CLI

Una introducción a Amazon S3 Glacier

[Introducción a Amazon S3 Glacier](#)

En este tema se muestran ejemplos de AWS CLI comandos que realizan tareas comunes en S3 Glacier. Los ejemplos muestran cómo usarlo AWS CLI para cargar un archivo grande en S3 Glacier dividiéndolo en partes más pequeñas y cargándolas desde la línea de comandos.

Puede acceder a las funciones de Amazon S3 Glacier mediante AWS Command Line Interface (AWS CLI). Para ver una lista de los AWS CLI comandos de S3 Glacier, utilice el siguiente comando.

```
aws glacier help
```

## Note

Para obtener más ejemplos de referencia de comandos y ver otros ejemplos, consulte [aws glacier](#) en la Referencia de los comandos de AWS CLI .

## Temas

- [Requisitos previos](#)
- [Cree un almacén Amazon S3 Glacier](#)
- [Preparación de un archivo para cargarlo](#)
- [Inicio de una carga multiparte y carga de archivos](#)
- [Finalización de la carga](#)
- [Recursos](#)

## Requisitos previos

Para ejecutar los comandos de `glacier`, necesitará:

- Instalar y configurar la AWS CLI. Para obtener más información, consulte [Instalar la AWS CLI y Credenciales de autenticación y acceso](#).

- Este tutorial utiliza varias herramientas de línea de comandos que normalmente vienen preinstaladas en los sistemas operativos tipo Unix, como Linux y macOS. Los usuarios de Windows pueden usar las mismas herramientas instalando [Cygwin](#) y ejecutando los comandos desde el terminal de Cygwin. Se indicará en cada caso si hay disponibles utilidades y comandos nativos de Windows que realicen las mismas funciones.

## Cree un almacén Amazon S3 Glacier

Cree un almacén con el comando [create-vault](#).

```
$ aws glacier create-vault --account-id - --vault-name myvault
{
  "location": "/123456789012/vaults/myvault"
}
```

### Note

Todos los comandos de S3 Glacier requieren un parámetro de ID de cuenta. Utilice el carácter de guion (`--account-id -`) para utilizar la cuenta actual.

## Preparación de un archivo para cargarlo

Cree un archivo para la carga de prueba. Los siguientes comandos crean un archivo llamado *largefile* que contiene exactamente 3 MiB de datos aleatorios.

Linux o macOS

```
$ dd if=/dev/urandom of=largefile bs=3145728 count=1
1+0 records in
1+0 records out
3145728 bytes (3.1 MB) copied, 0.205813 s, 15.3 MB/s
```

`dd` es una utilidad que copia un número de bytes de un archivo de entrada en un archivo de salida. En el ejemplo anterior se usa el archivo de dispositivo de sistema `/dev/urandom` como origen de datos aleatorios. `fsutil` realiza una función similar en Windows.

Windows

```
C:\> fsutil file createnew largefile 3145728
```



```
File C:\temp\largefile is created
```

A continuación, divida el archivo en fragmentos de 1 MiB (1.048.576 bytes) con un divisor de archivos.

```
$ split -b 1048576 --verbose largefile chunk
creating file `chunkaa'
creating file `chunkab'
creating file `chunkac'
```

## Inicio de una carga multiparte y carga de archivos

Cree una carga multiparte en Amazon S3 Glacier con el comando [initiate-multipart-upload](#).

```
$ aws glacier initiate-multipart-upload --account-id - --archive-description "multipart
upload test" --part-size 1048576 --vault-name myvault
{
  "uploadId": "19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
0ssZtLqyFu7sY1_1R7vgFuJV6NtcV5zpsJ",
  "location": "/123456789012/vaults/myvault/multipart-
uploads/19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
0ssZtLqyFu7sY1_1R7vgFuJV6NtcV5zpsJ"
}
```

S3 Glacier requiere especificar el tamaño de cada parte en bytes (1 MiB en este ejemplo), el nombre del almacén y el ID de cuenta para configurar la carga multiparte. Genera un ID AWS CLI de carga cuando se completa la operación. Guarde el ID de carga en una variable de shell para su uso posterior.

### Linux o macOS

```
$ UPLOADID="19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
0ssZtLqyFu7sY1_1R7vgFuJV6NtcV5zpsJ"
```

### Windows

```
C:\> set UPLOADID="19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
0ssZtLqyFu7sY1_1R7vgFuJV6NtcV5zpsJ"
```

A continuación, use el comando [upload-multipart-part](#) para cargar cada una de las tres partes.

```

$ aws glacier upload-multipart-part --upload-id $UPLOADID --body chunkaa --range 'bytes
0-1048575/*' --account-id - --vault-name myvault
{
  "checksum": "e1f2a7cd6e047fa606fe2f0280350f69b9f8cfa602097a9a026360a7edc1f553"
}
$ aws glacier upload-multipart-part --upload-id $UPLOADID --body chunkab --range 'bytes
1048576-2097151/*' --account-id - --vault-name myvault
{
  "checksum": "e1f2a7cd6e047fa606fe2f0280350f69b9f8cfa602097a9a026360a7edc1f553"
}
$ aws glacier upload-multipart-part --upload-id $UPLOADID --body chunkac --range 'bytes
2097152-3145727/*' --account-id - --vault-name myvault
{
  "checksum": "e1f2a7cd6e047fa606fe2f0280350f69b9f8cfa602097a9a026360a7edc1f553"
}

```

### Note

En el ejemplo anterior se usa el signo de dólar (\$) para hacer referencia al contenido de la variable de shell UPLOADID en Linux. En la línea de comandos de Windows, inserte un signo de porcentaje (%) a ambos lados del nombre de la variable (por ejemplo, %UPLOADID%).

Debe especificar el intervalo de bytes de cada parte al cargarla, para que S3 Glacier pueda volver a ensamblarlas en el orden adecuado. Cada parte tiene 1048576 bytes, por lo que la primera ocupa los bytes del 0 al 1048575, la segunda del 1048576 al 2097151 y la tercera del 2097152 al 3145727.

## Finalización de la carga

Amazon S3 Glacier requiere un hash de árbol del archivo original para confirmar que todas las piezas cargadas están AWS intactas.

Para calcular un hash de árbol, debes dividir el archivo en partes de 1 MiB y calcular un hash binario de SHA-256 para cada pieza. A continuación, divide la lista de hashes en pares, combine los dos hashes binarios de cada par y tome los hashes de los resultados. Repita este proceso hasta que solo quede un hash. Si hay un número impar de hashes en cualquier nivel, páselo al nivel siguiente sin modificarlo.

La clave para calcular un algoritmo hash en árbol correctamente al usar utilidades de línea de comandos es almacenar cada hash en formato binario y convertir a hexadecimal solamente en

el último paso. Si la versión hexadecimal de cualquier hash del árbol se combina o se convierte a algoritmo hash, se obtendrá un resultado incorrecto.

### Note

Los usuarios de Windows pueden utilizar el comando `type` en lugar de `cat`. Open SSL está disponible para Windows en [SSLOpen.org](https://SSLOpen.org).

Para calcular un algoritmo hash en árbol

1. Divida el archivo original en partes de 1 MiB, si no lo ha hecho aún.

```
$ split --bytes=1048576 --verbose largefile chunk
creating file `chunkaa'
creating file `chunkab'
creating file `chunkac'
```

2. Calcule y almacene el hash binario SHA -256 de cada fragmento.

```
$ openssl dgst -sha256 -binary chunkaa > hash1
$ openssl dgst -sha256 -binary chunkab > hash2
$ openssl dgst -sha256 -binary chunkac > hash3
```

3. Combine los dos primeros hashes y tome el hash binario del resultado.

```
$ cat hash1 hash2 > hash12
$ openssl dgst -sha256 -binary hash12 > hash12hash
```

4. Combine el hash principal de los fragmentos aa y ab con el hash del fragmento ac y cree un hash del resultado, esta vez con formato hexadecimal. Almacene el resultado en una variable de shell.

```
$ cat hash12hash hash3 > hash123
$ openssl dgst -sha256 hash123
SHA256(hash123)= 9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67
$ TREEHASH=9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67
```

Por último, complete la carga con el comando [complete-multipart-upload](#), que toma el tamaño del archivo original en bytes, el valor final del algoritmo hash en árbol en formato hexadecimal, su ID de cuenta y el nombre del almacén.

```
$ aws glacier complete-multipart-upload --checksum $TREEHASH --archive-size 3145728 --
upload-id $UPLOADID --account-id - --vault-name myvault
{
  "archiveId": "d3AbWhE0YE1m6f_fI1jPG82F8xzbMEEZmrALLGAA0NJAzo5QdP-
N83MKqd96Unspoa5H51ItWX-sK8-QS0ZhwsyGiu9-R-
kwWUyS1dSB1mgPPWkEbeFfqDSav053rU7FvVLHfRc6hg",
  "checksum": "9628195fcdbcbbe76cde932d4646fa7de5f219fb39823836d81f0cc0e18aa67",
  "location": "/123456789012/vaults/myvault/archives/
d3AbWhE0YE1m6f_fI1jPG82F8xzbMEEZmrALLGAA0NJAzo5QdP-N83MKqd96Unspoa5H51ItWX-sK8-
QS0ZhwsyGiu9-R-kwWUyS1dSB1mgPPWkEbeFfqDSav053rU7FvVLHfRc6hg"
}
```

También puede comprobar el estado del almacén con el comando [describe-vault](#).

```
$ aws glacier describe-vault --account-id - --vault-name myvault
{
  "SizeInBytes": 3178496,
  "VaultARN": "arn:aws:glacier:us-west-2:123456789012:vaults/myvault",
  "LastInventoryDate": "2018-12-07T00:26:19.028Z",
  "NumberOfArchives": 1,
  "CreationDate": "2018-12-06T21:23:45.708Z",
  "VaultName": "myvault"
}
```

#### Note

El estado del almacén se actualiza una vez al día. Consulte [Trabajar con almacenes](#) para obtener más información.

Ahora puede quitar sin problemas el fragmento y los archivos hash que ha creado.

```
$ rm chunk* hash*
```

Para obtener más información sobre las cargas multiparte, consulte las secciones [Carga de archivos grandes por partes](#) y [Cálculo de sumas de comprobación](#) en la Guía para desarrolladores de Amazon S3 Glacier.

## Recursos

AWS CLI referencia:

- [aws glacier](#)
- [aws glacier complete-multipart-upload](#)
- [aws glacier create-vault](#)
- [aws glacier describe-vault](#)
- [aws glacier initiate-multipart-upload](#)

Referencia de servicio:

- [Guía para desarrolladores de Amazon S3 Glacier](#)
- [Carga de archivos grandes por partes](#) en la Guía para desarrolladores de Amazon S3 Glacier
- [Cálculo de sumas de comprobación](#) en la Guía para desarrolladores de Amazon S3 Glacier
- [Uso de almacenes](#) en la Guía para desarrolladores de Amazon S3 Glacier

## AWS Identity and Access Management Úselo desde AWS CLI

Una introducción a AWS Identity and Access Management

[Introducción a AWS Identity and Access Management](#)

Puede acceder a las funciones de AWS Identity and Access Management (IAM) mediante AWS Command Line Interface (AWS CLI). Para ver una lista de los AWS CLI comandos IAM, utilice el siguiente comando.

```
aws iam help
```

En este tema se muestran ejemplos de AWS CLI comandos que realizan tareas comunes para IAM.

Antes de ejecutar los comandos, defina sus credenciales predeterminadas. Para obtener más información, consulte [Configure el AWS CLI](#).

Para obtener más información sobre el IAM servicio, consulte la [Guía del AWS Identity and Access Management usuario](#).

## Temas

- [Crear usuarios y grupos de IAM](#)
- [Adjunte una política IAM gestionada a un usuario](#)
- [Establezca una contraseña inicial para un IAM usuario](#)
- [Cree una clave de acceso para un IAM usuario](#)

## Crear usuarios y grupos de IAM

Para crear un grupo y añadirle un usuario

1. Utilice el comando [create-group](#) para crear el grupo.

```
$ aws iam create-group --group-name MyIamGroup
{
  "Group": {
    "GroupName": "MyIamGroup",
    "CreateDate": "2018-12-14T03:03:52.834Z",
    "GroupId": "AGPAJNUJ2W4IJVEXAMPLE",
    "Arn": "arn:aws:iam::123456789012:group/MyIamGroup",
    "Path": "/"
  }
}
```

2. Use el comando [create-user](#) para crear el usuario.

```
$ aws iam create-user --user-name MyUser
{
  "User": {
    "UserName": "MyUser",
    "Path": "/",
    "CreateDate": "2018-12-14T03:13:02.581Z",
    "UserId": "AIDAJY2PE5XUZ4EXAMPLE",
    "Arn": "arn:aws:iam::123456789012:user/MyUser"
  }
}
```

```
}

```

- Use el comando [add-user-to-group](#) para agregar el usuario al grupo.

```
$ aws iam add-user-to-group --user-name MyUser --group-name MyIamGroup
```

- Para verificar que el grupo MyIamGroup contiene MyUser, use el comando [get-group](#).

```
$ aws iam get-group --group-name MyIamGroup
{
  "Group": {
    "GroupName": "MyIamGroup",
    "CreateDate": "2018-12-14T03:03:52Z",
    "GroupId": "AGPAJNUJ2W4IJVEXAMPLE",
    "Arn": "arn:aws:iam::123456789012:group/MyIamGroup",
    "Path": "/"
  },
  "Users": [
    {
      "UserName": "MyUser",
      "Path": "/",
      "CreateDate": "2018-12-14T03:13:02Z",
      "UserId": "AIDAJY2PE5XUZ4EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:user/MyUser"
    }
  ],
  "IsTruncated": "false"
}
```

## Adjunte una política IAM gestionada a un usuario

La política de este ejemplo ofrece al usuario "Power User Access" (acceso avanzado).

Para adjuntar una política IAM administrada a un usuario

- Determine el nombre del recurso de Amazon (ARN) de la política que se va a adjuntar. El siguiente comando `list-policies` se utiliza para buscar ARN la política con ese nombre `PowerUserAccess`. Luego lo almacena ARN en una variable de entorno.

```
$ export POLICYARN=$(aws iam list-policies --query 'Policies[?
PolicyName==`PowerUserAccess`].{ARN:Arn}' --output text) ~
```

```
$ echo $POLICYARN
arn:aws:iam::aws:policy/PowerUserAccess
```

2. Para adjuntar la política, utilice el [attach-user-policy](#) comando y haga referencia a la variable de entorno que contiene la políticaARN.

```
$ aws iam attach-user-policy --user-name MyUser --policy-arn $POLICYARN
```

3. Verifique que la política se ha adjuntado al usuario ejecutando el comando [list-attached-user-policies](#).

```
$ aws iam list-attached-user-policies --user-name MyUser
{
  "AttachedPolicies": [
    {
      "PolicyName": "PowerUserAccess",
      "PolicyArn": "arn:aws:iam::aws:policy/PowerUserAccess"
    }
  ]
}
```

Para obtener más información, consulte [Recursos de administración de acceso](#). En este tema se proporcionan enlaces a una descripción general de los permisos y las políticas, y enlaces a ejemplos de políticas para acceder a Amazon S3EC2, Amazon y otros servicios.

## Establezca una contraseña inicial para un IAM usuario

El comando siguiente utiliza [create-login-profile](#) para configurar una contraseña inicial en el usuario especificado. Cuando el usuario inicia sesión por primera vez, se le solicitará que cambie la contraseña y elija una que solo él conozca.

```
$ aws iam create-login-profile --user-name MyUser --password My!User1Login8P@ssword --
password-reset-required
{
  "LoginProfile": {
    "UserName": "MyUser",
    "CreateDate": "2018-12-14T17:27:18Z",
    "PasswordResetRequired": true
  }
}
```



Puede usar el comando `update-login-profile` para cambiar la contraseña de un usuario.

```
$ aws iam update-login-profile --user-name MyUser --password My!User1ADifferentP@ssword
```

## Cree una clave de acceso para un IAM usuario

Puede utilizar el comando [create-access-key](#) para crear una clave de acceso para un usuario. Una clave de acceso es un conjunto de credenciales de seguridad que se compone de un ID de clave de acceso y una clave secreta.

Un usuario solo puede crear dos claves de acceso al mismo tiempo. Si intenta crear un tercer conjunto, el comando devuelve un error `LimitExceeded`.

```
$ aws iam create-access-key --user-name MyUser
{
  "AccessKey": {
    "UserName": "MyUser",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "Status": "Active",
    "SecretAccessKey": "wJa1rXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY",
    "CreateDate": "2018-12-14T17:34:16Z"
  }
}
```

Utilice el comando [delete-access-key](#) para eliminar una clave de acceso para un usuario. Especifique la clave de acceso que quiera eliminar mediante el ID de clave de acceso.

```
$ aws iam delete-access-key --user-name MyUser --access-key-id AKIAIOSFODNN7EXAMPLE
```

## Utilice Amazon S3 con AWS CLI

Una introducción a Amazon Simple Storage Service (Amazon S3)

[Introducción a Amazon Simple Storage Service \(Amazon S3\): almacenamiento en la nube en AWS](#)

Puede acceder a las funciones de Amazon Simple Storage Service (Amazon S3) mediante AWS Command Line Interface (). AWS CLI Amazon S3 es un servicio de almacenamiento de

objetos duradero y altamente escalable. Amazon S3 está diseñado para ofrecer una capacidad de almacenamiento prácticamente ilimitada, lo que lo convierte en una solución ideal para una amplia gama de necesidades de almacenamiento y administración de datos.

Amazon S3 le permite almacenar y recuperar cualquier cantidad de datos, desde archivos pequeños hasta conjuntos de datos de gran tamaño, en forma de objetos. Cada objeto se almacena en un contenedor denominado bucket, al que se puede acceder y administrar mediante las herramientas y AWS SDKs, AWS Management Console o mediante programación. AWS CLI

Amazon S3, que incluye almacenamiento básico, también ofrece una gama de características que incluyen la administración del ciclo de vida, el control de versiones, la escalabilidad y la seguridad. Se integran con otras, Servicios de AWS lo que le permite crear soluciones basadas en la nube que se adapten a sus necesidades.

AWS CLI Proporciona dos niveles de comandos para acceder a Amazon S3:

- `s3`: comandos personalizados de alto nivel creados específicamente para simplificar la AWS CLI realización de tareas comunes, como crear, manipular, eliminar y sincronizar objetos y depósitos.
- `s3api`: ofrece acceso directo a todas las operaciones de Amazon S3, lo que le permite llevar a cabo API operaciones avanzadas.

Temas de esta guía:

- [Utilice comandos de alto nivel \(s3\) con AWS CLI](#)
- [Utilice los API comandos -Level \(s3api\) con AWS CLI](#)
- [Ejemplo de scripting de operaciones de ciclo de vida de bucket de Amazon S3](#)

## Utilice comandos de alto nivel (s3) con AWS CLI

En este tema se describe alguno de los comandos que puede utilizar para administrar los buckets y los objetos de Simple Storage Service (Amazon S3) a través de comandos [aws s3](#) en AWS CLI. Para ver los comandos que no se tratan en este tema y ejemplos de comandos adicionales, consulte los comandos de [aws s3](#) en la Referencia de AWS CLI .

Los comandos de `aws s3` de alto nivel simplifican la administración de objetos de Amazon S3. Estos comandos le permiten administrar el contenido de Amazon S3 dentro de sí mismo y con directorios locales.

Temas

- [Requisitos previos](#)
- [Antes de comenzar](#)
- [Crear un bucket](#)
- [Lista de buckets y objetos](#)
- [Eliminar buckets](#)
- [Eliminar objetos](#)
- [Mover objetos](#)
- [Copia de objetos](#)
- [Sincronización de objetos](#)
- [Opciones utilizadas con frecuencia para los comandos s3](#)
- [Recursos](#)

## Requisitos previos

Para ejecutar los comandos de s3, necesitará:

- Instalar y configurar la AWS CLI. Para obtener más información, consulte [Instalar la AWS CLI y Credenciales de autenticación y acceso](#).
- El perfil que utilice debe tener permisos que permitan realizar las AWS operaciones de los ejemplos.
- Comprenda estos términos de Amazon S3:
  - Bucket: una carpeta de Amazon S3 de nivel superior.
  - Prefijo: una carpeta de Amazon S3 en un bucket.
  - Objeto: cualquier artículo alojado en un bucket de Amazon S3.

## Antes de comenzar

En esta sección se describen algunas cosas que hay que tener en cuenta antes de utilizar los comandos de `aws s3`.

### Cargas de objetos grandes

Cuando utiliza comandos `aws s3` para cargar objetos grandes en un bucket de Amazon S3, la AWS CLI automáticamente realiza una carga multiparte. Los errores de carga no pueden reanudarse cuando se usan estos comandos de `aws s3`.

Si la carga multiparte falla debido a un tiempo de espera, o si la cancelaste manualmente AWS CLI, AWS CLI se detiene la carga y se limpian los archivos que se hayan creado. Este proceso puede tardar varios minutos.

Si el proceso de carga o limpieza multiparte se cancela por un comando de cierre o un error del sistema, los archivos creados permanecen en el bucket de Amazon S3. <https://docs.aws.amazon.com/cli/latest/reference/s3api/abort-multipart-upload.html>

## Propiedades de archivo y etiquetas en copias multiparte

Cuando utiliza la AWS CLI versión 1 de los comandos del espacio de `aws s3` nombres para copiar un archivo de una ubicación de bucket de Amazon S3 a otra ubicación de bucket de Amazon S3 y esa operación utiliza una copia [multiparte, no se copia](#) ninguna propiedad de archivo del objeto de origen en el objeto de destino.

## Crear un bucket

Utilice el comando `s3 mb` para crear un bucket. Los nombres de los buckets deben ser únicos a nivel mundial (únicos en todo Amazon S3) y deben DNS cumplir con los requisitos.

Los nombres de los buckets pueden contener minúsculas, números, guiones y puntos. Los nombres de los buckets solo pueden empezar y terminar con una letra o número, y no pueden contener un punto junto a un guion u otro punto.

## Sintaxis

```
$ aws s3 mb <target> [--options]
```

## Ejemplos de s3 mb

En el siguiente ejemplo se crea el bucket `s3://bucket-name`.

```
$ aws s3 mb s3://bucket-name
```

## Lista de buckets y objetos

Para mostrar los buckets, carpetas u objetos, utilice el comando `s3 ls`. El uso del comando sin destino u opciones muestra todos los buckets.

## Sintaxis

```
$ aws s3 ls <target> [--options]
```

Para ver algunas opciones comunes que se pueden utilizar con este comando y ejemplos, consulte [Opciones utilizadas con frecuencia para los comandos s3](#). Para ver una lista completa de las opciones disponibles, consulte [s3 ls](#) en la Referencia de los comandos de AWS CLI .

## Ejemplos de s3 ls

En el siguiente ejemplo se enumeran todos los buckets de Amazon S3.

```
$ aws s3 ls
2018-12-11 17:08:50 my-bucket
2018-12-14 14:55:44 my-bucket2
```

El siguiente comando muestra todos los objetos y prefijos de un bucket. En este resultado de ejemplo, el prefijo `example/` tiene un archivo llamado `MyFile1.txt`.

```
$ aws s3 ls s3://bucket-name
                PRE example/
2018-12-04 19:05:48          3 MyFile1.txt
```

Para filtrar la salida por un prefijo específico, inclúyalo en el comando. El siguiente comando muestra los objetos de `bucket-name/example/` (es decir, objetos en `bucket-name` filtrados por el prefijo `example/`).

```
$ aws s3 ls s3://bucket-name/example/
2018-12-06 18:59:32          3 MyFile1.txt
```

## Eliminar buckets

Para eliminar un bucket, utilice el comando [s3 rb](#).

## Sintaxis

```
$ aws s3 rb <target> [--options]
```

## Ejemplos de s3 rb

En el siguiente ejemplo se quita el bucket `s3://bucket-name`.

```
$ aws s3 rb s3://bucket-name
```

De forma predeterminada, el bucket debe estar vacío para que la operación se realice correctamente. Para eliminar un bucket que no esté vacío, debe incluir la opción `--force`. Si utiliza un bucket versionado que contenga objetos eliminados previamente, pero que se conservan, este comando no le permitirá eliminar el bucket. En primer lugar, debe eliminar todo el contenido.

El siguiente ejemplo elimina todos los objetos y los prefijos en el bucket, luego elimina el bucket.

```
$ aws s3 rb s3://bucket-name --force
```

## Eliminar objetos

Para eliminar objetos en un bucket o en el directorio local, utilice el comando [s3 rm](#).

## Sintaxis

```
$ aws s3 rm <target> [--options]
```

Para ver algunas opciones comunes que se pueden utilizar con este comando y ejemplos, consulte [Opciones utilizadas con frecuencia para los comandos s3](#). Para ver una lista completa de opciones, consulte [s3 rm](#) en la Referencia de los comandos de AWS CLI .

## Ejemplos de s3 rm

En el siguiente ejemplo se elimina `filename.txt` de `s3://bucket-name/example`.

```
$ aws s3 rm s3://bucket-name/example/filename.txt
```

En el siguiente ejemplo se eliminan todos los objetos de `s3://bucket-name/example` utilizando la opción `--recursive`.

```
$ aws s3 rm s3://bucket-name/example --recursive
```

## Mover objetos

Use el comando [s3 mv](#) para mover objetos de un bucket o un directorio local. El `s3 mv` comando copia el objeto o archivo de origen en el destino especificado y, a continuación, elimina el objeto o archivo de origen.

## Sintaxis

```
$ aws s3 mv <source> <target> [--options]
```

Para ver algunas opciones comunes que se pueden utilizar con este comando y ejemplos, consulte [Opciones utilizadas con frecuencia para los comandos s3](#). Para ver una lista completa de las opciones disponibles, consulte [s3 mv](#) en la Referencia de los comandos de AWS CLI .

### Warning

Si utiliza algún tipo de punto de acceso ARNs o alias de punto de acceso en su fuente o destino de Amazon S3URIs, debe tener especial cuidado de que su Amazon S3 de origen y destino se URIs resuelvan en diferentes buckets subyacentes. Si los depósitos de origen y destino son los mismos, el archivo u objeto de origen se puede mover a sí mismo, lo que puede provocar la eliminación accidental del archivo u objeto de origen. Para comprobar que los depósitos de origen y destino no son los mismos, utilice el `--validate-same-s3-paths` parámetro o defina la variable [AWS\\_CLI\\_S3\\_MV\\_VALIDATE\\_SAME\\_S3\\_PATHS](#) de entorno en `true`

## Ejemplos de s3 mv

En el siguiente ejemplo se mueven todos los objetos de `s3://bucket-name/example` a `s3://my-bucket/`.

```
$ aws s3 mv s3://bucket-name/example s3://my-bucket/
```

En el siguiente ejemplo se mueve un archivo local del directorio de trabajo actual al bucket de Amazon S3 con el comando de `s3 mv`.

```
$ aws s3 mv filename.txt s3://bucket-name
```

En el siguiente ejemplo se mueve un archivo del bucket de Amazon S3 a su directorio de trabajo actual, donde `./` especifica su directorio de trabajo actual.

```
$ aws s3 mv s3://bucket-name/filename.txt ./
```

## Copia de objetos

Use el comando [s3 cp](#) para copiar objetos de un bucket o un directorio local.

### Sintaxis

```
$ aws s3 cp <source> <target> [--options]
```

Puede usar el parámetro guión para el streaming de archivos a la entrada estándar (stdin) o salida estándar (stdout).

#### Warning

Si lo estás utilizando PowerShell, el shell podría alterar la codificación de una entrada CRLF o salida canalizada, añadir una CRLF a una entrada o salida canalizada o redirigir la salida.

El comando `s3 cp` utiliza la siguiente sintaxis para cargar una secuencia de archivos desde `stdin` hasta un bucket especificado.

### Sintaxis

```
$ aws s3 cp - <target> [--options]
```

El comando `s3 cp` utiliza la siguiente sintaxis para descargar una secuencia de archivos de Amazon S3 para `stdout`.

### Sintaxis

```
$ aws s3 cp <target> [--options] -
```

Para ver algunas opciones comunes que se pueden utilizar con este comando y ejemplos, consulte [Opciones utilizadas con frecuencia para los comandos s3](#). Para ver una lista completa de opciones, consulte [s3 cp](#) en la Referencia de los comandos de AWS CLI .

## Ejemplos de `s3 cp`

El siguiente ejemplo copia todos los registros de `s3://bucket-name/example` a `s3://my-bucket/`.



```
$ aws s3 cp s3://bucket-name/example s3://my-bucket/
```

En el siguiente ejemplo se copian un archivo local del directorio de trabajo actual en el bucket de Amazon S3 con el `s3 cp` comando.

```
$ aws s3 cp filename.txt s3://bucket-name
```

En el siguiente ejemplo se copia un archivo de su bucket de Amazon S3 en su directorio de trabajo actual, donde `./` especifica su directorio de trabajo actual.

```
$ aws s3 cp s3://bucket-name/filename.txt ./
```

En el siguiente ejemplo se utiliza la repetición para transmitir el texto “hola mundo” al archivo `s3://bucket-name/filename.txt`.

```
$ echo "hello world" | aws s3 cp - s3://bucket-name/filename.txt
```

En el ejemplo siguiente se transmite el archivo `s3://bucket-name/filename.txt` a `stdout` y se imprime el contenido en la consola.

```
$ aws s3 cp s3://bucket-name/filename.txt -  
hello world
```

En el siguiente ejemplo se transmiten los contenidos de `s3://bucket-name/pre` a `stdout`, se utiliza el comando `bzip2` para comprimir los archivos y se carga el nuevo archivo comprimido llamado `key.bz2` a `s3://bucket-name`.

```
$ aws s3 cp s3://bucket-name/pre - | bzip2 --best | aws s3 cp - s3://bucket-name/  
key.bz2
```

## Sincronización de objetos

El comando [s3 sync](#) sincroniza el contenido de un bucket y un directorio o los contenidos de dos buckets. Normalmente, `s3 sync` copia archivos u objetos que estén desactualizados o que falten entre el origen y el destino. Sin embargo, también puede introducir la opción `--delete` para quitar archivos u objetos desde el destino que no se encuentran en el origen.

## Sintaxis

```
$ aws s3 sync <source> <target> [--options]
```

Para ver algunas opciones comunes que se pueden utilizar con este comando y ejemplos, consulte [Opciones utilizadas con frecuencia para los comandos s3](#). Para ver una lista completa de opciones, consulte [s3 sync](#) en la Referencia de los comandos de AWS CLI .

### Ejemplos de sincronización de s3

En el siguiente ejemplo se sincroniza el contenido de un prefijo Amazon S3 llamado ruta en el bucket denominado my-bucket con el directorio activo actual.

s3 sync actualiza los archivos que tengan un tamaño o tiempo de modificación diferente que los archivos con el mismo nombre en el destino. La salida muestra las operaciones específicas realizadas durante la sincronización. Observe que la operación sincroniza de forma recursiva el subdirectorio MySubdirectory y su contenido con s3://my-bucket/path/MySubdirectory.

```
$ aws s3 sync . s3://my-bucket/path
upload: MySubdirectory\MyFile3.txt to s3://my-bucket/path/MySubdirectory/MyFile3.txt
upload: MyFile2.txt to s3://my-bucket/path/MyFile2.txt
upload: MyFile1.txt to s3://my-bucket/path/MyFile1.txt
```

En el siguiente ejemplo, que amplía el anterior, se muestra cómo funciona la opción --delete.

```
// Delete local file
$ rm ./MyFile1.txt

// Attempt sync without --delete option - nothing happens
$ aws s3 sync . s3://my-bucket/path

// Sync with deletion - object is deleted from bucket
$ aws s3 sync . s3://my-bucket/path --delete
delete: s3://my-bucket/path/MyFile1.txt

// Delete object from bucket
$ aws s3 rm s3://my-bucket/path/MySubdirectory/MyFile3.txt
delete: s3://my-bucket/path/MySubdirectory/MyFile3.txt

// Sync with deletion - local file is deleted
$ aws s3 sync s3://my-bucket/path . --delete
```

```
delete: MySubdirectory\MyFile3.txt

// Sync with Infrequent Access storage class
$ aws s3 sync . s3://my-bucket/path --storage-class STANDARD_IA
```

Cuando se utiliza la opción `--delete`, las opciones `--exclude` y `--include` pueden filtrar archivos u objetos para eliminarlos durante una operación de `s3 sync`. En este caso, la cadena del parámetro debe especificar qué archivos se deben excluir o incluir en la eliminación, en el contexto del directorio o bucket de destino. A continuación se muestra un ejemplo.

```
Assume local directory and s3://my-bucket/path currently in sync and each contains 3
files:
MyFile1.txt
MyFile2.rtf
MyFile88.txt
...

// Sync with delete, excluding files that match a pattern. MyFile88.txt is deleted,
while remote MyFile1.txt is not.
$ aws s3 sync . s3://my-bucket/path --delete --exclude "path/MyFile?.txt"
delete: s3://my-bucket/path/MyFile88.txt
...

// Sync with delete, excluding MyFile2.rtf - local file is NOT deleted
$ aws s3 sync s3://my-bucket/path . --delete --exclude "./MyFile2.rtf"
download: s3://my-bucket/path/MyFile1.txt to MyFile1.txt
...

// Sync with delete, local copy of MyFile2.rtf is deleted
$ aws s3 sync s3://my-bucket/path . --delete
delete: MyFile2.rtf
```

## Opciones utilizadas con frecuencia para los comandos s3

Las siguientes opciones se utilizan con frecuencia para los comandos descritos en este tema. Para obtener una lista completa de las opciones que puede usar en un comando, consulte el comando específico en la guía de [AWS CLI referencia de la AWS CLI versión 2 de la guía](#) .

### acl

`s3 sync` y `s3 cp` pueden utilizar la opción `--acl`. Esto le permite configurar los permisos de acceso para archivos copiados en Amazon S3. La opción `--acl` admite los valores `private`,

`public-read` y `public-read-write`. Para obtener más información, consulte [Canned ACL](#) en la Guía del usuario de Amazon S3.

```
$ aws s3 sync . s3://my-bucket/path --acl public-read
```

## excluya

Cuando utiliza el comando `s3 cp`, `s3 mv`, `s3 sync` o `s3 rm`, puede filtrar los resultados mediante la opción `--exclude` o `--include`. La opción `--exclude` establece reglas para excluir únicamente objetos del comando y las opciones se aplican en el orden especificado. Esto se muestra en el siguiente ejemplo.

```
Local directory contains 3 files:
MyFile1.txt
MyFile2.rtf
MyFile88.txt

// Exclude all .txt files, resulting in only MyFile2.rtf being copied
$ aws s3 cp . s3://my-bucket/path --exclude "*.txt"

// Exclude all .txt files but include all files with the "MyFile*.txt" format,
  resulting in, MyFile1.txt, MyFile2.rtf, MyFile88.txt being copied
$ aws s3 cp . s3://my-bucket/path --exclude "*.txt" --include "MyFile*.txt"

// Exclude all .txt files, but include all files with the "MyFile*.txt" format,
  but exclude all files with the "MyFile?.txt" format resulting in, MyFile2.rtf and
  MyFile88.txt being copied
$ aws s3 cp . s3://my-bucket/path --exclude "*.txt" --include "MyFile*.txt" --
exclude "MyFile?.txt"
```

## incluir

Cuando utiliza el comando `s3 cp`, `s3 mv`, `s3 sync` o `s3 rm`, puede filtrar los resultados utilizando la opción `--exclude` o `--include`. La opción `--include` establece reglas para incluir únicamente objetos específicos para el comando y las opciones se aplican en el orden especificado. Esto se muestra en el siguiente ejemplo.

```
Local directory contains 3 files:
MyFile1.txt
MyFile2.rtf
MyFile88.txt
```

```
// Include all .txt files, resulting in MyFile1.txt and MyFile88.txt being copied
$ aws s3 cp . s3://my-bucket/path --include "*.txt"

// Include all .txt files but exclude all files with the "MyFile*.txt" format,
resulting in no files being copied
$ aws s3 cp . s3://my-bucket/path --include "*.txt" --exclude "MyFile*.txt"

// Include all .txt files, but exclude all files with the "MyFile*.txt" format, but
include all files with the "MyFile?.txt" format resulting in MyFile1.txt being
copied

$ aws s3 cp . s3://my-bucket/path --include "*.txt" --exclude "MyFile*.txt" --
include "MyFile?.txt"
```

## concesión

Los comandos `s3 cp`, `s3 mv` y `s3 sync` incluyen una opción `--grants` que puede usar para conceder permisos sobre el objeto a usuarios o grupos específicos. Configure la opción `--grants` para obtener una lista de permisos mediante la siguiente sintaxis. Reemplace `Permission`, `Grantee_Type` y `Grantee_ID` por sus propios valores.

## Sintaxis

```
--grants Permission=Grantee_Type=Grantee_ID
        [Permission=Grantee_Type=Grantee_ID ...]
```

Cada valor contiene los siguientes elementos:

- *Permission* — Especifica los permisos concedidos. Se puede establecer en `read`, `readacl`, `writeacl` o `full`.
- *Grantee\_Type* — Especifica cómo identificar al concesionario. Se puede establecer en `uri`, `emailaddress` o `id`.
- *Grantee\_ID* — Especifica el concesionario en función de *Grantee\_Type*.
  - `uri`— Del grupo. URI Para obtener más información, consulte la sección [¿Quién es un beneficiario?](#)
  - `emailaddress`: dirección de correo electrónico de la cuenta.
  - `id`: ID canónico de la cuenta.

Para obtener más información acerca de cómo controlar el acceso a Amazon S3, consulte la sección de [Control de acceso](#).

En el siguiente ejemplo se copia un objeto en un bucket. Concede permisos read sobre el objeto a todos los usuarios y permisos full (read, readacl y writeacl) a la cuenta asociada con user@example.com.

```
$ aws s3 cp file.txt s3://my-bucket/ --grants read=http://acs.amazonaws.com/groups/global/AllUsers full=emailaddress=user@example.com
```

También puede especificar una clase de almacenamiento no predeterminada (REDUCED\_REDUNDANCY o STANDARD\_IA) para los objetos que se cargan en Amazon S3. Para ello, utilice la opción --storage-class.

```
$ aws s3 cp file.txt s3://my-bucket/ --storage-class REDUCED_REDUNDANCY
```

## recursive

Cuando use esta opción, el comando se ejecuta en todos los objetos o archivos del directorio especificado o con el prefijo indicado. En el siguiente ejemplo se elimina s3://my-bucket/path y todos sus contenidos.

```
$ aws s3 rm s3://my-bucket/path --recursive
```

## Recursos

AWS CLI referencia:

- [aws s3](#)
- [aws s3 cp](#)
- [aws s3 mb](#)
- [aws s3 mv](#)
- [aws s3 ls](#)
- [aws s3 rb](#)
- [aws s3 rm](#)
- [aws s3 sync](#)

Referencia de servicio:

- [Uso de buckets de Amazon S3](#) en la Guía del usuario de Amazon S3
- [Uso de objetos de Amazon S3](#) en la Guía del usuario de Amazon S3
- [Listar las claves jerárquicamente mediante un prefijo y un delimitador](#) en la Guía del usuario de Amazon S3
- [Anule las cargas de varias partes a un bucket de S3 utilizando el AWS SDK for .NET \(nivel bajo\) de la](#) Guía del usuario de Amazon S3

## Utilice los API comandos -Level (s3api) con AWS CLI

Los comandos API de nivel (incluidos en el conjunto de `s3api` comandos) proporcionan acceso directo al Amazon Simple Storage Service (Amazon APIs S3) y permiten algunas operaciones que no están expuestas en los comandos de alto nivel. `s3` Estos comandos son equivalentes a los demás AWS servicios que proporcionan acceso de API nivel 1 a la funcionalidad de los servicios. Para obtener más información acerca de los comandos `s3`, consulte [Utilice comandos de alto nivel \(s3\) con AWS CLI](#)

En este tema se proporcionan ejemplos que demuestran cómo utilizar los comandos de nivel inferior que se asignan a Amazon S3 APIs. Además, puede encontrar ejemplos de cada API comando de S3 en la `s3api` sección de la guía de [AWS CLI referencia de la AWS CLI versión 2 de la guía](#) .

### Temas

- [Requisitos previos](#)
- [Aplica un personalizado ACL](#)
- [Configuración de una política de registro](#)
- [Recursos](#)

### Requisitos previos

Para ejecutar los comandos de `s3api`, necesitará:

- Instalar y configurar la AWS CLI. Para obtener más información, consulte [Instalar la AWS CLI y Credenciales de autenticación y acceso](#).
- El perfil que utilice debe tener permisos que permitan realizar las AWS operaciones de los ejemplos.
- Comprenda estos términos de Amazon S3:
  - Bucket: una carpeta de Amazon S3 de nivel superior.

- Prefijo: una carpeta de Amazon S3 en un bucket.
- Objeto: cualquier artículo alojado en un bucket de Amazon S3.

## Aplica un personalizado ACL

Con los comandos de alto nivel, puede utilizar la `--acl` opción para aplicar listas de control de acceso predefinidas (ACLs) a los objetos de Amazon S3. Sin embargo, no puede usar ese comando para configurar todo el depósito. ACLs Sin embargo, puede hacerlo mediante el comando [put-bucket-acl](#) API -level.

El siguiente ejemplo muestra cómo conceder el control total a dos AWS usuarios (`user1@example.com` y `user2@example.com`) y el permiso de lectura a todos. El identificador de «todos» proviene de un especial URI que se pasa como parámetro.

```
$ aws s3api put-bucket-acl --bucket MyBucket --grant-full-control  
'emailaddress="user1@example.com",emailaddress="user2@example.com"' --grant-read  
'uri="http://acs.amazonaws.com/groups/global/AllUsers"'
```

Para obtener más información sobre cómo construir el ACLs, consulte [PUTBucket acl](#) en la API referencia de Amazon Simple Storage Service. Los `s3api` ACL comandos de CLI, por ejemplo `put-bucket-acl`, utilizan la misma notación [abreviada de argumentos](#).

## Configuración de una política de registro

El API comando `put-bucket-logging` configura una política de registro de cubos.

En el siguiente ejemplo, el AWS usuario `user@example.com` tiene el control total sobre los archivos de registro y todos los usuarios tienen acceso de lectura a ellos. Tenga en cuenta que el `put-bucket-acl` comando también es necesario para conceder al sistema de entrega de registros de Amazon S3 (especificado mediante aURI) los permisos necesarios para leer y escribir los registros en el bucket.

```
$ aws s3api put-bucket-acl --bucket MyBucket --grant-read-acp 'URI="http://  
acs.amazonaws.com/groups/s3/LogDelivery"' --grant-write 'URI="http://acs.amazonaws.com/  
groups/s3/LogDelivery"'
```

```
$ aws s3api put-bucket-logging --bucket MyBucket --bucket-logging-status file://  
logging.json
```

El archivo `logging.json` del comando anterior contiene lo siguiente.



```
{
  "LoggingEnabled": {
    "TargetBucket": "MyBucket",
    "TargetPrefix": "MyBucketLogs/",
    "TargetGrants": [
      {
        "Grantee": {
          "Type": "AmazonCustomerByEmail",
          "EmailAddress": "user@example.com"
        },
        "Permission": "FULL_CONTROL"
      },
      {
        "Grantee": {
          "Type": "Group",
          "URI": "http://acs.amazonaws.com/groups/global/AllUsers"
        },
        "Permission": "READ"
      }
    ]
  }
}
```

## Recursos

### AWS CLI referencia:

- [aws s3api](#)
- [aws s3api put-bucket-acl](#)
- [aws s3api put-bucket-logging](#)

### Referencia de servicio:

- [Uso de buckets de Amazon S3](#) en la Guía del usuario de Amazon S3
- [Uso de objetos de Amazon S3](#) en la Guía del usuario de Amazon S3
- [Listar las claves jerárquicamente mediante un prefijo y un delimitador](#) en la Guía del usuario de Amazon S3
- [Anule las cargas de varias partes a un bucket de S3 utilizando el AWS SDK for .NET \(nivel bajo de la](#) Guía del usuario de Amazon S3

## Ejemplo de scripting de operaciones de ciclo de vida de bucket de Amazon S3

En este tema se utiliza un ejemplo de scripting de Bash para las operaciones del ciclo de vida del bucket de Amazon S3 mediante la AWS Command Line Interface (AWS CLI). En este ejemplo de scripting se utiliza el conjunto de comandos de [aws s3api](#). Los scripts de shell son programas diseñados para ejecutarse en una interfaz de línea de comandos.

### Temas

- [Antes de comenzar](#)
- [Acerca de este ejemplo](#)
- [Archivos](#)
- [Referencias](#)

### Antes de comenzar

Antes de que pueda ejecutar cualquiera de los siguientes ejemplos, se debe completar lo siguiente.

- Instalar y configurar la AWS CLI. Para obtener más información, consulte [Instalar la AWS CLI y Credenciales de autenticación y acceso](#).
- El perfil que utilice debe tener permisos que permitan realizar las AWS operaciones que se realizan en los ejemplos.
- Como práctica AWS recomendada, conceda a este código los privilegios mínimos o solo los permisos necesarios para realizar una tarea. Para obtener más información, consulte [Concesión de privilegios mínimos](#) en la Guía del IAM usuario.
- Este código no se ha probado en todas AWS las regiones. Algunos AWS servicios solo están disponibles en regiones específicas. Para obtener más información, consulte [Puntos de enlace de servicio y cuotas](#) en la Guía de referencia general de AWS .
- La ejecución de este código puede generar cargos en su AWS cuenta. Es su responsabilidad asegurarse de que los recursos creados por este script se eliminen cuando haya terminado con ellos.

El servicio Amazon S3 utiliza los siguientes términos:

- Bucket: una carpeta de Amazon S3 de nivel superior.
- Prefijo: una carpeta de Amazon S3 en un bucket.

- Objeto: cualquier artículo alojado en un bucket de Amazon S3.

## Acerca de este ejemplo

En este ejemplo se muestra cómo interactuar con algunas de las operaciones básicas de Amazon S3 mediante un conjunto de funciones en archivos de script de shell. Las funciones se encuentran en el archivo de script de shell llamado `bucket-operations.sh`. Puede llamar a estas funciones en otro archivo. Cada archivo de script contiene comentarios que describen cada una de las funciones.

Para ver los resultados intermedios de cada paso, ejecute el script con un parámetro `-i`. Para ver el estado actual del bucket o su contenido, puede usar la consola de Amazon S3. El scripting solo pasa al siguiente paso cuando se pulsa Enter (Intro) en el símbolo del sistema.

Para ver el ejemplo completo y los archivos de script descargables, consulte [Amazon S3 Bucket Lifecycle Operations](#) en el repositorio de ejemplos de AWS código en GitHub.

## Archivos

El ejemplo contiene los siguientes archivos:

### `bucket-operations.sh`

Este archivo de script principal puede obtenerse de otro archivo. Incluye funciones que realizan las siguientes tareas:

- Creación de un bucket y verificación de que existe
- Copia de un archivo desde el equipo local a un bucket
- Copia de un archivo desde una ubicación de bucket a otra ubicación de bucket
- Listado de contenidos de un bucket
- Eliminación a archivos desde un bucket
- Eliminación de un bucket

Vea el código de [bucket-operations.sh](#) on GitHub.

### `test-bucket-operations.sh`

El archivo de script de shell `test-bucket-operations.sh` muestra cómo llamar a las funciones mediante el suministro de archivo `bucket-operations.sh` y el llamando a cada una de las funciones. Después de llamar a las funciones, el script de prueba elimina todos los recursos que creó.

Vea el código de [test-bucket-operations.sh](#) on. GitHub

## awsdocs-general.sh

El archivo de script `awsdocs-general.sh` contiene funciones de uso general utilizadas en ejemplos de código avanzados para la AWS CLI.

Ver el código de [awsdocs-general.sh](#) on GitHub.

## Referencias

AWS CLI referencia:

- [aws s3api](#)
- [aws s3api create-bucket](#)
- [aws s3api copy-object](#)
- [aws s3api delete-bucket](#)
- [aws s3api delete-object](#)
- [aws s3api head-bucket](#)
- [aws s3api list-objects](#)
- [aws s3api put-object](#)

Otra referencia:

- [Uso de buckets de Amazon S3](#) en la Guía del usuario de Amazon S3
- [Uso de objetos de Amazon S3](#) en la Guía del usuario de Amazon S3
- Para ver ejemplos de AWS CLI código AWS SDK y contribuir a ellos, consulte el [repositorio AWS de ejemplos de código](#) en GitHub.

## Usar Amazon SNS con la AWS CLI

Puede acceder a las características de Amazon Simple Notification Service (Amazon SNS) a través de AWS Command Line Interface (AWS CLI). Para mostrar la lista de los comandos de AWS CLI para Amazon SNS, utilice el siguiente comando.

## aws sns help

Antes de ejecutar los comandos, defina sus credenciales predeterminadas. Para obtener más información, consulte [Configure el AWS CLI](#).

En este tema se muestran ejemplos de comandos de la AWS CLI que realizan tareas comunes para Amazon SNS.

### Temas

- [Crear un tema](#)
- [Suscripción a un tema](#)
- [Publicar en un tema](#)
- [Cancelación de la suscripción a un tema](#)
- [Eliminación de un tema](#)

### Crear un tema

Para crear un tema, utilice el comando [sns create-topic](#) y especifique el nombre que desea asignar al tema.

```
$ aws sns create-topic --name my-topic
{
  "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic"
}
```

Anote el valor de `TopicArn` de la respuesta, que utilizará más adelante para publicar un mensaje.

### Suscripción a un tema

Para suscribirse a un tema, utilice el comando [sns subscribe](#).

En el siguiente ejemplo se especifica el protocolo `email` y una dirección de correo electrónico para el `notification-endpoint`.

```
$ aws sns subscribe --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic --
protocol email --notification-endpoint saanvi@example.com
```

```
{
  "SubscriptionArn": "pending confirmation"
}
```

AWS envía inmediatamente un mensaje de confirmación a la dirección de correo electrónico que ha especificado en el comando `subscribe`. El mensaje de correo electrónico tiene este texto.

```
You have chosen to subscribe to the topic:
arn:aws:sns:us-west-2:123456789012:my-topic
To confirm this subscription, click or visit the following link (If this was in error
no action is necessary):
Confirm subscription
```

Cuando el destinatario hace clic en el enlace `Confirm subscription` (Confirmar suscripción), el navegador del destinatario muestra un mensaje de notificación con información similar a la siguiente.

```
Subscription confirmed!

You have subscribed saanvi@example.com to the topic:my-topic.

Your subscription's id is:
arn:aws:sns:us-west-2:123456789012:my-topic:1328f057-de93-4c15-512e-8bb22EXAMPLE

If it was not your intention to subscribe, click here to unsubscribe.
```

## Publicar en un tema

Para enviar un mensaje a todos los suscriptores de un tema, utilice el comando [sns publish](#).

En el siguiente ejemplo se envía el mensaje “Hola mundo” a todos los suscriptores del tema especificado.

```
$ aws sns publish --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic --
message "Hello World!"
{
  "MessageId": "4e41661d-5eec-5ddf-8dab-2c867EXAMPLE"
}
```

En este ejemplo, AWS envía un mensaje de correo electrónico con el texto “Hola mundo” a `saanvi@example.com`.

## Cancelación de la suscripción a un tema

Para cancelar la suscripción a un tema y dejar de recibir los mensajes que se publican en él, utilice el comando [sns unsubscribe](#) y especifique el ARN del tema del que desea cancelar la suscripción.

```
$ aws sns unsubscribe --subscription-arn arn:aws:sns:us-west-2:123456789012:my-  
topic:1328f057-de93-4c15-512e-8bb22EXAMPLE
```

Para verificar que se ha cancelado la suscripción correctamente, utilice el comando [sns list-subscriptions](#) para confirmar que el ARN ya no aparece en la lista.

```
$ aws sns list-subscriptions
```

## Eliminación de un tema

Para eliminar un tema, ejecute el comando [sns delete-topic](#).

```
$ aws sns delete-topic --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic
```

Para verificar que AWS ha eliminado el tema correctamente, utilice el comando [sns list-topics](#) para confirmar que el tema ya no aparece en la lista.

```
$ aws sns list-topics
```

## AWS CLI ejemplos de comandos

Los ejemplos de código de este tema muestran cómo usar el AWS Command Line Interface with AWS.

Los conceptos básicos son ejemplos de código que muestran cómo realizar las operaciones esenciales dentro de un servicio.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Los escenarios son ejemplos de código que muestran cómo llevar a cabo una tarea específica a través de llamadas a varias funciones dentro del servicio o combinado con otros Servicios de AWS.

## Servicios

- [ACMEjemplos que utilizan AWS CLI](#)
- [APIEjemplos de puertas de enlace que utilizan AWS CLI](#)
- [APIGateway HTTP y WebSocket API ejemplos de uso AWS CLI](#)
- [APIAPIEjemplos de administración de puertas de enlace utilizando AWS CLI](#)
- [Ejemplos de App Mesh usando AWS CLI](#)
- [Ejemplos de App Runner que utilizan AWS CLI](#)
- [AWS AppConfig ejemplos que utilizan AWS CLI](#)
- [Ejemplos de Application Auto Scaling utilizando AWS CLI](#)
- [Ejemplos de Application Discovery Service que utilizan AWS CLI](#)
- [AppRegistry ejemplos que utilizan AWS CLI](#)
- [Ejemplos de Athena usando AWS CLI](#)
- [Ejemplos de Auto Scaling usando AWS CLI](#)
- [Ejemplos de planes de Auto Scaling utilizando AWS CLI](#)
- [AWS Backup ejemplos que utilizan AWS CLI](#)
- [AWS Batch ejemplos que utilizan AWS CLI](#)
- [AWS Budgets ejemplos que utilizan AWS CLI](#)
- [Ejemplos de Amazon Chime con AWS CLI](#)
- [APIEjemplos de Cloud Control con AWS CLI](#)
- [AWS Cloud Map ejemplos que utilizan AWS CLI](#)
- [AWS Cloud9 ejemplos que utilizan AWS CLI](#)
- [AWS CloudFormation ejemplos que utilizan AWS CLI](#)
- [CloudFront ejemplos que utilizan AWS CLI](#)
- [CloudSearch Ejemplos de Amazon que utilizan AWS CLI](#)
- [CloudTrail ejemplos que utilizan AWS CLI](#)
- [CloudWatch ejemplos que utilizan AWS CLI](#)
- [CloudWatch Registra ejemplos usando AWS CLI](#)
- [CloudWatch Ejemplos de monitoreo de red usando AWS CLI](#)



- [CodeArtifact ejemplos que utilizan AWS CLI](#)
- [CodeBuild ejemplos que utilizan AWS CLI](#)
- [CodeCommit ejemplos que utilizan AWS CLI](#)
- [CodeDeploy ejemplos que utilizan AWS CLI](#)
- [CodeGuru Ejemplos de revisores que utilizan AWS CLI](#)
- [CodePipeline ejemplos que utilizan AWS CLI](#)
- [AWS CodeStar ejemplos que utilizan AWS CLI](#)
- [AWS CodeStar Ejemplos de notificaciones que utilizan AWS CLI](#)
- [CodeConnections ejemplos que utilizan AWS CLI](#)
- [Ejemplos de Amazon Cognito Identity utilizando AWS CLI](#)
- [Ejemplos de Amazon Cognito Identity Provider que utilizan AWS CLI](#)
- [Amazon Comprehend ejemplos utilizando AWS CLI](#)
- [Ejemplos de Amazon Comprehend Medical utilizando AWS CLI](#)
- [AWS Config ejemplos que utilizan AWS CLI](#)
- [Ejemplos de Amazon Connect que utilizan AWS CLI](#)
- [AWS Cost and Usage Report ejemplos que utilizan AWS CLI](#)
- [Ejemplos del servicio Cost Explorer utilizando AWS CLI](#)
- [Ejemplos de Firehose usando AWS CLI](#)
- [Ejemplos de Amazon Data Lifecycle Manager que utilizan AWS CLI](#)
- [AWS Data Pipeline ejemplos que utilizan AWS CLI](#)
- [DataSync ejemplos que utilizan AWS CLI](#)
- [DAXejemplos que utilizan AWS CLI](#)
- [Ejemplos de Detectives que utilizan AWS CLI](#)
- [Ejemplos de Device Farm que utilizan AWS CLI](#)
- [AWS Direct Connect ejemplos que utilizan AWS CLI](#)
- [AWS Directory Service ejemplos que utilizan AWS CLI](#)
- [AWS DMS ejemplos que utilizan AWS CLI](#)
- [Ejemplos de Amazon DocumentDB que utilizan AWS CLI](#)
- [Ejemplos de DynamoDB que utilizan AWS CLI](#)

- [Ejemplos de DynamoDB Streams que utilizan AWS CLI](#)
- [EC2Ejemplos de Amazon que utilizan AWS CLI](#)
- [Ejemplos de Amazon EC2 Instance Connect que utilizan AWS CLI](#)
- [ECREjemplos de Amazon que utilizan AWS CLI](#)
- [ECSEjemplos de Amazon que utilizan AWS CLI](#)
- [EFSEjemplos de Amazon que utilizan AWS CLI](#)
- [EKSEjemplos de Amazon que utilizan AWS CLI](#)
- [Ejemplos de Elastic Beanstalk usando AWS CLI](#)
- [Ejemplos de Elastic Load Balancing: versión 1 utilizando AWS CLI](#)
- [Ejemplos de Elastic Load Balancing: versión 2 utilizando AWS CLI](#)
- [Ejemplos de Elastic Transcoder usando AWS CLI](#)
- [ElastiCache ejemplos que utilizan AWS CLI](#)
- [MediaStore ejemplos que utilizan AWS CLI](#)
- [EMREjemplos de Amazon que utilizan AWS CLI](#)
- [Amazon EMR sobre EKS ejemplos que utilizan AWS CLI](#)
- [EventBridge ejemplos que utilizan AWS CLI](#)
- [Ejemplos de Firewall Manager que utilizan AWS CLI](#)
- [AWS FIS ejemplos que utilizan AWS CLI](#)
- [GameLift Ejemplos de Amazon que utilizan AWS CLI](#)
- [Ejemplos de Global Accelerator utilizando AWS CLI](#)
- [AWS Glue ejemplos que utilizan AWS CLI](#)
- [GuardDuty ejemplos que utilizan AWS CLI](#)
- [AWS Health ejemplos que utilizan AWS CLI](#)
- [HealthImaging ejemplos que utilizan AWS CLI](#)
- [HealthLake ejemplos que utilizan AWS CLI](#)
- [HealthOmics ejemplos que utilizan AWS CLI](#)
- [IAMEjemplos que utilizan AWS CLI](#)
- [IAMEjemplos de Access Analyzer utilizando AWS CLI](#)

- [Ejemplos de Image Builder utilizando AWS CLI](#)
- [Ejemplos de Incident Manager que utilizan AWS CLI](#)
- [Ejemplos de contactos de Incident Manager que utilizan AWS CLI](#)
- [Ejemplos de Amazon Inspector que utilizan AWS CLI](#)
- [AWS IoT ejemplos que utilizan AWS CLI](#)
- [AWS IoT 1-Click Ejemplos de dispositivos que utilizan AWS CLI](#)
- [AWS IoT 1-Click Ejemplos de proyectos que utilizan AWS CLI](#)
- [AWS IoT Analytics ejemplos que utilizan AWS CLI](#)
- [Ejemplos de Device Advisor que utilizan AWS CLI](#)
- [AWS IoT data ejemplos que utilizan AWS CLI](#)
- [AWS IoT Events ejemplos que utilizan AWS CLI](#)
- [AWS IoT Events-Data ejemplos que utilizan AWS CLI](#)
- [AWS IoT Greengrass ejemplos que utilizan AWS CLI](#)
- [AWS IoT Greengrass V2 ejemplos que utilizan AWS CLI](#)
- [AWS IoT Jobs SDK release ejemplos que utilizan AWS CLI](#)
- [AWS IoT SiteWise ejemplos que utilizan AWS CLI](#)
- [AWS IoT Things Graph ejemplos que utilizan AWS CLI](#)
- [AWS IoT Wireless ejemplos que utilizan AWS CLI](#)
- [IVSEjemplos de Amazon que utilizan AWS CLI](#)
- [Ejemplos de Amazon IVS Chat que utilizan AWS CLI](#)
- [Ejemplos de Amazon IVS Real-Time Streaming utilizando AWS CLI](#)
- [Ejemplos de Amazon Kendra que utilizan AWS CLI](#)
- [Ejemplos de Kinesis que utilizan AWS CLI](#)
- [AWS KMS ejemplos que utilizan AWS CLI](#)
- [Ejemplos de Lake Formation usando AWS CLI](#)
- [Ejemplos de Lambda que utilizan AWS CLI](#)
- [Ejemplos de License Manager que utilizan AWS CLI](#)
- [Ejemplos de Lightsail usando AWS CLI](#)
- [Ejemplos de Macie usando AWS CLI](#)

- [Ejemplos de Grafana gestionados por Amazon utilizando AWS CLI](#)
- [MediaConnect ejemplos que utilizan AWS CLI](#)
- [MediaConvert ejemplos que utilizan AWS CLI](#)
- [MediaLive ejemplos que utilizan AWS CLI](#)
- [MediaPackage ejemplos que utilizan AWS CLI](#)
- [MediaPackage VODejemplos que utilizan AWS CLI](#)
- [MediaStore Ejemplos de planos de datos que utilizan AWS CLI](#)
- [MediaTailor ejemplos que utilizan AWS CLI](#)
- [Ejemplos de MemoryDB usando AWS CLI](#)
- [MSKEjemplos de Amazon que utilizan AWS CLI](#)
- [Ejemplos de Network Manager que utilizan AWS CLI](#)
- [Ejemplos de Nimble Studio usando AWS CLI](#)
- [OpenSearch Ejemplos de servicios que utilizan AWS CLI](#)
- [AWS OpsWorks ejemplos que utilizan AWS CLI](#)
- [AWS OpsWorks CM ejemplos que utilizan AWS CLI](#)
- [Ejemplos de organizaciones que utilizan AWS CLI](#)
- [AWS Outposts ejemplos que utilizan AWS CLI](#)
- [AWS Payment Cryptography ejemplos que utilizan AWS CLI](#)
- [AWS Payment Cryptography Ejemplos de planos de datos que utilizan AWS CLI](#)
- [Ejemplos de Amazon Pinpoint que utilizan AWS CLI](#)
- [Ejemplos de Amazon Polly que utilizan AWS CLI](#)
- [Lista de precios de AWS ejemplos que utilizan AWS CLI](#)
- [AWS Private CA ejemplos que utilizan AWS CLI](#)
- [AWS Proton ejemplos que utilizan AWS CLI](#)
- [QLDBejemplos que utilizan AWS CLI](#)
- [RDSEjemplos de Amazon que utilizan AWS CLI](#)
- [Ejemplos RDS de Amazon Data Service que utilizan AWS CLI](#)
- [Ejemplos de Amazon RDS Performance Insights utilizando AWS CLI](#)
- [Ejemplos de Amazon Redshift que utilizan AWS CLI](#)

- [Ejemplos de Amazon Rekognition que utilizan AWS CLI](#)
- [AWS RAM ejemplos que utilizan AWS CLI](#)
- [Ejemplos de Resource Explorer que utilizan AWS CLI](#)
- [Ejemplos de Resource Groups que utilizan AWS CLI](#)
- [Resource Groups: API ejemplos de etiquetado mediante AWS CLI](#)
- [AWS RoboMaker ejemplos que utilizan AWS CLI](#)
- [Ejemplos de Route 53 usando AWS CLI](#)
- [Ejemplos de registro de dominios de Route 53 utilizando AWS CLI](#)
- [Ejemplos de Route 53 Resolver usando AWS CLI](#)
- [Ejemplos de Amazon S3 que utilizan AWS CLI](#)
- [Ejemplos de Amazon S3 Control que utilizan AWS CLI](#)
- [Ejemplos de S3 Glacier utilizando AWS CLI](#)
- [Ejemplos de Secrets Manager usando AWS CLI](#)
- [Ejemplos de Security Hub que utilizan AWS CLI](#)
- [AWS Serverless Application Repository ejemplos que utilizan AWS CLI](#)
- [Ejemplos de Service Catalog que utilizan AWS CLI](#)
- [Ejemplos de Service Quotas que utilizan AWS CLI](#)
- [SESEjemplos de Amazon que utilizan AWS CLI](#)
- [Ejemplos de Shield usando AWS CLI](#)
- [Ejemplos de firmantes que utilizan AWS CLI](#)
- [Ejemplos de Snowball usando AWS CLI](#)
- [SNSEjemplos de Amazon que utilizan AWS CLI](#)
- [SQSEjemplos de Amazon que utilizan AWS CLI](#)
- [Ejemplos de Storage Gateway que utilizan AWS CLI](#)
- [AWS STS ejemplos que utilizan AWS CLI](#)
- [AWS Support ejemplos que utilizan AWS CLI](#)
- [SWFEjemplos de Amazon que utilizan AWS CLI](#)
- [Ejemplos de Systems Manager que utilizan AWS CLI](#)
- [Ejemplos de Amazon Textract usando AWS CLI](#)

- [Ejemplos de Amazon Transcribe utilizando AWS CLI](#)
- [Ejemplos de Amazon Translate utilizando AWS CLI](#)
- [Trusted Advisor ejemplos que utilizan AWS CLI](#)
- [Ejemplos de permisos verificados mediante AWS CLI](#)
- [VPC Ejemplos de celosía utilizando AWS CLI](#)
- [AWS WAF Classic ejemplos que utilizan AWS CLI](#)
- [AWS WAF Classic Regional ejemplos que utilizan AWS CLI](#)
- [AWS WAFV2 ejemplos que utilizan AWS CLI](#)
- [WorkDocs Ejemplos de Amazon que utilizan AWS CLI](#)
- [WorkMail Ejemplos de Amazon que utilizan AWS CLI](#)
- [Ejemplos de Amazon WorkMail Message Flow utilizando AWS CLI](#)
- [WorkSpaces ejemplos que utilizan AWS CLI](#)
- [Ejemplos de rayos X que utilizan AWS CLI](#)

## ACMEjemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with ACM.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **add-tags-to-certificate**

En el siguiente ejemplo de código, se muestra cómo usar `add-tags-to-certificate`.

## AWS CLI

Para añadir etiquetas a un ACM certificado existente

El siguiente comando `add-tags-to-certificate` añade dos etiquetas al certificado especificado. Utilice un espacio para separar varias etiquetas:

```
aws acm add-tags-to-certificate --certificate-arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --tags Key=Admin,Value=Alice Key=Purpose,Value=Website
```

- Para API obtener más información, consulte [AddTagsToCertificate](#) la Referencia de AWS CLI comandos.

## `delete-certificate`

En el siguiente ejemplo de código, se muestra cómo usar `delete-certificate`.

## AWS CLI

Para eliminar un ACM certificado de su cuenta

El siguiente `delete-certificate` comando elimina el certificado con lo especificadoARN:

```
aws acm delete-certificate --certificate-arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

- Para API obtener más información, consulte [DeleteCertificate](#) la Referencia de AWS CLI comandos.

## `describe-certificate`

En el siguiente ejemplo de código, se muestra cómo usar `describe-certificate`.

## AWS CLI

Para recuperar los campos contenidos en un ACM certificado

El siguiente `describe-certificate` comando recupera todos los campos del certificado con lo especificadoARN:

```
aws acm describe-certificate --certificate-arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

Se muestra una salida similar a la siguiente:

```
{
  "Certificate": {
    "CertificateArn":
"arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012",
    "CreatedAt": 1446835267.0,
    "DomainName": "www.example.com",
    "DomainValidationOptions": [
      {
        "DomainName": "www.example.com",
        "ValidationDomain": "www.example.com",
        "ValidationEmails": [
          "hostmaster@example.com",
          "admin@example.com",
          "owner@example.com.whoisprivacyservice.org",
          "tech@example.com.whoisprivacyservice.org",
          "admin@example.com.whoisprivacyservice.org",
          "postmaster@example.com",
          "webmaster@example.com",
          "administrator@example.com"
        ]
      },
      {
        "DomainName": "www.example.net",
        "ValidationDomain": "www.example.net",
        "ValidationEmails": [
          "postmaster@example.net",
          "admin@example.net",
          "owner@example.net.whoisprivacyservice.org",
          "tech@example.net.whoisprivacyservice.org",
          "admin@example.net.whoisprivacyservice.org",
          "hostmaster@example.net",
          "administrator@example.net",
          "webmaster@example.net"
        ]
      }
    ],
    "InUseBy": [],
    "IssuedAt": 1446835815.0,
```



```

    "Issuer": "Amazon",
    "KeyAlgorithm": "RSA-2048",
    "NotAfter": 1478433600.0,
    "NotBefore": 1446768000.0,
    "Serial": "0f:ac:b0:a3:8d:ea:65:52:2d:7d:01:3a:39:36:db:d6",
    "SignatureAlgorithm": "SHA256WITHRSA",
    "Status": "ISSUED",
    "Subject": "CN=www.example.com",
    "SubjectAlternativeNames": [
      "www.example.com",
      "www.example.net"
    ]
  }
}

```

- Para API obtener más información, consulte [DescribeCertificate](#) la Referencia de AWS CLI comandos.

## export-certificate

En el siguiente ejemplo de código, se muestra cómo usar `export-certificate`.

### AWS CLI

Para exportar un certificado privado emitido por una entidad emisora de certificados privada.

El siguiente `export-certificate` comando exporta un certificado privado, una cadena de certificados y una clave privada a la pantalla:

```

aws acm export-certificate --certificate-
arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --
passphrase file://path-to-passphrase-file

```

Para exportar el certificado, la cadena y la clave privada a un archivo local, utilice el siguiente comando:

```

aws acm export-certificate --certificate-
arn arn:aws:acm:region:sccount:certificate/12345678-1234-1234-1234-123456789012 --
passphrase file://path-to-passphrase-file > c:\temp\export.txt

```

- Para API obtener más información, consulte [ExportCertificate](#) la Referencia de AWS CLI comandos.

## get-certificate

En el siguiente ejemplo de código, se muestra cómo usar `get-certificate`.

### AWS CLI

Para recuperar un ACM certificado

El siguiente `get-certificate` comando recupera el certificado de la cadena de certificados especificada ARN y de la cadena de certificados:

```
aws acm get-certificate --certificate-arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

Se muestra una salida similar a la siguiente:

```
{
  "Certificate": "-----BEGIN CERTIFICATE-----
MIICiTCCAFICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBA5TC0lBTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVxHmAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC0lBTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVxHmAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTC2XADZ4nB+BLygVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVvXyUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvjx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----",
  "CertificateChain": "-----BEGIN CERTIFICATE-----
MIICiTCCAFICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBA5TC0lBTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVxHmAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC0lBTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVxHmAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
```

```

21uUSfwfEvySwTC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcVQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUHVvXyUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----",
"-----BEGIN CERTIFICATE-----
MIICiCCAfICCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRawDgYDQVQHEwDQZWF0dGx1MQ8wDQYDQVQKEwZBbWF6
b24xFDASBgNVBAwTC01BTSBDb25zb2x1MRIwEAYDQVQDEwLUZlZG93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcVQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUHVvXyUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----",
"-----BEGIN CERTIFICATE-----
MIICiCCAfICCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRawDgYDQVQHEwDQZWF0dGx1MQ8wDQYDQVQKEwZBbWF6
b24xFDASBgNVBAwTC01BTSBDb25zb2x1MRIwEAYDQVQDEwLUZlZG93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcVQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUHVvXyUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----"
}

```

- Para API obtener más información, consulte [GetCertificate](#) la Referencia de AWS CLI comandos.

## import-certificate

En el siguiente ejemplo de código, se muestra cómo usar `import-certificate`.

### AWS CLI

Para importar un certificado aACM.

El siguiente `import-certificate` comando importa un certificado aACM. Reemplace los nombres de los archivos por los suyos:

```
aws acm import-certificate --certificate file://Certificate.pem --certificate-chain file://CertificateChain.pem --private-key file://PrivateKey.pem
```

- Para API obtener más información, consulte [ImportCertificate](#) la Referencia de AWS CLI comandos.

## list-certificates

En el siguiente ejemplo de código, se muestra cómo usar `list-certificates`.

### AWS CLI

Para ver una lista de los ACM certificados de una AWS cuenta

El siguiente `list-certificates` comando muestra ARNs los certificados de su cuenta:

```
aws acm list-certificates
```

El comando anterior produce un resultado similar al siguiente:

```
{
  "CertificateSummaryList": [
    {
      "CertificateArn":
"arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012",
      "DomainName": "www.example.com"
    },
    {
      "CertificateArn": "arn:aws:acm:region:account:certificate/aaaaaaaa-bbbb-
cccc-dddd-eeeeeeeeeeeeee",

```

```

        "DomainName": "www.example.net"
    }
]
}

```

Puede decidir cuántos certificados quiere mostrar cada vez que llame a `list-certificates`. Por ejemplo, si tiene cuatro certificados y no quiere mostrar más de dos a la vez, establezca el argumento `max-items` en 2, como se muestra en el siguiente ejemplo:

```
aws acm list-certificates --max-items 2
```

Aparecerán dos certificados ARNs y un `NextToken` valor:

```

"CertificateSummaryList": [
  {
    "CertificateArn": "arn:aws:acm:region:account: \
      certificate/12345678-1234-1234-1234-123456789012",
    "DomainName": "www.example.com"
  },
  {
    "CertificateArn": "arn:aws:acm:region:account: \
      certificate/aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee",
    "DomainName": "www.example.net"
  }
],
"NextToken": "9f4d9f69-275a-41fe-b58e-2b837bd9ba48"

```

Para mostrar los dos certificados siguientes de su cuenta, establezca este valor `NextToken` en la próxima llamada:

```
aws acm list-certificates --max-items 2 --next-token 9f4d9f69-275a-41fe-  
b58e-2b837bd9ba48
```

Puede filtrar su salida mediante el argumento `certificate-statuses`. El siguiente comando muestra los certificados que tienen el `VALIDATION` estado `PENDING` \_:

```
aws acm list-certificates --certificate-statuses PENDING_VALIDATION
```

También puede filtrar la salida mediante el argumento `includes`. El siguiente comando muestra los certificados filtrados en las siguientes propiedades. Los certificados que se van a mostrar:

- Specify that the RSA algorithm and a 2048 bit key are used to generate key pairs.
- Contain a Key Usage extension that specifies that the certificates can be used to create digital signatures.
- Contain an Extended Key Usage extension that specifies that the certificates can be used for code signing.

```
aws acm list-certificates --max-items 10 --includes
extendedKeyUsage=CODE_SIGNING,keyUsage=DIGITAL_SIGNATURE,keyTypes=RSA_2048
```

- Para API obtener más información, consulte [ListCertificates](#) la Referencia de AWS CLI comandos.

## list-tags-for-certificate

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-certificate`.

### AWS CLI

Para ver una lista de las etiquetas aplicadas a un ACM certificado

El siguiente comando `list-tags-for-certificate` muestra las etiquetas aplicadas a un certificado de su cuenta:

```
aws acm list-tags-for-certificate --certificate-
arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

El comando anterior produce un resultado similar al siguiente:

```
{
  "Tags": [
    {
      "Value": "Website",
      "Key": "Purpose"
    },
    {
      "Value": "Alice",
      "Key": "Admin"
    }
  ]
}
```

- Para API obtener más información, consulte [ListTagsForCertificate](#) la Referencia de AWS CLI comandos.

## **remove-tags-from-certificate**

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags-from-certificate`.

### AWS CLI

Para eliminar una etiqueta de un ACM certificado

El siguiente comando `remove-tags-from-certificate` elimina las dos etiquetas del certificado especificado. Utilice un espacio para separar varias etiquetas:

```
aws acm remove-tags-from-certificate --certificate-arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --tags Key=Admin,Value=Alice Key=Purpose,Value=Website
```

- Para API obtener más información, consulte [RemoveTagsFromCertificate](#) la Referencia de AWS CLI comandos.

## **request-certificate**

En el siguiente ejemplo de código, se muestra cómo usar `request-certificate`.

### AWS CLI

Para solicitar un ACM certificado nuevo

El siguiente `request-certificate` comando solicita un certificado nuevo para el dominio `www.example.com` mediante DNS la validación:

```
aws acm request-certificate --domain-name www.example.com --validation-method DNS
```

Puede introducir un token de idempotencia para distinguir entre las llamadas a `request-certificate`.

```
aws acm request-certificate --domain-name www.example.com --validation-method DNS --idempotency-token 91adc45q
```

Puede introducir uno o varios nombres alternativos de asunto para solicitar un certificado que proteja más de un dominio de ápex:

```
aws acm request-certificate --domain-name example.com --validation-method DNS --  
idempotency-token 91adc45q --subject-alternative-names www.example.net
```

Puede introducir un nombre alternativo que también se pueda utilizar para acceder a su sitio web:

```
aws acm request-certificate --domain-name example.com --validation-method DNS --  
idempotency-token 91adc45q --subject-alternative-names www.example.com
```

Puede utilizar un asterisco (\*) como comodín para crear un certificado para varios subdominios del mismo dominio:

```
aws acm request-certificate --domain-name example.com --validation-method DNS --  
idempotency-token 91adc45q --subject-alternative-names *.example.com
```

También puede introducir varios nombres alternativos:

```
aws acm request-certificate --domain-name example.com --validation-method DNS --  
subject-alternative-names b.example.com c.example.com d.example.com
```

Si utiliza el correo electrónico para la validación, puede introducir las opciones de validación del dominio para especificar el dominio al que se enviará el correo electrónico de validación:

```
aws acm request-certificate --domain-name example.com --validation-  
method EMAIL --subject-alternative-names www.example.com --domain-validation-  
options DomainName=example.com,ValidationDomain=example.com
```

El siguiente comando cancela el registro de transparencia de certificados cuando solicita un certificado nuevo:

```
aws acm request-certificate --domain-name www.example.com --validation-method DNS --  
options CertificateTransparencyLoggingPreference=DISABLED --idempotency-token 184627
```

- Para API obtener más información, consulte la Referencia [RequestCertificate](#) de AWS CLI comandos.



## resend-validation-email

En el siguiente ejemplo de código, se muestra cómo usar `resend-validation-email`.

### AWS CLI

Para reenviar el correo electrónico de validación de su solicitud ACM de certificado

El siguiente comando `resend-validation-email` indica a la autoridad de certificación de Amazon que envíe un correo electrónico de validación a las direcciones correspondientes:

```
aws acm resend-validation-email --certificate-arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --domain www.example.com --validation-domain example.com
```

- Para API obtener más información, consulte [ResendValidationEmail](#) la Referencia de AWS CLI comandos.

## update-certificate-options

En el siguiente ejemplo de código, se muestra cómo usar `update-certificate-options`.

### AWS CLI

Para actualizar las opciones de certificado

El siguiente `update-certificate-options` comando inhabilita el registro de transparencia de los certificados:

```
aws acm update-certificate-options --certificate-arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --options CertificateTransparencyLoggingPreference=DISABLED
```

- Para API obtener más información, consulte [UpdateCertificateOptions](#) la Referencia de AWS CLI comandos.

## API Ejemplos de puertas de enlace que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface API Gateway.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **create-api-key**

En el siguiente ejemplo de código, se muestra cómo usar `create-api-key`.

AWS CLI

Para crear una API clave que esté habilitada para un escenario API y un escenario existentes

Comando:

```
aws apigateway create-api-key --name 'Dev API Key' --description 'Used for
development' --enabled --stage-keys restApiId='a1b2c3d4e5',stageName='dev'
```

- Para API obtener más información, consulte [CreateApiKey](#) la Referencia de AWS CLI comandos.

### **create-authorizer**

En el siguiente ejemplo de código, se muestra cómo usar `create-authorizer`.

AWS CLI

Ejemplo 1: Para crear un autorizador personalizado de API Gateway basado en fichas para el API

En el siguiente `create-authorizer` ejemplo, se crea un autorizador basado en un token.

```
aws apigateway create-authorizer \
  --rest-api-id 1234123412 \
  --name 'First-Token-Custom-Authorizer' \
```

```

--type TOKEN \
--authorizer-uri 'arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthFunction/invocations' \
--identity-source 'method.request.header.Authorization' \
--authorizer-result-ttl-in-seconds 300

```

Salida:

```

{
  "authType": "custom",
  "name": "First-Token-Custom-Authorizer",
  "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthFunction/invocations",
  "authorizerResultTtlInSeconds": 300,
  "identitySource": "method.request.header.Authorization",
  "type": "TOKEN",
  "id": "z40xj0"
}

```

Ejemplo 2: Para crear un autorizador personalizado de API Gateway basado en grupos de usuarios de Cognito para API

El siguiente `create-authorizer` ejemplo crea un autorizador personalizado de API Gateway basado en grupos de usuarios de Cognito.

```

aws apigateway create-authorizer \
  --rest-api-id 1234123412 \
  --name 'First_Cognito_Custom_Authorizer' \
  --type COGNITO_USER_POOLS \
  --provider-arns 'arn:aws:cognito-idp:us-east-1:123412341234:userpool/us-
east-1_aWcZeQbuD' \
  --identity-source 'method.request.header.Authorization'

```

Salida:

```

{
  "authType": "cognito_user_pools",
  "identitySource": "method.request.header.Authorization",
  "name": "First_Cognito_Custom_Authorizer",
  "providerARNs": [
    "arn:aws:cognito-idp:us-east-1:342398297714:userpool/us-east-1_qWbZzQhzE"
  ],
}

```

```

    "type": "COGNITO_USER_POOLS",
    "id": "5yid1t"
  }

```

Ejemplo 3: Para crear un autorizador personalizado de API Gateway basado en solicitudes para API

El siguiente `create-authorizer` ejemplo crea un autorizador basado en solicitudes.

```

aws apigateway create-authorizer \
  --rest-api-id 1234123412 \
  --name 'First_Request_Custom_Authorizer' \
  --type REQUEST \
  --authorizer-uri 'arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthFunction/invocations' \
  --identity-source 'method.request.header.Authorization,context.accountId' \
  --authorizer-result-ttl-in-seconds 300

```

Salida:

```

{
  "id": "z40xj0",
  "name": "First_Request_Custom_Authorizer",
  "type": "REQUEST",
  "authType": "custom",
  "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthFunction/invocations",
  "identitySource": "method.request.header.Authorization,context.accountId",
  "authorizerResultTtlInSeconds": 300
}

```

- Para API obtener más información, consulte la Referencia de [CreateAuthorizer](#) comandos AWS CLI .

## create-base-path-mapping

En el siguiente ejemplo de código, se muestra cómo usar `create-base-path-mapping`.

AWS CLI

Para crear el mapeo de ruta base para un nombre de dominio personalizado

Comando:

```
aws apigateway create-base-path-mapping --domain-name subdomain.domain.tld --rest-api-id 1234123412 --stage prod --base-path v1
```

- Para API obtener más información, consulte [CreateBasePathMapping](#) la Referencia de AWS CLI comandos.

## create-deployment

En el siguiente ejemplo de código, se muestra cómo usar `create-deployment`.

AWS CLI

Para implementar los recursos configurados API para una nueva etapa

Comando:

```
aws apigateway create-deployment --rest-api-id 1234123412 --stage-name dev --stage-description 'Development Stage' --description 'First deployment to the dev stage'
```

Para implementar los recursos configurados API para una etapa existente

Comando:

```
aws apigateway create-deployment --rest-api-id 1234123412 --stage-name dev --description 'Second deployment to the dev stage'
```

Para implementar los recursos configurados API para una etapa existente con variables de etapa

```
aws apigateway create-deployment -- rest-api-id 1234123412 --stage-name dev --description «Tercer despliegue en la fase de desarrollo» --variables key='value', =' otherKey otherValue
```

- API Para obtener más [CreateDeployment AWS CLI](#) información, consulte la Referencia de comandos.

## create-domain-name

En el siguiente ejemplo de código, se muestra cómo usar `create-domain-name`.

## AWS CLI

Para crear el nombre de dominio personalizado

Comando:

```
aws apigateway create-domain-name --domain-name 'my.domain.tld' --
certificate-name 'my.domain.tld cert' --certificate-arn 'arn:aws:acm:us-
east-1:012345678910:certificate/fb1b9770-a305-495d-aefb-27e5e101ff3'
```

- Para API obtener más información, consulte [CreateDomainName](#) la Referencia de AWS CLI comandos.

## create-model

En el siguiente ejemplo de código, se muestra cómo usar create-model.

## AWS CLI

Para crear un modelo para un API

Comando:

```
aws apigateway create-model --rest-api-id 1234123412 --name 'firstModel' --
description 'The First Model' --content-type 'application/json' --schema
'{"$schema": "http://json-schema.org/draft-04/schema#", "title": "firstModel",
"type": "object", "properties": { "firstProperty" : { "type": "object",
"properties": { "key": { "type": "string" } } } } }'
```

Salida:

```
{
  "contentType": "application/json",
  "description": "The First Model",
  "name": "firstModel",
  "id": "2rzg01",
  "schema": "{ \"$schema\": \"http://json-schema.org/draft-04/schema#\", \"title
\": \"firstModel\", \"type\": \"object\", \"properties\": { \"firstProperty
\": { \"type\": \"object\", \"properties\": { \"key\": { \"type\": \"string
\" } } } } }"
```

- Para API obtener más información, consulte [CreateModel](#) la Referencia de AWS CLI comandos.

## create-resource

En el siguiente ejemplo de código, se muestra cómo usar `create-resource`.

AWS CLI

Para crear un recurso en un API

Comando:

```
aws apigateway create-resource --rest-api-id 1234123412 --parent-id a1b2c3 --path-part 'new-resource'
```

- Para API obtener más información, consulte [CreateResource](#) la Referencia de AWS CLI comandos.

## create-rest-api

En el siguiente ejemplo de código, se muestra cómo usar `create-rest-api`.

AWS CLI

Para crear un API

Comando:

```
aws apigateway create-rest-api --name 'My First API' --description 'This is my first API'
```

Para crear un duplicado API a partir de uno existente API

Comando:

```
aws apigateway create-rest-api --name 'Copy of My First API' --description 'This is a copy of my first API' --clone-from 1234123412
```

- Para API obtener más información, consulte [CreateRestApi](#) la Referencia de AWS CLI comandos.

## create-stage

En el siguiente ejemplo de código, se muestra cómo usar `create-stage`.

### AWS CLI

Para crear una etapa en una API que contendrá una implementación existente

Comando:

```
aws apigateway create-stage --rest-api-id 1234123412 --stage-name 'dev' --  
description 'Development stage' --deployment-id a1b2c3
```

Para crear una etapa en una API que contenga una implementación existente y variables de etapa personalizadas

Comando:

```
aws apigateway create-stage --rest-api-id 1234123412 --stage-name 'dev'  
--description 'Development stage' --deployment-id a1b2c3 --variables  
key='value',otherKey='otherValue'
```

- Para API obtener más información, consulte [CreateStage](#) la Referencia de AWS CLI comandos.

## create-usage-plan-key

En el siguiente ejemplo de código, se muestra cómo usar `create-usage-plan-key`.

### AWS CLI

Asocie una API clave existente a un plan de uso

Comando:

```
aws apigateway create-usage-plan-key --usage-plan-id a1b2c3 --key-type "API_KEY" --  
key-id 4vq3yryqm5
```

- Para API obtener más información, consulte [CreateUsagePlanKey](#) la Referencia de AWS CLI comandos.



## create-usage-plan

En el siguiente ejemplo de código, se muestra cómo usar `create-usage-plan`.

### AWS CLI

Para crear un plan de uso con límites de aceleración y cuota que se restablezca a principios de mes

Comando:

```
aws apigateway create-usage-plan --name "New Usage Plan" --description "A new usage plan" --throttle burstLimit=10,rateLimit=5 --quota limit=500,offset=0,period=MONTH
```

- Para API obtener más información, consulte la Referencia [CreateUsagePlan](#) de AWS CLI comandos.

## delete-api-key

En el siguiente ejemplo de código, se muestra cómo usar `delete-api-key`.

### AWS CLI

Para eliminar una API clave

Comando:

```
aws apigateway delete-api-key --api-key 8bk1k8b1k3sB38D9B3l0enyWT8c09B30lkq0blk
```

- Para API obtener más información, consulte [DeleteApiKey](#) la Referencia de AWS CLI comandos.

## delete-authorizer

En el siguiente ejemplo de código, se muestra cómo usar `delete-authorizer`.

### AWS CLI

Para eliminar un autorizador personalizado en un API

Comando:

```
aws apigateway delete-authorizer --rest-api-id 1234123412 --authorizer-id 7gkfbo
```

- Para API obtener más información, consulte [DeleteAuthorizer](#) la Referencia de AWS CLI comandos.

## delete-base-path-mapping

En el siguiente ejemplo de código, se muestra cómo usar `delete-base-path-mapping`.

### AWS CLI

Para eliminar un mapeo de ruta base para un nombre de dominio personalizado

Comando:

```
aws apigateway delete-base-path-mapping --domain-name 'api.domain.tld' --base-path 'dev'
```

- Para API obtener más información, consulte [DeleteBasePathMapping](#) la Referencia de AWS CLI comandos.

## delete-client-certificate

En el siguiente ejemplo de código, se muestra cómo usar `delete-client-certificate`.

### AWS CLI

Para eliminar un certificado de cliente

Comando:

```
aws apigateway delete-client-certificate --client-certificate-id a1b2c3
```

- Para API obtener más información, consulte [DeleteClientCertificate](#) la Referencia de AWS CLI comandos.

## delete-deployment

En el siguiente ejemplo de código, se muestra cómo usar `delete-deployment`.

## AWS CLI

Para eliminar un despliegue en un API

Comando:

```
aws apigateway delete-deployment --rest-api-id 1234123412 --deployment-id a1b2c3
```

- Para API obtener más información, consulte [DeleteDeployment](#) la Referencia de AWS CLI comandos.

## delete-domain-name

En el siguiente ejemplo de código, se muestra cómo usar delete-domain-name.

### AWS CLI

Para eliminar un nombre de dominio personalizado

Comando:

```
aws apigateway delete-domain-name --domain-name 'api.domain.tld'
```

- Para API obtener más información, consulte [DeleteDomainName](#) la Referencia de AWS CLI comandos.

## delete-integration-response

En el siguiente ejemplo de código, se muestra cómo usar delete-integration-response.

### AWS CLI

Para eliminar una respuesta de integración para un recurso, método y código de estado determinados en un API

Comando:

```
aws apigateway delete-integration-response --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --status-code 200
```

- Para API obtener más información, consulte [DeleteIntegrationResponse](#) la Referencia de AWS CLI comandos.

## delete-integration

En el siguiente ejemplo de código, se muestra cómo usar `delete-integration`.

### AWS CLI

Para eliminar una integración de un recurso y un método determinados en un API

Comando:

```
aws apigateway delete-integration --rest-api-id 1234123412 --resource-id a1b2c3 --  
http-method GET
```

- Para API obtener más información, consulte [DeleteIntegration](#) la Referencia de AWS CLI comandos.

## delete-method-response

En el siguiente ejemplo de código, se muestra cómo usar `delete-method-response`.

### AWS CLI

Para eliminar la respuesta de un método para el recurso, el método y el código de estado determinados en un API

Comando:

```
aws apigateway delete-method-response --rest-api-id 1234123412 --resource-id a1b2c3  
--http-method GET --status-code 200
```

- Para API obtener más información, consulte [DeleteMethodResponse](#) la Referencia de AWS CLI comandos.

## delete-method

En el siguiente ejemplo de código, se muestra cómo usar `delete-method`.

## AWS CLI

Para eliminar un método para un recurso determinado en un API

Comando:

```
aws apigateway delete-method --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET
```

- Para API obtener más información, consulte [DeleteMethod](#) la Referencia de AWS CLI comandos.

## **delete-model**

En el siguiente ejemplo de código, se muestra cómo usar `delete-model`.

## AWS CLI

Para eliminar un modelo en el campo dado API

Comando:

```
aws apigateway delete-model --rest-api-id 1234123412 --model-name 'customModel'
```

- Para API obtener más información, consulte [DeleteModel](#) la Referencia de AWS CLI comandos.

## **delete-resource**

En el siguiente ejemplo de código, se muestra cómo usar `delete-resource`.

## AWS CLI

Para eliminar un recurso de un API

Comando:

```
aws apigateway delete-resource --rest-api-id 1234123412 --resource-id a1b2c3
```

- Para API obtener más información, consulte [DeleteResource](#) la Referencia de AWS CLI comandos.

## delete-rest-api

En el siguiente ejemplo de código, se muestra cómo usar `delete-rest-api`.

### AWS CLI

Para eliminar un API

Comando:

```
aws apigateway delete-rest-api --rest-api-id 1234123412
```

- Para API obtener más información, consulte [DeleteRestApi](#) la Referencia de AWS CLI comandos.

## delete-stage

En el siguiente ejemplo de código, se muestra cómo usar `delete-stage`.

### AWS CLI

Para eliminar una etapa de un API

Comando:

```
aws apigateway delete-stage --rest-api-id 1234123412 --stage-name 'dev'
```

- Para API obtener más información, consulte [DeleteStage](#) la Referencia de AWS CLI comandos.

## delete-usage-plan-key

En el siguiente ejemplo de código, se muestra cómo usar `delete-usage-plan-key`.

### AWS CLI

Para eliminar una API clave de un plan de uso

Comando:

```
aws apigateway delete-usage-plan-key --usage-plan-id a1b2c3 --key-id 1NbjQzMReAkeEQPNAW8r3dXsU2rDD7fc7f2Sipnu
```

- Para API obtener más información, consulte [DeleteUsagePlanKey](#) la Referencia de AWS CLI comandos.

## delete-usage-plan

En el siguiente ejemplo de código, se muestra cómo usar delete-usage-plan.

### AWS CLI

Para eliminar un plan de uso

Comando:

```
aws apigateway delete-usage-plan --usage-plan-id a1b2c3
```

- Para API obtener más información, consulte [DeleteUsagePlan](#) la Referencia de AWS CLI comandos.

## flush-stage-authorizers-cache

En el siguiente ejemplo de código, se muestra cómo usar flush-stage-authorizers-cache.

### AWS CLI

Para vaciar todas las entradas de la caché del autorizador en un escenario

Comando:

```
aws apigateway flush-stage-authorizers-cache --rest-api-id 1234123412 --stage-name dev
```

- Para API obtener más información, consulte la Referencia [FlushStageAuthorizersCache](#) de AWS CLI comandos.

## flush-stage-cache

En el siguiente ejemplo de código, se muestra cómo usar flush-stage-cache.

### AWS CLI

Para vaciar la caché API de una etapa

Comando:

```
aws apigateway flush-stage-cache --rest-api-id 1234123412 --stage-name dev
```

- Para API obtener más información, consulte [FlushStageCache](#) la Referencia de AWS CLI comandos.

## generate-client-certificate

En el siguiente ejemplo de código, se muestra cómo usar `generate-client-certificate`.

AWS CLI

Para crear un certificado del lado del cliente SSL

Comando:

```
aws apigateway generate-client-certificate --description 'My First Client Certificate'
```

- Para API obtener más información, consulte la Referencia [GenerateClientCertificate](#) de AWS CLI comandos.

## get-account

En el siguiente ejemplo de código, se muestra cómo usar `get-account`.

AWS CLI

Para obtener la configuración de la cuenta API Gateway

Comando:

```
aws apigateway get-account
```

Salida:

```
{
  "cloudwatchRoleArn": "arn:aws:iam::123412341234:role/APIGatewayToCloudWatchLogsRole",
```



```
"throttleSettings": {
  "rateLimit": 500.0,
  "burstLimit": 1000
}
```

- Para API obtener más información, consulte [GetAccount](#) la Referencia de AWS CLI comandos.

## get-api-key

En el siguiente ejemplo de código, se muestra cómo usar `get-api-key`.

### AWS CLI

Para obtener la información sobre una API clave específica

Comando:

```
aws apigateway get-api-key --api-key 8bk1k8b11k3sB38D9B310enyWT8c09B30lkq0b1k
```

Salida:

```
{
  "description": "My first key",
  "enabled": true,
  "stageKeys": [
    "a1b2c3d4e5/dev",
    "e5d4c3b2a1/dev"
  ],
  "lastUpdatedDate": 1456184515,
  "createdDate": 1456184452,
  "id": "8bk1k8b11k3sB38D9B310enyWT8c09B30lkq0b1k",
  "name": "My key"
}
```

- Para API obtener más información, consulte [GetApiKey](#) la Referencia de AWS CLI comandos.

## get-api-keys

En el siguiente ejemplo de código, se muestra cómo usar `get-api-keys`.

## AWS CLI

Para obtener la lista de API claves

Comando:

```
aws apigateway get-api-keys
```

Salida:

```
{
  "items": [
    {
      "description": "My first key",
      "enabled": true,
      "stageKeys": [
        "a1b2c3d4e5/dev",
        "e5d4c3b2a1/dev"
      ],
      "lastUpdatedDate": 1456184515,
      "createdDate": 1456184452,
      "id": "8bk1k8b11k3sB38D9B310enyWT8c09B301kq0blk",
      "name": "My key"
    }
  ]
}
```

- Para API obtener más información, consulte [GetApiKeys](#) la Referencia de AWS CLI comandos.

## get-authorizer

En el siguiente ejemplo de código, se muestra cómo usar `get-authorizer`.

## AWS CLI

Para obtener la configuración API de la pasarela por API autorizador

Comando:

```
aws apigateway get-authorizer --rest-api-id 1234123412 --authorizer-id gfi4n3
```

Salida:

```
{
  "authorizerResultTtlInSeconds": 300,
  "name": "MyAuthorizer",
  "type": "TOKEN",
  "identitySource": "method.request.header.Authorization",
  "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:authorizer_function/invocations",
  "id": "gfi4n3"
}
```

- Para API obtener más información, consulte [GetAuthorizer](#) la Referencia de AWS CLI comandos.

## get-authorizers

En el siguiente ejemplo de código, se muestra cómo usar `get-authorizers`.

### AWS CLI

Para obtener la lista de autorizadores de un REST API

Comando:

```
aws apigateway get-authorizers --rest-api-id 1234123412
```

Salida:

```
{
  "items": [
    {
      "name": "MyAuthorizer",
      "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/
functions/arn:aws:lambda:us-west-2:123412341234:function:My_Authorizer_Function/
invocations",
      "authorizerResultTtlInSeconds": 300,
      "identitySource": "method.request.header.Authorization",
      "type": "TOKEN",
      "id": "gfi4n3"
    }
  ]
}
```

- Para API obtener más información, consulte [GetAuthorizers](#) la Referencia de AWS CLI comandos.

## get-base-path-mapping

En el siguiente ejemplo de código, se muestra cómo usar `get-base-path-mapping`.

AWS CLI

Obtención de la asignación de ruta base para un nombre de dominio personalizado

Comando:

```
aws apigateway get-base-path-mapping --domain-name subdomain.domain.tld --base-path v1
```

Salida:

```
{
  "basePath": "v1",
  "restApiId": "1234w4321e",
  "stage": "api"
}
```

- Para API obtener más información, consulte [GetBasePathMapping](#) la Referencia de AWS CLI comandos.

## get-base-path-mappings

En el siguiente ejemplo de código, se muestra cómo usar `get-base-path-mappings`.

AWS CLI

Para obtener las asignaciones de rutas base para un nombre de dominio personalizado

Comando:

```
aws apigateway get-base-path-mappings --domain-name subdomain.domain.tld
```

Salida:

```
{
  "items": [
    {
      "basePath": "(none)",
      "restApiId": "1234w4321e",
      "stage": "dev"
    },
    {
      "basePath": "v1",
      "restApiId": "1234w4321e",
      "stage": "api"
    }
  ]
}
```

- Para API obtener más información, consulte la Referencia [GetBasePathMappings](#) de AWS CLI comandos.

## get-client-certificate

En el siguiente ejemplo de código, se muestra cómo usar `get-client-certificate`.

### AWS CLI

Para obtener un certificado de cliente

Comando:

```
aws apigateway get-client-certificate --client-certificate-id a1b2c3
```

- Para API obtener más información, consulte [GetClientCertificate](#) la Referencia de AWS CLI comandos.

## get-client-certificates

En el siguiente ejemplo de código, se muestra cómo usar `get-client-certificates`.

### AWS CLI

Para obtener una lista de certificados de cliente

Comando:

```
aws apigateway get-client-certificates
```

Salida:

```
{
  "items": [
    {
      "pemEncodedCertificate": "-----BEGIN CERTIFICATE----- <certificate
content> -----END CERTIFICATE-----",
      "clientCertificateId": "a1b2c3",
      "expirationDate": 1483556561,
      "description": "My Client Certificate",
      "createdDate": 1452020561
    }
  ]
}
```

- Para API obtener más información, consulte [GetClientCertificates](#) la Referencia de AWS CLI comandos.

## get-deployment

En el siguiente ejemplo de código, se muestra cómo usar `get-deployment`.

AWS CLI

Para obtener información sobre una implementación

Comando:

```
aws apigateway get-deployment --rest-api-id 1234123412 --deployment-id ztt4m2
```

Salida:

```
{
  "description": "myDeployment",
  "id": "ztt4m2",
  "createdDate": 1455218022
}
```

```
}
```

- Para API obtener más información, consulte [GetDeployment](#) la Referencia de AWS CLI comandos.

## get-deployments

En el siguiente ejemplo de código, se muestra cómo usar `get-deployments`.

### AWS CLI

Para obtener una lista de las implementaciones de un REST API

Comando:

```
aws apigateway get-deployments --rest-api-id 1234123412
```

Salida:

```
{
  "items": [
    {
      "createdDate": 1453797217,
      "id": "0a2b4c",
      "description": "Deployed my API for the first time"
    }
  ]
}
```

- Para API obtener más información, consulte [GetDeployments](#) la Referencia de AWS CLI comandos.

## get-domain-name

En el siguiente ejemplo de código, se muestra cómo usar `get-domain-name`.

### AWS CLI

Para obtener información sobre un nombre de dominio personalizado

Comando:

```
aws apigateway get-domain-name --domain-name api.domain.tld
```

Salida:

```
{
  "domainName": "api.domain.tld",
  "distributionDomainName": "d1a2f3a4c5o6d.cloudfront.net",
  "certificateName": "uploadedCertificate",
  "certificateUploadDate": 1462565487
}
```

- Para API obtener más información, consulte [GetDomainName](#) la Referencia de AWS CLI comandos.

## get-domain-names

En el siguiente ejemplo de código, se muestra cómo usar `get-domain-names`.

AWS CLI

Para obtener una lista de nombres de dominio personalizados

Comando:

```
aws apigateway get-domain-names
```

Salida:

```
{
  "items": [
    {
      "distributionDomainName": "d9511k3l09bkd.cloudfront.net",
      "certificateUploadDate": 1452812505,
      "certificateName": "my_custom_domain-certificate",
      "domainName": "subdomain.domain.tld"
    }
  ]
}
```

- Para API obtener más información, consulte [GetDomainNames](#) la Referencia de AWS CLI comandos.



## get-export

En el siguiente ejemplo de código, se muestra cómo usar `get-export`.

### AWS CLI

Para obtener la plantilla JSON Swagger para un escenario

Comando:

```
aws apigateway get-export --rest-api-id a1b2c3d4e5 --stage-name dev --export-type swagger /path/to/filename.json
```

Para obtener la plantilla JSON Swagger y las extensiones de API Gateway para un escenario

Comando:

```
aws apigateway get-export --parameters extensions='integrations' --rest-api-id a1b2c3d4e5 --stage-name dev --export-type swagger /path/to/filename.json
```

Para obtener la plantilla JSON Swagger y las extensiones de Postman para un escenario

Comando:

```
aws apigateway get-export --parameters extensions='postman' --rest-api-id a1b2c3d4e5 --stage-name dev --export-type swagger /path/to/filename.json
```

- Para API obtener más información, consulte la Referencia de [GetExport](#) comandos AWS CLI .

## get-integration-response

En el siguiente ejemplo de código, se muestra cómo usar `get-integration-response`.

### AWS CLI

Para obtener la configuración de la respuesta de integración para un HTTP método definido en REST API el recurso

Comando:

```
aws apigateway get-integration-response --rest-api-id 1234123412 --resource-id y9h6rt --http-method GET --status-code 200
```

Salida:

```
{
  "statusCode": "200",
  "responseTemplates": {
    "application/json": null
  }
}
```

- Para API obtener más información, consulte [GetIntegrationResponse](#) la Referencia de AWS CLI comandos.

## get-integration

En el siguiente ejemplo de código, se muestra cómo usar `get-integration`.

### AWS CLI

Para obtener la configuración de integración de un HTTP método definido en REST API el recurso

Comando:

```
aws apigateway get-integration --rest-api-id 1234123412 --resource-id y9h6rt --http-method GET
```

Salida:

```
{
  "httpMethod": "POST",
  "integrationResponses": {
    "200": {
      "responseTemplates": {
        "application/json": null
      },
      "statusCode": "200"
    }
  }
}
```

```
  },
  "cacheKeyParameters": [],
  "type": "AWS",
  "uri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:My_Function/invocations",
  "cacheNamespace": "y9h6rt"
}
```

- Para API obtener más información, consulte [GetIntegration](#) la Referencia de AWS CLI comandos.

## get-method-response

En el siguiente ejemplo de código, se muestra cómo usar `get-method-response`.

### AWS CLI

Para obtener la configuración del recurso de respuesta del HTTP método para un método definido en REST API el recurso

Comando:

```
aws apigateway get-method-response --rest-api-id 1234123412 --resource-id y9h6rt --
http-method GET --status-code 200
```

Salida:

```
{
  "responseModels": {
    "application/json": "Empty"
  },
  "statusCode": "200"
}
```

- Para API obtener más información, consulte [GetMethodResponse](#) la Referencia de AWS CLI comandos.

## get-method

En el siguiente ejemplo de código, se muestra cómo usar `get-method`.

## AWS CLI

Para obtener la configuración de recursos del método para un HTTP método definido en REST API el recurso

Comando:

```
aws apigateway get-method --rest-api-id 1234123412 --resource-id y9h6rt --http-method GET
```

Salida:

```
{
  "apiKeyRequired": false,
  "httpMethod": "GET",
  "methodIntegration": {
    "integrationResponses": {
      "200": {
        "responseTemplates": {
          "application/json": null
        },
        "statusCode": "200"
      }
    },
    "cacheKeyParameters": [],
    "uri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:123412341234:function:My_Function/invocations",
    "httpMethod": "POST",
    "cacheNamespace": "y9h6rt",
    "type": "AWS"
  },
  "requestParameters": {},
  "methodResponses": {
    "200": {
      "responseModels": {
        "application/json": "Empty"
      },
      "statusCode": "200"
    }
  },
  "authorizationType": "NONE"
}
```

- Para API obtener más información, consulte [GetMethod](#) la Referencia de AWS CLI comandos.

## get-model-template

En el siguiente ejemplo de código, se muestra cómo usar `get-model-template`.

### AWS CLI

Para obtener la plantilla de mapeo de un modelo definido en un REST API

Comando:

```
aws apigateway get-model-template --rest-api-id 1234123412 --model-name Empty
```

Salida:

```
{
  "value": "#set($inputRoot = $input.path('$'))\n{ }"
}
```

- Para API obtener más información, consulte [GetModelTemplate](#) la Referencia de AWS CLI comandos.

## get-model

En el siguiente ejemplo de código, se muestra cómo usar `get-model`.

### AWS CLI

Para obtener la configuración de un modelo definido en un REST API

Comando:

```
aws apigateway get-model --rest-api-id 1234123412 --model-name Empty
```

Salida:

```
{
  "contentType": "application/json",
  "description": "This is a default empty schema model",
  "name": "Empty",
}
```

```

    "id": "etd5w5",
    "schema": "{\n  \"$schema\": \"http://json-schema.org/draft-04/schema#\",\n  \"title\": \"Empty Schema\", \n  \"type\": \"object\"\n}"
  }

```

- Para API obtener más información, consulte [GetModel](#) la Referencia de AWS CLI comandos.

## get-models

En el siguiente ejemplo de código, se muestra cómo usar `get-models`.

### AWS CLI

Para obtener una lista de los modelos de REST API

Comando:

```
aws apigateway get-models --rest-api-id 1234123412
```

Salida:

```

{
  "items": [
    {
      "description": "This is a default error schema model",
      "schema": "{\n  \"$schema\" : \"http://json-schema.org/draft-04/schema#\",\n  \"title\" : \"Error Schema\", \n  \"type\" : \"object\", \n  \"properties\" : {\n    \"message\" : { \"type\" : \"string\" } \n  } \n}",
      "contentType": "application/json",
      "id": "7tpbze",
      "name": "Error"
    },
    {
      "description": "This is a default empty schema model",
      "schema": "{\n  \"$schema\": \"http://json-schema.org/draft-04/schema#\",\n  \"title\": \"Empty Schema\", \n  \"type\": \"object\"\n}",
      "contentType": "application/json",
      "id": "etd5w5",
      "name": "Empty"
    }
  ]
}

```

- Para API obtener más información, consulte [GetModels](#) la Referencia de AWS CLI comandos.

## get-resource

En el siguiente ejemplo de código, se muestra cómo usar `get-resource`.

### AWS CLI

Para obtener información sobre un recurso

Comando:

```
aws apigateway get-resource --rest-api-id 1234123412 --resource-id zwo0y3
```

Salida:

```
{
  "path": "/path",
  "pathPart": "path",
  "id": "zwo0y3",
  "parentId": "uyokt6ij2g"
}
```

- Para API obtener más información, consulte [GetResource](#) la Referencia de AWS CLI comandos.

## get-resources

En el siguiente ejemplo de código, se muestra cómo usar `get-resources`.

### AWS CLI

Para obtener una lista de recursos para un REST API

Comando:

```
aws apigateway get-resources --rest-api-id 1234123412
```

Salida:

```
{
```

```
    "items": [
      {
        "path": "/resource/subresource",
        "resourceMethods": {
          "POST": {}
        },
        "id": "024ace",
        "pathPart": "subresource",
        "parentId": "ai5b02"
      }
    ]
  }
```

- Para API obtener más información, consulte [GetResources](#) la Referencia de AWS CLI comandos.

## get-rest-api

En el siguiente ejemplo de código, se muestra cómo usar `get-rest-api`.

### AWS CLI

Para obtener información sobre un API

Comando:

```
aws apigateway get-rest-api --rest-api-id 1234123412
```

Salida:

```
{
  "name": "myAPI",
  "id": "o1y243m4f5",
  "createdDate": 1453416433
}
```

- Para API obtener más información, consulte [GetRestApi](#) la Referencia de AWS CLI comandos.

## get-rest-apis

En el siguiente ejemplo de código, se muestra cómo usar `get-rest-apis`.



## AWS CLI

Para obtener una lista de REST APIs

Comando:

```
aws apigateway get-rest-apis
```

Salida:

```
{
  "items": [
    {
      "createdDate": 1438884790,
      "id": "12s44z21rb",
      "name": "My First API"
    }
  ]
}
```

- Para API obtener más información, consulte [GetRestApis](#) la Referencia de AWS CLI comandos.

## get-sdk

En el siguiente ejemplo de código, se muestra cómo usar get-sdk.

## AWS CLI

Para obtener el Android SDK para un REST API escenario

Comando:

```
aws apigateway get-sdk --rest-api-id 1234123412 --stage-name dev --sdk-type android
--parameters
  groupId='com.mycompany',invokerPackage='com.mycompany.clientsdk',artifactId='Mycompany-
client',artifactVersion='1.0.0' /path/to/android_sdk.zip
```

Salida:

```
{
```

```
"contentType": "application/octet-stream",  
"contentDisposition": "attachment; filename=\"android_2016-02-22_23-52Z.zip\""  
}
```

Para conseguir el IOS SDK para un REST API escenario

Comando:

```
aws apigateway get-sdk --rest-api-id 1234123412 --stage-name dev --sdk-  
type objectivec --parameters classPrefix='myprefix' /path/to/iOS_sdk.zip
```

Salida:

```
{  
  "contentType": "application/octet-stream",  
  "contentDisposition": "attachment; filename=\"objectivec_2016-02-22_23-52Z.zip  
  \\  
  }"
```

Para obtener el Javascript SDK de un REST API escenario

Comando:

```
aws apigateway get-sdk --rest-api-id 1234123412 --stage-name dev --sdk-  
type javascript /path/to/javascript_sdk.zip
```

Salida:

```
{  
  "contentType": "application/octet-stream",  
  "contentDisposition": "attachment; filename=\"javascript_2016-02-22_23-52Z.zip  
  \\  
  }"
```

- Para API obtener más información, consulte [GetSdk](#) la Referencia de AWS CLI comandos.

## get-stage

En el siguiente ejemplo de código, se muestra cómo usar get-stage.

## AWS CLI

Para obtener información sobre API la etapa de un

Comando:

```
aws apigateway get-stage --rest-api-id 1234123412 --stage-name dev
```

Salida:

```
{
  "stageName": "dev",
  "cacheClusterSize": "0.5",
  "cacheClusterEnabled": false,
  "cacheClusterStatus": "NOT_AVAILABLE",
  "deploymentId": "rbh1fj",
  "lastUpdatedDate": 1466802961,
  "createdDate": 1460682074,
  "methodSettings": {
    "/*/*": {
      "cacheTtlInSeconds": 300,
      "loggingLevel": "INFO",
      "dataTraceEnabled": false,
      "metricsEnabled": true,
      "unauthorizedCacheControlHeaderStrategy":
"SUCCEED_WITH_RESPONSE_HEADER",
      "throttlingRateLimit": 500.0,
      "cacheDataEncrypted": false,
      "cachingEnabled": false,
      "throttlingBurstLimit": 1000,
      "requireAuthorizationForCacheControl": true
    },
    "~1resource/GET": {
      "cacheTtlInSeconds": 300,
      "loggingLevel": "INFO",
      "dataTraceEnabled": false,
      "metricsEnabled": true,
      "unauthorizedCacheControlHeaderStrategy":
"SUCCEED_WITH_RESPONSE_HEADER",
      "throttlingRateLimit": 500.0,
      "cacheDataEncrypted": false,
      "cachingEnabled": false,
      "throttlingBurstLimit": 1000,
```

```
        "requireAuthorizationForCacheControl": true
      }
    }
  }
```

- Para API obtener más información, consulte [GetStage](#) la Referencia de AWS CLI comandos.

## get-stages

En el siguiente ejemplo de código, se muestra cómo usar `get-stages`.

### AWS CLI

Para obtener la lista de etapas de un REST API

Comando:

```
aws apigateway get-stages --rest-api-id 1234123412
```

Salida:

```
{
  "item": [
    {
      "stageName": "dev",
      "cacheClusterSize": "0.5",
      "cacheClusterEnabled": true,
      "cacheClusterStatus": "AVAILABLE",
      "deploymentId": "123h64",
      "lastUpdatedDate": 1456185138,
      "createdDate": 1453589092,
      "methodSettings": {
        "~1resource~1subresource/POST": {
          "cacheTtlInSeconds": 300,
          "loggingLevel": "INFO",
          "dataTraceEnabled": true,
          "metricsEnabled": true,
          "throttlingRateLimit": 500.0,
          "cacheDataEncrypted": false,
          "cachingEnabled": false,
          "throttlingBurstLimit": 1000
        }
      }
    }
  ]
}
```

```
    }  
  }  
]
```

- Para API obtener más información, consulte [GetStages](#) la Referencia de AWS CLI comandos.

## get-usage-plan-key

En el siguiente ejemplo de código, se muestra cómo usar `get-usage-plan-key`.

### AWS CLI

Para obtener los detalles de una API clave asociada a un plan de uso

Comando:

```
aws apigateway get-usage-plan-key --usage-plan-id a1b2c3 --key-id 1NbjQzMReAkeEQPNAW8r3dXsU2rDD7fc7f2Sipnu
```

- Para API obtener más información, consulte [GetUsagePlanKey](#) la Referencia de AWS CLI comandos.

## get-usage-plan-keys

En el siguiente ejemplo de código, se muestra cómo usar `get-usage-plan-keys`.

### AWS CLI

Para obtener la lista de API claves asociadas a un plan de uso

Comando:

```
aws apigateway get-usage-plan-keys --usage-plan-id a1b2c3
```

- Para API obtener más información, consulte [GetUsagePlanKeys](#) la Referencia de AWS CLI comandos.

## get-usage-plan

En el siguiente ejemplo de código, se muestra cómo usar `get-usage-plan`.

## AWS CLI

Para obtener los detalles de un plan de uso

Comando:

```
aws apigateway get-usage-plan --usage-plan-id a1b2c3
```

- Para API obtener más información, consulte [GetUsagePlan](#) la Referencia de AWS CLI comandos.

## get-usage-plans

En el siguiente ejemplo de código, se muestra cómo usar `get-usage-plans`.

## AWS CLI

Para obtener los detalles de todos los planes de uso

Comando:

```
aws apigateway get-usage-plans
```

- Para API obtener más información, consulte [GetUsagePlans](#) la Referencia de AWS CLI comandos.

## get-usage

En el siguiente ejemplo de código, se muestra cómo usar `get-usage`.

## AWS CLI

Para obtener los detalles de uso de un plan de uso

Comando:

```
aws apigateway get-usage --usage-plan-id a1b2c3 --start-date "2016-08-16" --end-date "2016-08-17"
```

- Para API obtener más información, consulte [GetUsage](#) la Referencia de AWS CLI comandos.

## import-rest-api

En el siguiente ejemplo de código, se muestra cómo usar `import-rest-api`.

### AWS CLI

Para importar una plantilla Swagger y crear una API

Comando:

```
aws apigateway import-rest-api --body 'file:///path/to/API_Swagger_template.json'
```

- Para API obtener más información, consulte la Referencia [ImportRestApi](#) de AWS CLI comandos.

## put-integration-response

En el siguiente ejemplo de código, se muestra cómo usar `put-integration-response`.

### AWS CLI

Para crear una respuesta de integración como respuesta predeterminada con una plantilla de mapeo definida

Comando:

```
aws apigateway put-integration-response --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --status-code 200 --selection-pattern "" --response-templates '{"application/json": "{\"json\": \"template\"}"}'
```

Para crear una respuesta de integración con una expresión regular de 400 y un valor de encabezado definido estáticamente

Comando:

```
aws apigateway put-integration-response --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --status-code 400 --selection-pattern 400 --response-parameters '{"method.response.header.custom-header": ""}'
```

- Para API obtener más información, consulte [PutIntegrationResponse](#) la Referencia de AWS CLI comandos.

## put-integration

En el siguiente ejemplo de código, se muestra cómo usar `put-integration`.

### AWS CLI

Para crear una solicitud MOCK de integración

Comando:

```
aws apigateway put-integration --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --type MOCK --request-templates '{ "application/json": "{\\"statusCode\\": 200}" }'
```

Para crear una solicitud HTTP de integración

Comando:

```
aws apigateway put-integration --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --type HTTP --integration-http-method GET --uri 'https://domain.tld/path'
```

Para crear una solicitud de AWS integración con un punto final de función Lambda

Comando:

```
aws apigateway put-integration --rest-api-id 1234123412 --resource-id a1b2c3 --http-method GET --type AWS --integration-http-method POST --uri 'arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:123412341234:function:function_name/invocations'
```

- Para API obtener más información, consulte [PutIntegration](#) la Referencia de AWS CLI comandos.

## put-method-response

En el siguiente ejemplo de código, se muestra cómo usar `put-method-response`.

### AWS CLI

Creación de una respuesta de método en el código de estado especificado con un encabezado de respuesta de método personalizado



Comando:

```
aws apigateway put-method-response --rest-api-id 1234123412 --  
resource-id a1b2c3 --http-method GET --status-code 400 --response-  
parameters "method.response.header.custom-header=false"
```

- Para API obtener más información, consulte [PutMethodResponse](#) la Referencia de AWS CLI comandos.

## put-method

En el siguiente ejemplo de código, se muestra cómo usar `put-method`.

AWS CLI

Para crear un método para un recurso en un API encabezado de solicitud de método personalizado, sin autorización ni API clave

Comando:

```
aws apigateway put-method --rest-api-id 1234123412 --resource-id a1b2c3 --  
http-method PUT --authorization-type "NONE" --no-api-key-required --request-  
parameters "method.request.header.custom-header=false"
```

- Para API obtener más información, consulte [PutMethod](#) la Referencia de AWS CLI comandos.

## put-rest-api

En el siguiente ejemplo de código, se muestra cómo usar `put-rest-api`.

AWS CLI

Para sobrescribir una existente API mediante una plantilla Swagger

Comando:

```
aws apigateway put-rest-api --rest-api-id 1234123412 --mode overwrite --body  
'file:///path/to/API_Swagger_template.json'
```

Para fusionar una plantilla Swagger con una existente API

Comando:

```
aws apigateway put-rest-api --rest-api-id 1234123412 --mode merge --body 'fileb:///path/to/API_Swagger_template.json'
```

- Para API obtener más información, consulte la Referencia [PutRestApi](#) de AWS CLI comandos.

## test-invoke-authorizer

En el siguiente ejemplo de código, se muestra cómo usar `test-invoke-authorizer`.

AWS CLI

Para realizar una prueba, invoque una solicitud a un autorizador personalizado que incluya el encabezado y el valor necesarios

Comando:

```
aws apigateway test-invoke-authorizer --rest-api-id 1234123412 --authorizer-id 5yid1t --headers Authorization='Value'
```

- Para API obtener más información, consulte la Referencia [TestInvokeAuthorizer](#) de AWS CLI comandos.

## test-invoke-method

En el siguiente ejemplo de código, se muestra cómo usar `test-invoke-method`.

AWS CLI

Para probar, invoque el recurso raíz en an API realizando una solicitud GET

Comando:

```
aws apigateway test-invoke-method --rest-api-id 1234123412 --resource-id av15sg8fw8 --http-method GET --path-with-query-string '/'
```

Para probar, invoque un subrecurso en an API realizando una GET solicitud con un valor de parámetro de ruta especificado

Comando:

```
aws apigateway test-invoke-method --rest-api-id 1234123412 --resource-id 3gapai --  
http-method GET --path-with-query-string '/pets/1'
```

- Para API obtener más información, consulte la Referencia [TestInvokeMethod](#) de AWS CLI comandos.

## update-account

En el siguiente ejemplo de código, se muestra cómo usar `update-account`.

### AWS CLI

Para cambiar la IAM función ARN de registro a CloudWatch Logs

Comando:

```
aws apigateway update-account --patch-operations op='replace',path='/  
cloudwatchRoleArn',value='arn:aws:iam::123412341234:role/APIGatewayToCloudWatchLogs'
```

Salida:

```
{  
  "cloudwatchRoleArn": "arn:aws:iam::123412341234:role/  
APIGatewayToCloudWatchLogs",  
  "throttleSettings": {  
    "rateLimit": 1000.0,  
    "burstLimit": 2000  
  }  
}
```

- Para API obtener más información, consulte [UpdateAccount](#) la Referencia de AWS CLI comandos.

## update-api-key

En el siguiente ejemplo de código, se muestra cómo usar `update-api-key`.

### AWS CLI

Para cambiar el nombre de una API clave

**Comando:**

```
aws apigateway update-api-key --api-key sNvjQDMReA1eEQPNAW8r37XsU2rDD7fc7m2SiMnu --  
patch-operations op='replace',path='/name',value='newName'
```

**Salida:**

```
{  
  "description": "currentDescription",  
  "enabled": true,  
  "stageKeys": [  
    "41t2j324r5/dev"  
  ],  
  "lastUpdatedDate": 1470086052,  
  "createdDate": 1445460347,  
  "id": "sNvjQDMReA1vEQPNzW8r3dXsU2rrD7fcjm2SiMnu",  
  "name": "newName"  
}
```

**Para deshabilitar la API clave****Comando:**

```
aws apigateway update-api-key --api-key sNvjQDMReA1eEQPNAW8r37XsU2rDD7fc7m2SiMnu --  
patch-operations op='replace',path='/enabled',value='false'
```

**Salida:**

```
{  
  "description": "currentDescription",  
  "enabled": false,  
  "stageKeys": [  
    "41t2j324r5/dev"  
  ],  
  "lastUpdatedDate": 1470086052,  
  "createdDate": 1445460347,  
  "id": "sNvjQDMReA1vEQPNzW8r3dXsU2rrD7fcjm2SiMnu",  
  "name": "newName"  
}
```

- Para API obtener más información, consulte [UpdateApiKey](#) la Referencia de AWS CLI comandos.

## update-authorizer

En el siguiente ejemplo de código, se muestra cómo usar `update-authorizer`.

### AWS CLI

Para cambiar el nombre del autorizador personalizado

Comando:

```
aws apigateway update-authorizer --rest-api-id 1234123412 --authorizer-id gfi4n3 --patch-operations op='replace',path='/name',value='testAuthorizer'
```

Salida:

```
{
  "authType": "custom",
  "name": "testAuthorizer",
  "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:123412341234:function:customAuthorizer/invocations",
  "authorizerResultTtlInSeconds": 300,
  "identitySource": "method.request.header.Authorization",
  "type": "TOKEN",
  "id": "gfi4n3"
}
```

Para cambiar la función Lambda que invoca el autorizador personalizado

Comando:

```
aws apigateway update-authorizer --rest-api-id 1234123412 --authorizer-id gfi4n3 --patch-operations op='replace',path='/authorizerUri',value='arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:123412341234:function:newAuthorizer/invocations'
```

Salida:

```
{
  "authType": "custom",
  "name": "testAuthorizer",
  "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:123412341234:function:newAuthorizer/invocations",
  "authorizerResultTtlInSeconds": 300,
}
```

```
"identitySource": "method.request.header.Authorization",
"type": "TOKEN",
"id": "gfi4n3"
}
```

- Para API obtener más información, consulte la Referencia [UpdateAuthorizer](#) de AWS CLI comandos.

## update-base-path-mapping

En el siguiente ejemplo de código, se muestra cómo usar `update-base-path-mapping`.

### AWS CLI

Para cambiar la ruta base de un nombre de dominio personalizado

Comando:

```
aws apigateway update-base-path-mapping --domain-name api.domain.tld --base-path prod --patch-operations op='replace',path='/basePath',value='v1'
```

Salida:

```
{
  "basePath": "v1",
  "restApiId": "1234123412",
  "stage": "api"
}
```

- Para API obtener más información, consulte [UpdateBasePathMapping](#) la Referencia de AWS CLI comandos.

## update-client-certificate

En el siguiente ejemplo de código, se muestra cómo usar `update-client-certificate`.

### AWS CLI

Para actualizar la descripción de un certificado de cliente

Comando:

```
aws apigateway update-client-certificate --client-certificate-id a1b2c3 --patch-operations op='replace',path='/description',value='My new description'
```

- Para API obtener más información, consulte [UpdateClientCertificate](#) la Referencia de AWS CLI comandos.

## update-deployment

En el siguiente ejemplo de código, se muestra cómo usar update-deployment.

### AWS CLI

Para cambiar la descripción de una implementación

Comando:

```
aws apigateway update-deployment --rest-api-id 1234123412 --deployment-id ztt4m2 --patch-operations op='replace',path='/description',value='newDescription'
```

Salida:

```
{
  "description": "newDescription",
  "id": "ztt4m2",
  "createdDate": 1455218022
}
```

- Para API obtener más información, consulte [UpdateDeployment](#) la Referencia de AWS CLI comandos.

## update-domain-name

En el siguiente ejemplo de código, se muestra cómo usar update-domain-name.

### AWS CLI

Para cambiar el nombre del certificado por un nombre de dominio personalizado

En el siguiente update-domain-name ejemplo, se cambia el nombre del certificado de un dominio personalizado.

```
aws apigateway update-domain-name \
  --domain-name api.domain.tld \
  --patch-operations op='replace',path='/certificateArn',value='arn:aws:acm:us-
west-2:111122223333:certificate/CERTEXAMPLE123EXAMPLE'
```

Salida:

```
{
  "domainName": "api.domain.tld",
  "distributionDomainName": "d123456789012.cloudfront.net",
  "certificateArn": "arn:aws:acm:us-west-2:111122223333:certificate/
CERTEXAMPLE123EXAMPLE",
  "certificateUploadDate": 1462565487
}
```

Para obtener más información, consulte [Configurar un nombre de dominio personalizado para un servidor API en API Gateway](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [UpdateDomainName](#) la Referencia de AWS CLI comandos.

## update-integration-response

En el siguiente ejemplo de código, se muestra cómo usar `update-integration-response`.

AWS CLI

Para cambiar el encabezado de una respuesta de integración para que tenga un mapeo estático de '\*'

Comando:

```
aws apigateway update-integration-response --rest-api-id 1234123412 --
resource-id 3gapai --http-method GET --status-code 200 --patch-operations
op='replace',path='/responseParameters/method.response.header.Access-Control-Allow-
Origin',value='''''*''''''
```

Salida:

```
{
  "statusCode": "200",
```



```
"responseParameters": {
  "method.response.header.Access-Control-Allow-Origin": "*"
}
```

Para eliminar un encabezado de respuesta de integración

Comando:

```
aws apigateway update-integration-response --rest-api-id 1234123412 --resource-id 3gapai --http-method GET --status-code 200 --patch-operations op='remove',path='/responseParameters/method.response.header.Access-Control-Allow-Origin'
```

- Para API obtener más información, consulte [UpdateIntegrationResponse](#) la Referencia de AWS CLI comandos.

## update-integration

En el siguiente ejemplo de código, se muestra cómo usar update-integration.

AWS CLI

Para añadir la plantilla de mapeo «Content-Type: application/json» configurada con Input Passthrough

Comando:

```
aws apigateway update-integration \  
  --rest-api-id a1b2c3d4e5 \  
  --resource-id a1b2c3 \  
  --http-method POST \  
  --patch-operations "op='add',path='/requestTemplates/application~1json'"
```

Para actualizar (reemplazar) la plantilla de mapeo «Content-Type: application/json» configurada por una plantilla personalizada

Comando:

```
aws apigateway update-integration \  
  --rest-api-id a1b2c3d4e5 \  
  --resource-id a1b2c3 \  
  --http-method POST \  
  --patch-operations "op='replace',path='/requestTemplates/application~1json'"
```

```
--patch-operations "op='replace',path='/requestTemplates/
application~1json',value='{\"example\": \"json\"}'"
```

Para actualizar (reemplazar) una plantilla personalizada asociada a «Content-Type: application/json» por Input Passthrough

Comando:

```
aws apigateway update-integration \
  --rest-api-id a1b2c3d4e5 \
  --resource-id a1b2c3 \
  --http-method POST \
  --patch-operations "op='replace',path='requestTemplates/application~1json'"
```

Para eliminar la plantilla de mapeo «Content-Type: application/json»

Comando:

```
aws apigateway update-integration \
  --rest-api-id a1b2c3d4e5 \
  --resource-id a1b2c3 \
  --http-method POST \
  --patch-operations "op='remove',path='/requestTemplates/application~1json'"
```

- Para API obtener más información, consulte la Referencia de comandos.

[UpdateIntegration](#)AWS CLI

## update-method-response

En el siguiente ejemplo de código, se muestra cómo usar `update-method-response`.

AWS CLI

Para crear un nuevo encabezado de respuesta de método para la respuesta 200 de un método y definirlo como no obligatorio (predeterminado)

Comando:

```
aws apigateway update-method-response --rest-api-id 1234123412 --resource-
id a1b2c3 --http-method GET --status-code 200 --patch-operations op="add",path="/
responseParameters/method.response.header.custom-header",value="false"
```

Para eliminar un modelo de respuesta para la respuesta 200 en un método

Comando:

```
aws apigateway update-method-response --rest-api-id 1234123412 --resource-id a1b2c3
--http-method GET --status-code 200 --patch-operations op="remove",path="/
responseModels/application~1json"
```

- Para API obtener más información, consulte [UpdateMethodResponse](#) la Referencia de AWS CLI comandos.

## update-method

En el siguiente ejemplo de código, se muestra cómo usar `update-method`.

AWS CLI

Ejemplo 1: Modificar un método para que requiera una API clave

El siguiente `update-method` ejemplo modifica el método para que requiera una API clave.

```
aws apigateway update-method \
--rest-api-id 1234123412 \
--resource-id a1b2c3 \
--http-method GET \
--patch-operations op="replace",path="/apiKeyRequired",value="true"
```

Salida:

```
{
  "httpMethod": "GET",
  "authorizationType": "NONE",
  "apiKeyRequired": true,
  "methodResponses": {
    "200": {
      "statusCode": "200",
      "responseModels": {}
    }
  },
  "methodIntegration": {
    "type": "AWS",
    "httpMethod": "POST",
```

```

    "uri": "arn:aws:apigateway:us-east-1:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-east-1:123456789111:function:hello-world/invocations",
    "passthroughBehavior": "WHEN_NO_MATCH",
    "contentHandling": "CONVERT_TO_TEXT",
    "timeoutInMillis": 29000,
    "cacheNamespace": "h7i8j9",
    "cacheKeyParameters": [],
    "integrationResponses": {
      "200": {
        "statusCode": "200",
        "responseTemplates": {}
      }
    }
  }
}

```

## Ejemplo 2: Modificar un método para que requiera autorización IAM

El siguiente update-method ejemplo modifica el método para que requiera IAM autorización.

```

aws apigateway update-method \
  --rest-api-id 1234123412 \
  --resource-id a1b2c3 \
  --http-method GET \
  --patch-operations op="replace",path="/authorizationType",value="AWS_IAM"

```

Salida:

```

{
  "httpMethod": "GET",
  "authorizationType": "AWS_IAM",
  "apiKeyRequired": false,
  "methodResponses": {
    "200": {
      "statusCode": "200",
      "responseModels": {}
    }
  },
  "methodIntegration": {
    "type": "AWS",
    "httpMethod": "POST",
    "uri": "arn:aws:apigateway:us-east-1:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-east-1:123456789111:function:hello-world/invocations",

```

```

    "passthroughBehavior": "WHEN_NO_MATCH",
    "contentHandling": "CONVERT_TO_TEXT",
    "timeoutInMillis": 29000,
    "cacheNamespace": "h7i8j9",
    "cacheKeyParameters": [],
    "integrationResponses": {
      "200": {
        "statusCode": "200",
        "responseTemplates": {}
      }
    }
  }
}

```

### Ejemplo 3: Modificar un método para que requiera la autorización de Lambda

En el siguiente `update-method` ejemplo, se modifica el método para adaptarlo a la autorización Lambda requerida.

```

aws apigateway update-method --rest-api-id 1234123412 \
  --resource-id a1b2c3 \
  --http-method GET \
  --patch-operations op="replace",path="/authorizationType",value="CUSTOM"
  op="replace",path="/authorizerId",value="e4f5g6"

```

Salida:

```

{
  "httpMethod": "GET",
  "authorizationType": "CUSTOM",
  "authorizerId": "e4f5g6",
  "apiKeyRequired": false,
  "methodResponses": {
    "200": {
      "statusCode": "200",
      "responseModels": {}
    }
  },
  "methodIntegration": {
    "type": "AWS",
    "httpMethod": "POST",
    "uri": "arn:aws:apigateway:us-east-1:lambda:path/2015-03-31/functions/arn:aws:lambda:us-east-1:123456789111:function:hello-world/invocations",

```

```

    "passthroughBehavior": "WHEN_NO_MATCH",
    "contentHandling": "CONVERT_TO_TEXT",
    "timeoutInMillis": 29000,
    "cacheNamespace": "h7i8j9",
    "cacheKeyParameters": [],
    "integrationResponses": {
      "200": {
        "statusCode": "200",
        "responseTemplates": {}
      }
    }
  }
}

```

Para obtener más información, consulte [Crear, configurar y probar planes de uso mediante la API puerta de enlace y Control CLI REST API y administración del acceso a una puerta de API enlace REST API interna en](#) la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [UpdateMethod](#) la Referencia de AWS CLI comandos.

## update-model

En el siguiente ejemplo de código, se muestra cómo usar `update-model`.

### AWS CLI

Para cambiar la descripción de un modelo en un API

Comando:

```
aws apigateway update-model --rest-api-id 1234123412 --model-name 'Empty' --patch-operations op=replace,path=/description,value='New Description'
```

Para cambiar el esquema de un modelo en un API

Comando:

```
aws apigateway update-model --rest-api-id 1234123412 --model-name 'Empty' --patch-operations op=replace,path=/schema,value='{ \"$schema\": \"http://json-schema.org/draft-04/schema#\", \"title\": \"Empty Schema\", \"type\": \"object\" }''
```

- Para API obtener más información, consulte [UpdateModel](#) la Referencia de AWS CLI comandos.

## update-resource

En el siguiente ejemplo de código, se muestra cómo usar `update-resource`.

### AWS CLI

Para mover un recurso y colocarlo debajo de un recurso principal diferente en un API

Comando:

```
aws apigateway update-resource --rest-api-id 1234123412 --resource-id 1a2b3c --  
patch-operations op=replace,path=/parentId,value='3c2b1a'
```

Salida:

```
{  
  "path": "/resource",  
  "pathPart": "resource",  
  "id": "1a2b3c",  
  "parentId": "3c2b1a"  
}
```

Para cambiar el nombre de un recurso (`pathPart`) en un API

Comando:

```
aws apigateway update-resource --rest-api-id 1234123412 --resource-id 1a2b3c --  
patch-operations op=replace,path=/pathPart,value=newresourcename
```

Salida:

```
{  
  "path": "/newresourcename",  
  "pathPart": "newresourcename",  
  "id": "1a2b3c",  
  "parentId": "3c2b1a"  
}
```

- Para API obtener más información, consulte [UpdateResource](#) la Referencia de AWS CLI comandos.

## update-rest-api

En el siguiente ejemplo de código, se muestra cómo usar `update-rest-api`.

### AWS CLI

Para cambiar el nombre de un API

Comando:

```
aws apigateway update-rest-api --rest-api-id 1234123412 --patch-operations  
op=replace,path=/name,value='New Name'
```

Para cambiar la descripción de un API

Comando:

```
aws apigateway update-rest-api --rest-api-id 1234123412 --patch-operations  
op=replace,path=/description,value='New Description'
```

- Para API obtener más información, consulte [UpdateRestApi](#) la Referencia de AWS CLI comandos.

## update-stage

En el siguiente ejemplo de código, se muestra cómo usar `update-stage`.

### AWS CLI

Ejemplo 1: Para anular la configuración de escenario de un recurso y un método

El siguiente `update-stage` ejemplo anula la configuración del escenario y desactiva el registro completo de solicitudes y respuestas para un recurso y un método específicos.

```
aws apigateway update-stage \  
--rest-api-id 1234123412 \  
--patch-operations op=replace,path=/name,value='New Name'
```



```
--stage-name 'dev' \  
--patch-operations op=replace,path=~1resourceName/GET/logging/  
dataTrace,value=false
```

Salida:

```
{  
  "deploymentId": "5ubd17",  
  "stageName": "dev",  
  "cacheClusterEnabled": false,  
  "cacheClusterStatus": "NOT_AVAILABLE",  
  "methodSettings": {  
    "~1resourceName/GET": {  
      "metricsEnabled": false,  
      "dataTraceEnabled": false,  
      "throttlingBurstLimit": 5000,  
      "throttlingRateLimit": 10000.0,  
      "cachingEnabled": false,  
      "cacheTtlInSeconds": 300,  
      "cacheDataEncrypted": false,  
      "requireAuthorizationForCacheControl": true,  
      "unauthorizedCacheControlHeaderStrategy": "SUCCEED_WITH_RESPONSE_HEADER"  
    }  
  },  
  "tracingEnabled": false,  
  "createdDate": "2022-07-18T10:11:18-07:00",  
  "lastUpdatedDate": "2022-07-18T10:19:04-07:00"  
}
```

Para obtener más información, consulte [Configuración de un escenario para un REST API](#) en la Guía para desarrolladores de Amazon API Gateway.

Ejemplo 2: Para actualizar la configuración del escenario para todos los recursos y métodos de un API escenario

El siguiente `update-stage` ejemplo activa el registro completo de solicitudes y respuestas para todos los recursos y métodos de una etapa. API

```
aws apigateway update-stage \  
--rest-api-id 1234123412 \  
--stage-name 'dev' \  
--patch-operations 'op=replace,path=/*/*/logging/dataTrace,value=true'
```

**Salida:**

```
{
  "deploymentId": "5ubd17",
  "stageName": "dev",
  "cacheClusterEnabled": false,
  "cacheClusterStatus": "NOT_AVAILABLE",
  "methodSettings": {
    "/*/*": {
      "metricsEnabled": false,
      "dataTraceEnabled": true,
      "throttlingBurstLimit": 5000,
      "throttlingRateLimit": 10000.0,
      "cachingEnabled": false,
      "cacheTtlInSeconds": 300,
      "cacheDataEncrypted": false,
      "requireAuthorizationForCacheControl": true,
      "unauthorizedCacheControlHeaderStrategy": "SUCCEED_WITH_RESPONSE_HEADER"
    }
  },
  "tracingEnabled": false,
  "createdDate": "2022-07-18T10:11:18-07:00",
  "lastUpdatedDate": "2022-07-18T10:31:04-07:00"
}
```

Para obtener más información, consulte [Configuración de un escenario para un REST API](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [UpdateStage](#) la Referencia de AWS CLI comandos.

**update-usage-plan**

En el siguiente ejemplo de código, se muestra cómo usar `update-usage-plan`.

**AWS CLI**

Para cambiar el período definido en un plan de uso

Comando:

```
aws apigateway update-usage-plan --usage-plan-id a1b2c3 --patch-operations
  op="replace",path="/quota/period",value="MONTH"
```

Para cambiar el límite de cuota definido en un plan de uso

Comando:

```
aws apigateway update-usage-plan --usage-plan-id a1b2c3 --patch-operations  
op="replace",path="/quota/limit",value="500"
```

Para cambiar el límite de velocidad de aceleración definido en un plan de uso

Comando:

```
aws apigateway update-usage-plan --usage-plan-id a1b2c3 --patch-operations  
op="replace",path="/throttle/rateLimit",value="10"
```

Para cambiar el límite de ráfaga del acelerador definido en un plan de uso

Comando:

```
aws apigateway update-usage-plan --usage-plan-id a1b2c3 --patch-operations  
op="replace",path="/throttle/burstLimit",value="20"
```

- Para API obtener más información, consulte [UpdateUsagePlan](#) la Referencia de AWS CLI comandos.

## update-usage

En el siguiente ejemplo de código, se muestra cómo usar `update-usage`.

### AWS CLI

Para modificar temporalmente la cuota de una API clave durante el período actual definido en el plan de uso

Comando:

```
aws apigateway update-usage --usage-plan-id a1b2c3 --key-  
id 1NbjQzMReAkeEQPNAW8r3dXsU2rDD7fc7f2Sipnu --patch-operations op="replace",path="/  
remaining",value="50"
```

- Para API obtener más información, consulte [UpdateUsage](#) la Referencia de AWS CLI comandos.

## APIGateway HTTP y WebSocket API ejemplos de uso AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el AWS Command Line Interface uso de API Gateway HTTP y WebSocket API.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-api-mapping**

En el siguiente ejemplo de código, se muestra cómo usar `create-api-mapping`.

AWS CLI

Para crear un API mapeo para un API

El siguiente `create-api-mapping` ejemplo asigna la `test` etapa de un API a la `/myApi` ruta del nombre de dominio `regional.example.com` personalizado.

```
aws apigatewayv2 create-api-mapping \  
  --domain-name regional.example.com \  
  --api-mapping-key myApi \  
  --api-id a1b2c3d4 \  
  --stage test
```

Salida:

```
{  
  "ApiId": "a1b2c3d4",  
  "ApiMappingId": "0qzs2sy7bh",
```

```
"ApiMappingKey": "myApi"
"Stage": "test"
}
```

Para obtener más información, consulte [Configuración de un nombre de dominio personalizado regional en API Gateway en](#) la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [CreateApiMapping](#) la Referencia de AWS CLI comandos.

## create-api

En el siguiente ejemplo de código, se muestra cómo usar `create-api`.

### AWS CLI

Para crear un HTTP API

En el siguiente `create-api` ejemplo se crea un HTTP API mediante la creación rápida. Puede usar Quick Create para crear una API con una AWS Lambda o una HTTP integración, una ruta global predeterminada y una etapa predeterminada que esté configurada para implementar los cambios automáticamente. El siguiente comando usa quick create para crear una HTTP API que se integre con una función Lambda.

```
aws apigatewayv2 create-api \
  --name my-http-api \
  --protocol-type HTTP \
  --target arn:aws:lambda:us-west-2:123456789012:function:my-lambda-function
```

Salida:

```
{
  "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
  "ApiId": "a1b2c3d4",
  "ApiKeySelectionExpression": "$request.header.x-api-key",
  "CreateDate": "2020-04-08T19:05:45+00:00",
  "Name": "my-http-api",
  "ProtocolType": "HTTP",
  "RouteSelectionExpression": "$request.method $request.path"
}
```

Para obtener más información, consulte [Desarrollo de un HTTP API in API Gateway](#) en la Guía para desarrolladores de Amazon API Gateway.

Para crear un WebSocket API

En el siguiente `create-api` ejemplo, se crea un WebSocket API con el nombre especificado.

```
aws apigatewayv2 create-api \  
  --name "myWebSocketApi" \  
  --protocol-type WEBSOCKET \  
  --route-selection-expression '$request.body.action'
```

Salida:

```
{  
  "ApiKeySelectionExpression": "$request.header.x-api-key",  
  "Name": "myWebSocketApi",  
  "CreateDate": "2018-11-15T06:23:51Z",  
  "ProtocolType": "WEBSOCKET",  
  "RouteSelectionExpression": "'$request.body.action'",  
  "ApiId": "aabbccdde"  
}
```

Para obtener más información, consulte [Create a WebSocket API in API Gateway](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [CreateApi](#) la Referencia de AWS CLI comandos.

## **create-authorizer**

En el siguiente ejemplo de código, se muestra cómo usar `create-authorizer`.

### AWS CLI

Para crear un JWT autorizador para un HTTP API

En el siguiente `create-authorizer` ejemplo, se crea un JWT autorizador que utiliza Amazon Cognito como proveedor de identidad.

```
aws apigatewayv2 create-authorizer \  
  --name my-jwt-authorizer \  
  --api-id a1b2c3d4 \  
  --identity-provider cognito-identity-provider
```

```
--authorizer-type JWT \  
--identity-source '$request.header.Authorization' \  
--jwt-configuration Audience=123456abc,Issuer=https://cognito-idp.us-west-2.amazonaws.com/us-west-2_abc123
```

Salida:

```
{  
  "AuthorizerId": "a1b2c3",  
  "AuthorizerType": "JWT",  
  "IdentitySource": [  
    "$request.header.Authorization"  
  ],  
  "JwtConfiguration": {  
    "Audience": [  
      "123456abc"  
    ],  
    "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-west-2_abc123"  
  },  
  "Name": "my-jwt-authorizer"  
}
```

Para obtener más información, consulte [Control del acceso HTTP APIs con JWT autorizadores](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [CreateAuthorizer](#) la Referencia de AWS CLI comandos.

## create-deployment

En el siguiente ejemplo de código, se muestra cómo usar create-deployment.

AWS CLI

Para crear un despliegue para un API

El siguiente create-deployment ejemplo crea un despliegue para un API y lo asocia a la dev etapa del API.

```
aws apigatewayv2 create-deployment \  
  --api-id a1b2c3d4 \  
  --stage-name dev
```

Salida:

```
{
  "AutoDeployed": false,
  "CreatedDate": "2020-04-06T23:38:08Z",
  "DeploymentId": "531z91",
  "DeploymentStatus": "DEPLOYED"
}
```

Para obtener más información, consulte la [APIimplementación](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [CreateDeployment](#) la Referencia de AWS CLI comandos.

## create-domain-name

En el siguiente ejemplo de código, se muestra cómo usar `create-domain-name`.

### AWS CLI

Para crear un nombre de dominio personalizado

El siguiente `create-domain-name` ejemplo crea un nombre de dominio regional personalizado para unAPI.

```
aws apigatewayv2 create-domain-name \
  --domain-name regional.example.com \
  --domain-name-configurations CertificateArn=arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678
```

Salida:

```
{
  "ApiMappingSelectionExpression": "$request.basepath",
  "DomainName": "regional.example.com",
  "DomainNameConfigurations": [
    {
      "ApiGatewayDomainName": "d-id.execute-api.us-west-2.amazonaws.com",
      "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",

```



```

        "EndpointType": "REGIONAL",
        "HostedZoneId": "123456789111",
        "SecurityPolicy": "TLS_1_2",
        "DomainNameStatus": "AVAILABLE"
    }
]
}

```

Para obtener más información, consulte [Configuración de un nombre de dominio personalizado regional en API Gateway en](#) la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [CreateDomainName](#) la Referencia de AWS CLI comandos.

## create-integration

En el siguiente ejemplo de código, se muestra cómo usar `create-integration`.

### AWS CLI

Para crear una WebSocket API integración

El siguiente `create-integration` ejemplo crea una integración simulada para un WebSocket API.

```

aws apigatewayv2 create-integration \
  --api-id aabbccdee \
  --passthrough-behavior WHEN_NO_MATCH \
  --timeout-in-millis 29000 \
  --connection-type INTERNET \
  --integration-type MOCK

```

Salida:

```

{
  "ConnectionType": "INTERNET",
  "IntegrationId": "0abcdef",
  "IntegrationResponseSelectionExpression": "${integration.response.statuscode}",
  "IntegrationType": "MOCK",
  "PassthroughBehavior": "WHEN_NO_MATCH",
  "PayloadFormatVersion": "1.0",
  "TimeoutInMillis": 29000
}

```

```
}
```

Para obtener más información, consulte [Configurar una solicitud de WebSocket API integración en API Gateway en](#) la Guía para desarrolladores de Amazon API Gateway.

Para crear una HTTP API integración

El siguiente `create-integration` ejemplo crea una integración de AWS Lambda para un HTTP API

```
aws apigatewayv2 create-integration \  
  --api-id a1b2c3d4 \  
  --integration-type AWS_PROXY \  
  --integration-uri arn:aws:lambda:us-west-2:123456789012:function:my-function \  
  --payload-format-version 2.0
```

Salida:

```
{  
  "ConnectionType": "INTERNET",  
  "IntegrationId": "0abcdef",  
  "IntegrationMethod": "POST",  
  "IntegrationType": "AWS_PROXY",  
  "IntegrationUri": "arn:aws:lambda:us-west-2:123456789012:function:my-function",  
  "PayloadFormatVersion": "2.0",  
  "TimeoutInMillis": 30000  
}
```

Para obtener más información, consulte [Configuración de integraciones HTTP APIs en la Guía para](#) desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [CreateIntegration](#) la Referencia de AWS CLI comandos.

## create-route

En el siguiente ejemplo de código, se muestra cómo usar `create-route`.

AWS CLI

Para crear una ruta `$default` para un WebSocket o HTTP API

El siguiente `create-route` ejemplo crea una `$default` ruta para un WebSocket o HTTPAPI.

```
aws apigatewayv2 create-route \  
  --api-id aabbccdee \  
  --route-key '$default'
```

Salida:

```
{  
  "ApiKeyRequired": false,  
  "AuthorizationType": "NONE",  
  "RouteKey": "$default",  
  "RouteId": "1122334"  
}
```

Para obtener más información, consulte [Trabajar con rutas WebSocket APIs en la Guía para desarrolladores de Amazon API Gateway](#).

Para crear una ruta para un HTTP API

En el siguiente `create-route` ejemplo, se crea una ruta denominada `signup` que acepta POST solicitudes.

```
aws apigatewayv2 create-route \  
  --api-id aabbccdee \  
  --route-key 'POST /signup'
```

Salida:

```
{  
  "ApiKeyRequired": false,  
  "AuthorizationType": "NONE",  
  "RouteKey": "POST /signup",  
  "RouteId": "1122334"  
}
```

Para obtener más información, consulte [Trabajar con rutas HTTP APIs en la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [CreateRoute](#) la Referencia de AWS CLI comandos.

## create-stage

En el siguiente ejemplo de código, se muestra cómo usar `create-stage`.

### AWS CLI

Para crear un escenario

El siguiente `create-stage` ejemplo crea un escenario denominado `dev` para un API.

```
aws apigatewayv2 create-stage \  
  --api-id a1b2c3d4 \  
  --stage-name dev
```

Salida:

```
{  
  "CreateDate": "2020-04-06T23:23:46Z",  
  "DefaultRouteSettings": {  
    "DetailedMetricsEnabled": false  
  },  
  "LastUpdatedDate": "2020-04-06T23:23:46Z",  
  "RouteSettings": {},  
  "StageName": "dev",  
  "StageVariables": {},  
  "Tags": {}  
}
```

Para obtener más información, consulte [Trabajar con las etapas de HTTP APIs la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [CreateStage](#) la Referencia de AWS CLI comandos.

## create-vpc-link

En el siguiente ejemplo de código, se muestra cómo usar `create-vpc-link`.

### AWS CLI

Para crear un VPC enlace para un HTTP API

En el siguiente `create-vpc-link` ejemplo se crea un VPC enlace para HTTP APIs.

```
aws apigatewayv2 create-vpc-link \  
  --name MyVpcLink \  
  --subnet-ids subnet-aaaa subnet-bbbb \  
  --security-group-ids sg1234 sg5678
```

Salida:

```
{  
  "CreateDate": "2020-04-07T00:11:46Z",  
  "Name": "MyVpcLink",  
  "SecurityGroupIds": [  
    "sg1234",  
    "sg5678"  
  ],  
  "SubnetIds": [  
    "subnet-aaaa",  
    "subnet-bbbb"  
  ],  
  "Tags": {},  
  "VpcLinkId": "abcd123",  
  "VpcLinkStatus": "PENDING",  
  "VpcLinkStatusMessage": "VPC link is provisioning ENIs",  
  "VpcLinkVersion": "V2"  
}
```

Para obtener más información, consulte [Cómo trabajar con VPC los enlaces de HTTP APIs la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [CreateVpcLink](#) la Referencia de AWS CLI comandos.

## delete-access-log-settings

En el siguiente ejemplo de código, se muestra cómo usar `delete-access-log-settings`.

### AWS CLI

Para deshabilitar el registro de acceso de un API

El siguiente `delete-access-log-settings` ejemplo elimina la configuración del registro de acceso de la `$default` etapa de un API. Para deshabilitar el registro de acceso de una etapa, elimine su configuración del registro de acceso.

```
aws apigatewayv2 delete-access-log-settings \  
  --api-id a1b2c3d4 \  
  --stage-name '$default'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración del registro para un HTTP API](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [DeleteAccessLogSettings](#) la Referencia de AWS CLI comandos.

## delete-api-mapping

En el siguiente ejemplo de código, se muestra cómo usar `delete-api-mapping`.

AWS CLI

Para eliminar un API mapeo

En el siguiente `delete-api-mapping` ejemplo, se elimina una API asignación del nombre de dominio `api.example.com` personalizado.

```
aws apigatewayv2 delete-api-mapping \  
  --api-mapping-id a1b2c3 \  
  --domain-name api.example.com
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de un nombre de dominio personalizado regional en API Gateway en](#) la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [DeleteApiMapping](#) la Referencia de AWS CLI comandos.

## delete-api

En el siguiente ejemplo de código, se muestra cómo usar `delete-api`.

AWS CLI

Para eliminar un API

En el `delete-api` ejemplo siguiente se elimina un API.

```
aws apigatewayv2 delete-api \  
  --api-id a1b2c3d4
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con HTTP APIs](#) y [trabajar con WebSocket APIs](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [DeleteApi](#) la Referencia de AWS CLI comandos.

## **delete-authorizer**

En el siguiente ejemplo de código, se muestra cómo usar `delete-authorizer`.

AWS CLI

Para eliminar un autorizador

En el siguiente `delete-authorizer` ejemplo, se elimina un autorizador.

```
aws apigatewayv2 delete-authorizer \  
  --api-id a1b2c3d4 \  
  --authorizer-id a1b2c3
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Control del acceso HTTP APIs con JWT autorizadores](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [DeleteAuthorizer](#) la Referencia de AWS CLI comandos.

## **delete-cors-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `delete-cors-configuration`.

AWS CLI

Para eliminar la CORS configuración de un HTTP API

En el siguiente `delete-cors-configuration` ejemplo, se deshabilita CORS para un HTTP API mediante la eliminación de su CORS configuración.

```
aws apigatewayv2 delete-cors-configuration \  
  --api-id a1b2c3d4
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de un HTTP API en la Guía CORS para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [DeleteCorsConfiguration](#) la Referencia de AWS CLI comandos.

## **delete-deployment**

En el siguiente ejemplo de código, se muestra cómo usar `delete-deployment`.

AWS CLI

Para eliminar un despliegue

El siguiente `delete-deployment` ejemplo elimina el despliegue de un API.

```
aws apigatewayv2 delete-deployment \  
  --api-id a1b2c3d4 \  
  --deployment-id a1b2c3
```

Este comando no genera ninguna salida.

Para obtener más información, consulte la [API implementación](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [DeleteDeployment](#) la Referencia de AWS CLI comandos.

## **delete-domain-name**

En el siguiente ejemplo de código, se muestra cómo usar `delete-domain-name`.



## AWS CLI

Para eliminar un nombre de dominio personalizado

En el siguiente `delete-domain-name` ejemplo, se elimina un nombre de dominio personalizado.

```
aws apigatewayv2 delete-domain-name \  
  --domain-name api.example.com
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de un nombre de dominio personalizado regional en API Gateway en](#) la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [DeleteDomainName](#) la Referencia de AWS CLI comandos.

## `delete-integration`

En el siguiente ejemplo de código, se muestra cómo usar `delete-integration`.

## AWS CLI

Para eliminar una integración

En el siguiente `delete-integration` ejemplo, se elimina una API integración.

```
aws apigatewayv2 delete-integration \  
  --api-id a1b2c3d4 \  
  --integration-id a1b2c3
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración HTTP APIs y configuración de WebSocket API integraciones en la Guía para](#) desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte la Referencia [DeleteIntegration](#) de AWS CLI comandos.

## `delete-route-settings`

En el siguiente ejemplo de código, se muestra cómo usar `delete-route-settings`.

## AWS CLI

Para eliminar la configuración de la ruta

El siguiente `delete-route-settings` ejemplo elimina la configuración de ruta de la ruta especificada.

```
aws apigatewayv2 delete-route-settings \  
  --api-id a1b2c3d4 \  
  --stage-name dev \  
  --route-key 'GET /pets'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con rutas HTTP APIs en la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [DeleteRouteSettings](#) la Referencia de AWS CLI comandos.

## **delete-route**

En el siguiente ejemplo de código, se muestra cómo usar `delete-route`.

## AWS CLI

Para eliminar una ruta

En el siguiente `delete-route` ejemplo, se elimina una API ruta.

```
aws apigatewayv2 delete-route \  
  --api-id a1b2c3d4 \  
  --route-id a1b2c3
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con rutas HTTP APIs en la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [DeleteRoute](#) la Referencia de AWS CLI comandos.

## delete-stage

En el siguiente ejemplo de código, se muestra cómo usar `delete-stage`.

### AWS CLI

Para eliminar una etapa

En el `delete-stage` ejemplo siguiente se elimina la `test` etapa de un API.

```
aws apigatewayv2 delete-stage \  
  --api-id a1b2c3d4 \  
  --stage-name test
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con las etapas de HTTP APIs la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [DeleteStage](#) la Referencia de AWS CLI comandos.

## delete-vpc-link

En el siguiente ejemplo de código, se muestra cómo usar `delete-vpc-link`.

### AWS CLI

Para eliminar un VPC enlace de un HTTP API

En el siguiente `delete-vpc-link` ejemplo, se elimina un VPC enlace.

```
aws apigatewayv2 delete-vpc-link \  
  --vpc-link-id abcd123
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cómo trabajar con VPC los enlaces de HTTP APIs la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [DeleteVpcLink](#) la Referencia de AWS CLI comandos.

## export-api

En el siguiente ejemplo de código, se muestra cómo usar `export-api`.

### AWS CLI

Para exportar una API definición abierta de un HTTP API

El siguiente `export-api` ejemplo exporta una definición de Open API 3.0 de una API etapa denominada `prod` a un YAML archivo denominado `stage-definition.yaml`. El archivo de definición exportado incluye las extensiones API Gateway de forma predeterminada.

```
aws apigatewayv2 export-api \  
  --api-id a1b2c3d4 \  
  --output-type YAML \  
  --specification OAS30 \  
  --stage-name prod \  
  stage-definition.yaml
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Exportación y HTTP API desde API Gateway](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [ExportApi](#) la Referencia de AWS CLI comandos.

## get-api-mapping

En el siguiente ejemplo de código, se muestra cómo usar `get-api-mapping`.

### AWS CLI

Para obtener información sobre la API asignación de un nombre de dominio personalizado

El siguiente `get-api-mapping` ejemplo muestra información sobre una API asignación para el nombre de dominio `api.example.com` personalizado.

```
aws apigatewayv2 get-api-mapping \  
  --api-mapping-id a1b2c3 \  
  --domain-name api.example.com
```

Salida:

```
{
  "ApiId": "a1b2c3d4",
  "ApiMappingId": "a1b2c3d5",
  "ApiMappingKey": "myTestApi"
  "Stage": "test"
}
```

Para obtener más información, consulte [Configuración de un nombre de dominio personalizado regional en API Gateway en](#) la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [GetApiMapping](#) la Referencia de AWS CLI comandos.

## get-api-mappings

En el siguiente ejemplo de código, se muestra cómo usar `get-api-mappings`.

AWS CLI

Para obtener API las asignaciones de un nombre de dominio personalizado

En el siguiente `get-api-mappings` ejemplo, se muestra una lista de todas las API asignaciones del nombre de dominio personalizado. `api.example.com`

```
aws apigatewayv2 get-api-mappings \
  --domain-name api.example.com
```

Salida:

```
{
  "Items": [
    {
      "ApiId": "a1b2c3d4",
      "ApiMappingId": "a1b2c3d5",
      "ApiMappingKey": "myTestApi"
      "Stage": "test"
    },
    {
      "ApiId": "a5b6c7d8",
```

```

        "ApiMappingId": "a1b2c3d6",
        "ApiMappingKey": "myDevApi"
        "Stage": "dev"
    },
]
}

```

Para obtener más información, consulte [Configuración de un nombre de dominio personalizado regional en API Gateway en la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [GetApiMappings](#) la Referencia de AWS CLI comandos.

## get-api

En el siguiente ejemplo de código, se muestra cómo usar `get-api`.

### AWS CLI

Para recuperar información sobre un API

El siguiente `get-api` ejemplo muestra información sobre un API.

```

aws apigatewayv2 get-api \
  --api-id a1b2c3d4

```

Salida:

```

{
  "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
  "ApiId": "a1b2c3d4",
  "ApiKeySelectionExpression": "$request.header.x-api-key",
  "CreateDate": "2020-03-28T00:32:37Z",
  "Name": "my-api",
  "ProtocolType": "HTTP",
  "RouteSelectionExpression": "$request.method $request.path",
  "Tags": {
    "department": "finance"
  }
}

```

- Para API obtener más información, consulte [GetApi](#) la Referencia de AWS CLI comandos.

## get-apis

En el siguiente ejemplo de código, se muestra cómo usar `get-apis`.

### AWS CLI

Para recuperar una lista de APIs

En el siguiente `get-apis` ejemplo, se enumeran todos APIs los del usuario actual.

```
aws apigatewayv2 get-apis
```

Salida:

```
{
  "Items": [
    {
      "ApiEndpoint": "wss://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
      "ApiId": "a1b2c3d4",
      "ApiKeySelectionExpression": "$request.header.x-api-key",
      "CreateDate": "2020-04-07T20:21:59Z",
      "Name": "my-websocket-api",
      "ProtocolType": "WEBSOCKET",
      "RouteSelectionExpression": "$request.body.message",
      "Tags": {}
    },
    {
      "ApiEndpoint": "https://a1b2c3d5.execute-api.us-west-2.amazonaws.com",
      "ApiId": "a1b2c3d5",
      "ApiKeySelectionExpression": "$request.header.x-api-key",
      "CreateDate": "2020-04-07T20:23:50Z",
      "Name": "my-http-api",
      "ProtocolType": "HTTP",
      "RouteSelectionExpression": "$request.method $request.path",
      "Tags": {}
    }
  ]
}
```

Para obtener más información, consulte [Trabajar con HTTP APIs](#) y [trabajar con WebSocket APIs](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [GetApis](#) la Referencia de AWS CLI comandos.

## get-authorizer

En el siguiente ejemplo de código, se muestra cómo usar `get-authorizer`.

### AWS CLI

Para recuperar información sobre un autorizador

En el siguiente `get-authorizer` ejemplo, se muestra información sobre un autorizador.

```
aws apigatewayv2 get-authorizer \  
  --api-id a1b2c3d4 \  
  --authorizer-id a1b2c3
```

Salida:

```
{  
  "AuthorizerId": "a1b2c3",  
  "AuthorizerType": "JWT",  
  "IdentitySource": [  
    "$request.header.Authorization"  
  ],  
  "JwtConfiguration": {  
    "Audience": [  
      "123456abc"  
    ],  
    "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-west-2_abc123"  
  },  
  "Name": "my-jwt-authorizer"  
}
```

Para obtener más información, consulte [Control del acceso HTTP APIs con JWT autorizadores](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [GetAuthorizer](#) la Referencia de AWS CLI comandos.

## get-authorizers

En el siguiente ejemplo de código, se muestra cómo usar `get-authorizers`.



## AWS CLI

Para recuperar una lista de autorizadores de un API

El siguiente `get-authorizers` ejemplo muestra una lista de todos los autorizadores de un API

```
aws apigatewayv2 get-authorizers \  
  --api-id a1b2c3d4
```

Salida:

```
{  
  "Items": [  
    {  
      "AuthorizerId": "a1b2c3",  
      "AuthorizerType": "JWT",  
      "IdentitySource": [  
        "$request.header.Authorization"  
      ],  
      "JwtConfiguration": {  
        "Audience": [  
          "123456abc"  
        ],  
        "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-  
west-2_abc123"  
      },  
      "Name": "my-jwt-authorizer"  
    },  
    {  
      "AuthorizerId": "a1b2c4",  
      "AuthorizerType": "JWT",  
      "IdentitySource": [  
        "$request.header.Authorization"  
      ],  
      "JwtConfiguration": {  
        "Audience": [  
          "6789abcde"  
        ],  
        "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-  
west-2_abc234"  
      },  
      "Name": "new-jwt-authorizer"  
    }  
  ]  
}
```

```
]
}
```

Para obtener más información, consulte [Control del acceso HTTP APIs con JWT autorizadores](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [GetAuthorizers](#) la Referencia de AWS CLI comandos.

## get-deployment

En el siguiente ejemplo de código, se muestra cómo usar `get-deployment`.

### AWS CLI

Para recuperar información sobre una implementación

En el siguiente `get-deployment` ejemplo, se muestra información sobre una implementación.

```
aws apigatewayv2 get-deployment \
  --api-id a1b2c3d4 \
  --deployment-id abcdef
```

Salida:

```
{
  "AutoDeployed": true,
  "CreateDate": "2020-04-07T23:58:40Z",
  "DeploymentId": "abcdef",
  "DeploymentStatus": "DEPLOYED",
  "Description": "Automatic deployment triggered by changes to the Api
configuration"
}
```

Para obtener más información, consulte la [API implementación](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [GetDeployment](#) la Referencia de AWS CLI comandos.

## get-deployments

En el siguiente ejemplo de código, se muestra cómo usar `get-deployments`.

### AWS CLI

Para recuperar una lista de despliegues

En el siguiente `get-deployments` ejemplo, se muestra una lista de todas las implementaciones API de un usuario.

```
aws apigatewayv2 get-deployments \  
  --api-id a1b2c3d4
```

Salida:

```
{  
  "Items": [  
    {  
      "AutoDeployed": true,  
      "CreateDate": "2020-04-07T23:58:40Z",  
      "DeploymentId": "abcdef",  
      "DeploymentStatus": "DEPLOYED",  
      "Description": "Automatic deployment triggered by changes to the Api  
configuration"  
    },  
    {  
      "AutoDeployed": true,  
      "CreateDate": "2020-04-06T00:33:00Z",  
      "DeploymentId": "bcdefg",  
      "DeploymentStatus": "DEPLOYED",  
      "Description": "Automatic deployment triggered by changes to the Api  
configuration"  
    }  
  ]  
}
```

Para obtener más información, consulte la [APIimplementación](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [GetDeployments](#) la Referencia de AWS CLI comandos.

## get-domain-name

En el siguiente ejemplo de código, se muestra cómo usar `get-domain-name`.

### AWS CLI

Para recuperar información sobre un nombre de dominio personalizado

En el siguiente `get-domain-name` ejemplo, se muestra información sobre un nombre de dominio personalizado.

```
aws apigatewayv2 get-domain-name \  
  --domain-name api.example.com
```

Salida:

```
{  
  "ApiMappingSelectionExpression": "$request.basepath",  
  "DomainName": "api.example.com",  
  "DomainNameConfigurations": [  
    {  
      "ApiGatewayDomainName": "d-1234.execute-api.us-west-2.amazonaws.com",  
      "CertificateArn": "arn:aws:acm:us-  
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",  
      "EndpointType": "REGIONAL",  
      "HostedZoneId": "123456789111",  
      "SecurityPolicy": "TLS_1_2",  
      "DomainNameStatus": "AVAILABLE"  
    }  
  ],  
  "Tags": {}  
}
```

Para obtener más información, consulte [Configuración de un nombre de dominio personalizado regional en API Gateway en](#) la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [GetDomainName](#) la Referencia de AWS CLI comandos.

## get-domain-names

En el siguiente ejemplo de código, se muestra cómo usar `get-domain-names`.

## AWS CLI

Para recuperar una lista de nombres de dominio personalizados

En el siguiente `get-domain-names` ejemplo, se muestra una lista de todos los nombres de dominio personalizados del usuario actual.

```
aws apigatewayv2 get-domain-names
```

Salida:

```
{
  "Items": [
    {
      "ApiMappingSelectionExpression": "$request.basepath",
      "DomainName": "api.example.com",
      "DomainNameConfigurations": [
        {
          "ApiGatewayDomainName": "d-1234.execute-api.us-
west-2.amazonaws.com",
          "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
          "EndpointType": "REGIONAL",
          "HostedZoneId": "123456789111",
          "SecurityPolicy": "TLS_1_2",
          "DomainNameStatus": "AVAILABLE"
        }
      ]
    },
    {
      "ApiMappingSelectionExpression": "$request.basepath",
      "DomainName": "newApi.example.com",
      "DomainNameConfigurations": [
        {
          "ApiGatewayDomainName": "d-5678.execute-api.us-
west-2.amazonaws.com",
          "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
          "EndpointType": "REGIONAL",
          "HostedZoneId": "123456789222",
          "SecurityPolicy": "TLS_1_2",
          "DomainNameStatus": "AVAILABLE"
        }
      ]
    }
  ]
}
```

```

    ]
  }
]
}

```

Para obtener más información, consulte [Configuración de un nombre de dominio personalizado regional en API Gateway en](#) la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [GetDomainNames](#) la Referencia de AWS CLI comandos.

## get-integration

En el siguiente ejemplo de código, se muestra cómo usar `get-integration`.

AWS CLI

Para recuperar información sobre una integración

El siguiente `get-integration` ejemplo muestra información sobre una integración.

```

aws apigatewayv2 get-integration \
  --api-id a1b2c3d4 \
  --integration-id a1b2c3

```

Salida:

```

{
  "ApiGatewayManaged": true,
  "ConnectionType": "INTERNET",
  "IntegrationId": "a1b2c3",
  "IntegrationMethod": "POST",
  "IntegrationType": "AWS_PROXY",
  "IntegrationUri": "arn:aws:lambda:us-west-2:12356789012:function:hello12",
  "PayloadFormatVersion": "2.0",
  "TimeoutInMillis": 30000
}

```

Para obtener más información, consulte [Configuración HTTP APIs y configuración de WebSocket API integraciones en la Guía para](#) desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte la Referencia [GetIntegration](#) de AWS CLI comandos.

## get-integrations

En el siguiente ejemplo de código, se muestra cómo usar `get-integrations`.

### AWS CLI

Para recuperar una lista de integraciones

En el siguiente `get-integrations` ejemplo, se muestra una lista de todas las integraciones API de una persona.

```
aws apigatewayv2 get-integrations \  
  --api-id a1b2c3d4
```

Salida:

```
{  
  "Items": [  
    {  
      "ApiGatewayManaged": true,  
      "ConnectionType": "INTERNET",  
      "IntegrationId": "a1b2c3",  
      "IntegrationMethod": "POST",  
      "IntegrationType": "AWS_PROXY",  
      "IntegrationUri": "arn:aws:lambda:us-west-2:123456789012:function:my-  
function",  
      "PayloadFormatVersion": "2.0",  
      "TimeoutInMillis": 30000  
    },  
    {  
      "ConnectionType": "INTERNET",  
      "IntegrationId": "a1b2c4",  
      "IntegrationMethod": "ANY",  
      "IntegrationType": "HTTP_PROXY",  
      "IntegrationUri": "https://www.example.com",  
      "PayloadFormatVersion": "1.0",  
      "TimeoutInMillis": 30000  
    }  
  ]  
}
```

```
}
```

Para obtener más información, consulte [Configuración HTTP APIs y configuración de WebSocket API integraciones en la Guía para](#) desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte la Referencia [GetIntegrations](#) de AWS CLI comandos.

## get-route

En el siguiente ejemplo de código, se muestra cómo usar `get-route`.

### AWS CLI

Para recuperar información sobre una ruta

El siguiente `get-route` ejemplo muestra información sobre una ruta.

```
aws apigatewayv2 get-route \  
  --api-id a1b2c3d4 \  
  --route-id 72jz1wk
```

Salida:

```
{  
  "ApiKeyRequired": false,  
  "AuthorizationType": "NONE",  
  "RouteId": "72jz1wk",  
  "RouteKey": "ANY /pets",  
  "Target": "integrations/a1b2c3"  
}
```

Para obtener más información, consulte [Trabajar con rutas HTTP APIs en la Guía para](#) desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [GetRoute](#) la Referencia de AWS CLI comandos.

## get-routes

En el siguiente ejemplo de código, se muestra cómo usar `get-routes`.



## AWS CLI

Para recuperar una lista de rutas

En el siguiente `get-routes` ejemplo, se muestra una lista de todas las rutas API de un usuario.

```
aws apigatewayv2 get-routes \  
  --api-id a1b2c3d4
```

Salida:

```
{  
  "Items": [  
    {  
      "ApiKeyRequired": false,  
      "AuthorizationType": "NONE",  
      "RouteId": "72jz1wk",  
      "RouteKey": "ANY /admin",  
      "Target": "integrations/a1b2c3"  
    },  
    {  
      "ApiGatewayManaged": true,  
      "ApiKeyRequired": false,  
      "AuthorizationType": "NONE",  
      "RouteId": "go65gqi",  
      "RouteKey": "$default",  
      "Target": "integrations/a1b2c4"  
    }  
  ]  
}
```

Para obtener más información, consulte [Trabajar con rutas HTTP APIs en la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [GetRoutes](#) la Referencia de AWS CLI comandos.

## get-stage

En el siguiente ejemplo de código, se muestra cómo usar `get-stage`.

## AWS CLI

Para recuperar información sobre una etapa

En el siguiente `get-stage` ejemplo se muestra información sobre la `prod` etapa de una API.

```
aws apigatewayv2 get-stage \  
  --api-id a1b2c3d4 \  
  --stage-name prod
```

Salida:

```
{  
  "CreateDate": "2020-04-08T00:36:05Z",  
  "DefaultRouteSettings": {  
    "DetailedMetricsEnabled": false  
  },  
  "DeploymentId": "x1zwyv",  
  "LastUpdatedDate": "2020-04-08T00:36:13Z",  
  "RouteSettings": {},  
  "StageName": "prod",  
  "StageVariables": {  
    "function": "my-prod-function"  
  },  
  "Tags": {}  
}
```

Para obtener más información, consulte [Trabajar con las etapas de HTTP APIs la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [GetStage](#) la Referencia de AWS CLI comandos.

## get-stages

En el siguiente ejemplo de código, se muestra cómo usar `get-stages`.

AWS CLI

Para recuperar una lista de etapas

En el siguiente `get-stages` ejemplo se enumeran todas las etapas API de una.

```
aws apigatewayv2 get-stages \  
  --api-id a1b2c3d4
```

Salida:

```
{
  "Items": [
    {
      "ApiGatewayManaged": true,
      "AutoDeploy": true,
      "CreateDate": "2020-04-08T00:08:44Z",
      "DefaultRouteSettings": {
        "DetailedMetricsEnabled": false
      },
      "DeploymentId": "dty748",
      "LastDeploymentStatusMessage": "Successfully deployed stage with
deployment ID 'dty748'",
      "LastUpdatedDate": "2020-04-08T00:09:49Z",
      "RouteSettings": {},
      "StageName": "$default",
      "StageVariables": {},
      "Tags": {}
    },
    {
      "AutoDeploy": true,
      "CreateDate": "2020-04-08T00:35:06Z",
      "DefaultRouteSettings": {
        "DetailedMetricsEnabled": false
      },
      "LastUpdatedDate": "2020-04-08T00:35:48Z",
      "RouteSettings": {},
      "StageName": "dev",
      "StageVariables": {
        "function": "my-dev-function"
      },
      "Tags": {}
    },
    {
      "CreateDate": "2020-04-08T00:36:05Z",
      "DefaultRouteSettings": {
        "DetailedMetricsEnabled": false
      },
      "DeploymentId": "x1zwyv",
      "LastUpdatedDate": "2020-04-08T00:36:13Z",
      "RouteSettings": {},
      "StageName": "prod",
      "StageVariables": {
        "function": "my-prod-function"
      }
    }
  ]
}
```

```
    },  
    "Tags": {}  
  }  
]  
}
```

Para obtener más información, consulte [Trabajar con las etapas de HTTP APIs la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [GetStages](#) la Referencia de AWS CLI comandos.

## get-tags

En el siguiente ejemplo de código, se muestra cómo usar `get-tags`.

### AWS CLI

Para recuperar una lista de etiquetas de un recurso

En el siguiente `get-tags` ejemplo, se enumeran todas las etiquetas API de un usuario.

```
aws apigatewayv2 get-tags \  
  --resource-arn arn:aws:apigateway:us-west-2:/apis/a1b2c3d4
```

Salida:

```
{  
  "Tags": {  
    "owner": "dev-team",  
    "environment": "prod"  
  }  
}
```

Para obtener más información, consulte [Etiquetar los recursos de API Gateway](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [GetTags](#) la Referencia de AWS CLI comandos.

## get-vpc-link

En el siguiente ejemplo de código, se muestra cómo usar `get-vpc-link`.

## AWS CLI

Para recuperar información sobre un VPC enlace

En el siguiente `get-vpc-link` ejemplo, se muestra información acerca de un VPC enlace.

```
aws apigatewayv2 get-vpc-link \  
  --vpc-link-id abcd123
```

Salida:

```
{  
  "CreateDate": "2020-04-07T00:27:47Z",  
  "Name": "MyVpcLink",  
  "SecurityGroupIds": [  
    "sg1234",  
    "sg5678"  
  ],  
  "SubnetIds": [  
    "subnet-aaaa",  
    "subnet-bbbb"  
  ],  
  "Tags": {},  
  "VpcLinkId": "abcd123",  
  "VpcLinkStatus": "AVAILABLE",  
  "VpcLinkStatusMessage": "VPC link is ready to route traffic",  
  "VpcLinkVersion": "V2"  
}
```

Para obtener más información, consulte [Cómo trabajar con VPC los enlaces de HTTP APIs la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [GetVpcLink](#) la Referencia de AWS CLI comandos.

## get-vpc-links

En el siguiente ejemplo de código, se muestra cómo usar `get-vpc-links`.

## AWS CLI

Para recuperar una lista de VPC enlaces

En el siguiente `get-vpc-links` ejemplo, se muestra una lista de todos los VPC enlaces del usuario actual.

```
aws apigatewayv2 get-vpc-links
```

Salida:

```
{
  "Items": [
    {
      "CreateDate": "2020-04-07T00:27:47Z",
      "Name": "MyVpcLink",
      "SecurityGroupIds": [
        "sg1234",
        "sg5678"
      ],
      "SubnetIds": [
        "subnet-aaaa",
        "subnet-bbbb"
      ],
      "Tags": {},
      "VpcLinkId": "abcd123",
      "VpcLinkStatus": "AVAILABLE",
      "VpcLinkStatusMessage": "VPC link is ready to route traffic",
      "VpcLinkVersion": "V2"
    }
  ],
  {
    "CreateDate": "2020-04-07T00:27:47Z",
    "Name": "MyOtherVpcLink",
    "SecurityGroupIds": [
      "sg1234",
      "sg5678"
    ],
    "SubnetIds": [
      "subnet-aaaa",
      "subnet-bbbb"
    ],
    "Tags": {},
    "VpcLinkId": "abcd456",
    "VpcLinkStatus": "AVAILABLE",
    "VpcLinkStatusMessage": "VPC link is ready to route traffic",
    "VpcLinkVersion": "V2"
  }
}
```

```
]
}
```

Para obtener más información, consulte [Cómo trabajar con VPC los enlaces de HTTP APIs la Guía para](#) desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [GetVpcLinks](#) la Referencia de AWS CLI comandos.

## import-api

En el siguiente ejemplo de código, se muestra cómo usar `import-api`.

### AWS CLI

Para importar un HTTP API

En el siguiente `import-api` ejemplo, se crea una HTTP API a partir de un archivo de definición de Open API 3.0 denominado `api-definition.yaml`.

```
aws apigatewayv2 import-api \  
  --body file://api-definition.yaml
```

Contenidos de `api-definition.yaml`:

```
openapi: 3.0.1  
info:  
  title: My Lambda API  
  version: v1.0  
paths:  
  /hello:  
    x-amazon-apigateway-any-method:  
      x-amazon-apigateway-integration:  
        payloadFormatVersion: 2.0  
        type: aws_proxy  
        httpMethod: POST  
        uri: arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/  
arn:aws:lambda:us-west-2:123456789012:function:hello/invocations  
        connectionType: INTERNET
```

Salida:

```
{
```

```

"ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
"ApiId": "a1b2c3d4",
"ApiKeySelectionExpression": "$request.header.x-api-key",
"CreateDate": "2020-04-08T17:19:38+00:00",
"Name": "My Lambda API",
"ProtocolType": "HTTP",
"RouteSelectionExpression": "$request.method $request.path",
"Tags": {},
"Version": "v1.0"
}

```

Para obtener más información, consulte [Trabajar con API definiciones abiertas HTTP APIs en la Guía para](#) desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [ImportApi](#) Referencia de AWS CLI comandos.

## reimport-api

En el siguiente ejemplo de código, se muestra cómo usar `reimport-api`.

### AWS CLI

Para volver a importar un HTTP API

El siguiente `reimport-api` ejemplo actualiza uno existente HTTP API para usar la definición de Open API 3.0 especificada en `api-definition.yaml`.

```

aws apigatewayv2 reimport-api \
  --body file://api-definition.yaml \
  --api-id a1b2c3d4

```

Contenidos de `api-definition.yaml`:

```

openapi: 3.0.1
info:
  title: My Lambda API
  version: v1.0
paths:
  /hello:
    x-amazon-apigateway-any-method:
      x-amazon-apigateway-integration:
        payloadFormatVersion: 2.0

```



```

    type: aws_proxy
    httpMethod: POST
    uri: arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:12356789012:function:hello/invocations
    connectionType: INTERNET

```

Salida:

```

{
  "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
  "ApiId": "a1b2c3d4",
  "ApiKeySelectionExpression": "$request.header.x-api-key",
  "CreateDate": "2020-04-08T17:19:38+00:00",
  "Name": "My Lambda API",
  "ProtocolType": "HTTP",
  "RouteSelectionExpression": "$request.method $request.path",
  "Tags": {},
  "Version": "v1.0"
}

```

Para obtener más información, consulte [Trabajar con API definiciones abiertas HTTP APIs en la Guía para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [ReimportApi](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

### AWS CLI

Para etiquetar un recurso

En el siguiente tag-resource ejemplo, se agrega una etiqueta con el nombre de la clave Department y un valor de Accounting al especificadoAPI.

```

aws apigatewayv2 tag-resource \
  --resource-arn arn:aws:apigateway:us-west-2:/apis/a1b2c3d4 \
  --tags Department=Accounting

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar los recursos de API Gateway](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Cómo eliminar etiquetas de un recurso

En el siguiente `untag-resource` ejemplo, se eliminan las etiquetas con los nombres de las claves `Project` y `Owner` de las especificadas API.

```
aws apigatewayv2 untag-resource \  
  --resource-arn arn:aws:apigateway:us-west-2::/apis/a1b2c3d4 \  
  --tag-keys Project Owner
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar los recursos de API Gateway](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-api-mapping

En el siguiente ejemplo de código, se muestra cómo usar `update-api-mapping`.

### AWS CLI

Para actualizar un API mapeo

El siguiente `update-api-mapping` ejemplo cambia una API asignación de un nombre de dominio personalizado. Como resultado, la base que URL utiliza el nombre de dominio personalizado para la etapa API y especificada pasa a ser `https://api.example.com/dev`.

```
aws apigatewayv2 update-api-mapping \  
  --resource-arn arn:aws:apigateway:us-west-2::/apis/a1b2c3d4 \  
  --stage-name dev \  
  --target-url https://api.example.com/dev
```

```
--api-id a1b2c3d4 \  
--stage dev \  
--domain-name api.example.com \  
--api-mapping-id 0qzs2sy7bh \  
--api-mapping-key dev
```

Salida:

```
{  
  "ApiId": "a1b2c3d4",  
  "ApiMappingId": "0qzs2sy7bh",  
  "ApiMappingKey": "dev"  
  "Stage": "dev"  
}
```

Para obtener más información, consulte [Configuración de un nombre de dominio personalizado regional en API Gateway en](#) la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [UpdateApiMapping](#) la Referencia de AWS CLI comandos.

## update-api

En el siguiente ejemplo de código, se muestra cómo usar `update-api`.

AWS CLI

CORS Para habilitar un HTTP API

El siguiente `update-api` ejemplo actualiza la CORS configuración API de la especificación para permitir las solicitudes de `https://www.example.com`.

```
aws apigatewayv2 update-api \  
  --api-id a1b2c3d4 \  
  --cors-configuration AllowOrigins=https://www.example.com
```

Salida:

```
{  
  "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",  
  "ApiId": "a1b2c3d4",  
  "ApiKeySelectionExpression": "$request.header.x-api-key",
```

```
"CorsConfiguration": {
  "AllowCredentials": false,
  "AllowHeaders": [
    "header1",
    "header2"
  ],
  "AllowMethods": [
    "GET",
    "OPTIONS"
  ],
  "AllowOrigins": [
    "https://www.example.com"
  ]
},
"CreateDate": "2020-04-08T18:39:37+00:00",
"Name": "my-http-api",
"ProtocolType": "HTTP",
"RouteSelectionExpression": "$request.method $request.path",
"Tags": {},
"Version": "v1.0"
}
```

Para obtener más información, consulte [Configuración de un HTTP API en la Guía CORS para desarrolladores de Amazon API Gateway](#).

- Para API obtener más información, consulte [UpdateApi](#) la Referencia de AWS CLI comandos.

## update-authorizer

En el siguiente ejemplo de código, se muestra cómo usar update-authorizer.

### AWS CLI

Para actualizar un autorizador

En el siguiente update-authorizer ejemplo, se cambia la fuente de identidad de un JWT autorizador por un encabezado denominado Authorization

```
aws apigatewayv2 update-authorizer \
  --api-id a1b2c3d4 \
  --authorizer-id a1b2c3 \
  --identity-source '$request.header.Authorization'
```

Salida:

```
{
  "AuthorizerId": "a1b2c3",
  "AuthorizerType": "JWT",
  "IdentitySource": [
    "$request.header.Authorization"
  ],
  "JwtConfiguration": {
    "Audience": [
      "123456abc"
    ],
    "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-west-2_abc123"
  },
  "Name": "my-jwt-authorizer"
}
```

Para obtener más información, consulte [Control del acceso HTTP APIs con JWT autorizadores](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [UpdateAuthorizer](#) la Referencia de AWS CLI comandos.

## update-deployment

En el siguiente ejemplo de código, se muestra cómo usar update-deployment.

AWS CLI

Para cambiar la descripción de una implementación

El siguiente update-deployment ejemplo actualiza la descripción de una implementación.

```
aws apigatewayv2 update-deployment \
  --api-id a1b2c3d4 \
  --deployment-id abcdef \
  --description 'Manual deployment to fix integration test failures.'
```

Salida:

```
{
  "AutoDeployed": false,
```

```

    "CreateDate": "2020-02-05T16:21:48+00:00",
    "DeploymentId": "abcdef",
    "DeploymentStatus": "DEPLOYED",
    "Description": "Manual deployment to fix integration test failures."
  }

```

Para obtener más información, consulte [Desarrollo de un HTTP API in API Gateway](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [UpdateDeployment](#) la Referencia de AWS CLI comandos.

## update-domain-name

En el siguiente ejemplo de código, se muestra cómo usar `update-domain-name`.

### AWS CLI

Para actualizar un nombre de dominio personalizado

En el siguiente `update-domain-name` ejemplo, se especifica un ACM certificado nuevo para el nombre de dominio `api.example.com` personalizado.

```

aws apigatewayv2 update-domain-name \
  --domain-name api.example.com \
  --domain-name-configurations CertificateArn=arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678

```

Salida:

```

{
  "ApiMappingSelectionExpression": "$request.basepath",
  "DomainName": "regional.example.com",
  "DomainNameConfigurations": [
    {
      "ApiGatewayDomainName": "d-id.execute-api.us-west-2.amazonaws.com",
      "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
      "EndpointType": "REGIONAL",
      "HostedZoneId": "123456789111",
      "SecurityPolicy": "TLS_1_2",
      "DomainNameStatus": "AVAILABLE"
    }
  ]
}

```

```
]
}
```

Para obtener más información, consulte [Configuración de un nombre de dominio personalizado regional en API Gateway en](#) la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [UpdateDomainName](#) la Referencia de AWS CLI comandos.

## update-integration

En el siguiente ejemplo de código, se muestra cómo usar `update-integration`.

### AWS CLI

Para actualizar una integración de Lambda

El siguiente `update-integration` ejemplo actualiza una integración AWS Lambda existente para usar la función Lambda especificada.

```
aws apigatewayv2 update-integration \
  --api-id a1b2c3d4 \
  --integration-id a1b2c3 \
  --integration-uri arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/  
arn:aws:lambda:us-west-2:123456789012:function:my-new-function/invocations
```

Salida:

```
{
  "ConnectionType": "INTERNET",
  "IntegrationId": "a1b2c3",
  "IntegrationMethod": "POST",
  "IntegrationType": "AWS_PROXY",
  "IntegrationUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/
functions/arn:aws:lambda:us-west-2:123456789012:function:my-new-function/
invocations",
  "PayloadFormatVersion": "2.0",
  "TimeoutInMillis": 5000
}
```

Para obtener más información, consulte [Configuración HTTP APIs y configuración de WebSocket API integraciones en la Guía para](#) desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte la Referencia [UpdateIntegration](#) de AWS CLI comandos.

## update-route

En el siguiente ejemplo de código, se muestra cómo usar `update-route`.

### AWS CLI

Ejemplo 1: Para actualizar la integración de una ruta

El siguiente `update-route` ejemplo actualiza la integración de una ruta específica.

```
aws apigatewayv2 update-route \  
  --api-id a1b2c3d4 \  
  --route-id a1b2c3 \  
  --target integrations/a1b2c6
```

Salida:

```
{  
  "ApiKeyRequired": false,  
  "AuthorizationType": "NONE",  
  "RouteId": "a1b2c3",  
  "RouteKey": "ANY /pets",  
  "Target": "integrations/a1b2c6"  
}
```

Ejemplo 2: Para añadir un autorizador a una ruta

El siguiente `update-route` ejemplo actualiza la ruta especificada para usar un JWT autorizador.

```
aws apigatewayv2 update-route \  
  --api-id a1b2c3d4 \  
  --route-id a1b2c3 \  
  --authorization-type JWT \  
  --authorizer-id a1b2c5 \  
  --authorization-scopes user.id user.email
```

Salida:



```
{
  "ApiKeyRequired": false,
  "AuthorizationScopes": [
    "user.id",
    "user.email"
  ],
  "AuthorizationType": "JWT",
  "AuthorizerId": "a1b2c5",
  "OperationName": "GET HTTP",
  "RequestParameters": {},
  "RouteId": "a1b2c3",
  "RouteKey": "GET /pets",
  "Target": "integrations/a1b2c6"
}
```

Para obtener más información, consulte [Control del acceso HTTP APIs con JWT autorizadores](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [UpdateRoute](#) la Referencia de AWS CLI comandos.

## update-stage

En el siguiente ejemplo de código, se muestra cómo usar `update-stage`.

### AWS CLI

Para configurar una regulación personalizada

El siguiente `update-stage` ejemplo configura la regulación personalizada para la etapa y la ruta especificadas de un API

```
aws apigatewayv2 update-stage \
  --api-id a1b2c3d4 \
  --stage-name dev \
  --route-settings '{"GET /pets":  
{"ThrottlingBurstLimit":100,"ThrottlingRateLimit":2000}}'
```

Salida:

```
{
  "CreateDate": "2020-04-05T16:21:16+00:00",
```

```

"DefaultRouteSettings": {
  "DetailedMetricsEnabled": false
},
"DeploymentId": "shktxb",
"LastUpdatedDate": "2020-04-08T22:23:17+00:00",
"RouteSettings": {
  "GET /pets": {
    "ThrottlingBurstLimit": 100,
    "ThrottlingRateLimit": 2000.0
  }
},
"StageName": "dev",
"StageVariables": {},
"Tags": {}
}

```

Para obtener más información, consulte Cómo [protegerse HTTP API](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [UpdateStage](#) la Referencia de AWS CLI comandos.

## update-vpc-link

En el siguiente ejemplo de código, se muestra cómo usar `update-vpc-link`.

### AWS CLI

Para actualizar un VPC enlace

En el siguiente `update-vpc-link` ejemplo, se actualiza el nombre de un VPC enlace. Una vez creado un VPC enlace, no puede cambiar sus grupos de seguridad ni sus subredes.

```

aws apigatewayv2 update-vpc-link \
  --vpc-link-id abcd123 \
  --name MyUpdatedVpcLink

```

Salida:

```

{
  "CreateDate": "2020-04-07T00:27:47Z",
  "Name": "MyUpdatedVpcLink",
  "SecurityGroupIds": [
    "sg1234",

```

```
    "sg5678"  
  ],  
  "SubnetIds": [  
    "subnet-aaaa",  
    "subnet-bbbb"  
  ],  
  "Tags": {},  
  "VpcLinkId": "abcd123",  
  "VpcLinkStatus": "AVAILABLE",  
  "VpcLinkStatusMessage": "VPC link is ready to route traffic",  
  "VpcLinkVersion": "V2"  
}
```

Para obtener más información, consulte Cómo [trabajar con VPC los enlaces de HTTP APIs la Guía para](#) desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [UpdateVpcLink](#) la Referencia de AWS CLI comandos.

## API API Ejemplos de administración de puertas de enlace utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante el AWS Command Line Interface uso de API Gateway Management API.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

#### **delete-connection**

En el siguiente ejemplo de código, se muestra cómo usar `delete-connection`.

## AWS CLI

Para eliminar una WebSocket conexión

El siguiente `delete-connection` ejemplo desconecta un cliente del especificado WebSocket API.

```
aws apigatewaymanagementapi delete-connection \  
  --connection-id L0SM9c0FvHcCIhw= \  
  --endpoint-url https://aabbccdde.execute-api.us-west-2.amazonaws.com/prod
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Utilizar los comandos @connections en su servicio de backend](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [DeleteConnection](#) la Referencia de AWS CLI comandos.

## get-connection

En el siguiente ejemplo de código, se muestra cómo usar `get-connection`.

## AWS CLI

Para obtener información sobre una WebSocket conexión

El siguiente `get-connection` ejemplo describe una conexión a la especificada WebSocket API.

```
aws apigatewaymanagementapi get-connection \  
  --connection-id L0SM9c0FvHcCIhw= \  
  --endpoint-url https://aabbccdde.execute-api.us-west-2.amazonaws.com/prod
```

Salida:

```
{  
  "ConnectedAt": "2020-04-30T20:10:33.236Z",  
  "Identity": {  
    "SourceIp": "192.0.2.1"  
  },  
  "LastActiveAt": "2020-04-30T20:10:42.997Z"
```

```
}
```

Para obtener más información, consulte [Utilizar los comandos @connections en su servicio de backend](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [GetConnection](#) la Referencia de AWS CLI comandos.

## post-to-connection

En el siguiente ejemplo de código, se muestra cómo usar `post-to-connection`.

### AWS CLI

Para enviar datos a una WebSocket conexión

El siguiente `post-to-connection` ejemplo envía un mensaje a un cliente que está conectado al especificado WebSocket API.

```
aws apigatewaymanagementapi post-to-connection \  
  --connection-id L0SM9c0FvHcCIhw= \  
  --data "Hello from API Gateway!" \  
  --endpoint-url https://aabbccdee.execute-api.us-west-2.amazonaws.com/prod
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Utilizar los comandos @connections en su servicio de backend](#) en la Guía para desarrolladores de Amazon API Gateway.

- Para API obtener más información, consulte [PostToConnection](#) la Referencia de AWS CLI comandos.

## Ejemplos de App Mesh usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante App Mesh.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **create-mesh**

En el siguiente ejemplo de código, se muestra cómo usar `create-mesh`.

#### AWS CLI

Ejemplo 1: Para crear una nueva malla de servicios

El siguiente `create-mesh` ejemplo crea una malla de servicios.

```
aws appmesh create-mesh \  
  --mesh-name app1
```

Salida:

```
{  
  "mesh":{  
    "meshName":"app1",  
    "metadata":{  
      "arn":"arn:aws:appmesh:us-east-1:123456789012:mesh/app1",  
      "createdAt":1563809909.282,  
      "lastUpdatedAt":1563809909.282,  
      "uid":"a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version":1  
    },  
    "spec":{ },  
    "status":{  
      "status":"ACTIVE"  
    }  
  }  
}
```

Ejemplo 2: Para crear una nueva malla de servicios con varias etiquetas

El siguiente `create-mesh` ejemplo crea una malla de servicios con varias etiquetas.

```
aws appmesh create-mesh \
  --mesh-name app2 \
  --tags key=key1,value=value1 key=key2,value=value2 key=key3,value=value3
```

Salida:

```
{
  "mesh":{
    "meshName":"app2",
    "metadata":{
      "arn":"arn:aws:appmesh:us-east-1:123456789012:mesh/app2",
      "createdAt":1563822121.877,
      "lastUpdatedAt":1563822121.877,
      "uid":"a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version":1
    },
    "spec":{},
    "status":{
      "status":"ACTIVE"
    }
  }
}
```

Para obtener más información, consulte [Service Mesh en la](#) Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [CreateMesh](#) la Referencia de AWS CLI comandos.

## **create-route**

En el siguiente ejemplo de código, se muestra cómo usar `create-route`.

### AWS CLI

Para crear una nueva RPC ruta g

En el siguiente `create-route` ejemplo, se utiliza un archivo de JSON entrada para crear una RPC ruta g. GRPCel tráfico que tiene metadatos que comienzan por 123 se enruta a un nodo virtual denominadoserviceBgrpc. Si hay errores específicos (g) RPC o TCP errores al intentar comunicarse con el destino de la ruta, la ruta se vuelve a intentar tres veces. HTTP Hay un retraso de 15 segundos entre cada reintento.

```
aws appmesh create-route \  
  --cli-input-json file://create-route-grpc.json
```

Contenidos de create-route-grpc.json:

```
{  
  "meshName" : "apps",  
  "routeName" : "grpcRoute",  
  "spec" : {  
    "grpcRoute" : {  
      "action" : {  
        "weightedTargets" : [  
          {  
            "virtualNode" : "serviceBgrpc",  
            "weight" : 100  
          }  
        ]  
      },  
      "match" : {  
        "metadata" : [  
          {  
            "invert" : false,  
            "match" : {  
              "prefix" : "123"  
            },  
            "name" : "myMetadata"  
          }  
        ],  
        "methodName" : "GetColor",  
        "serviceName" : "com.amazonaws.services.ColorService"  
      },  
      "retryPolicy" : {  
        "grpcRetryEvents" : [ "deadline-exceeded" ],  
        "httpRetryEvents" : [ "server-error", "gateway-error" ],  
        "maxRetries" : 3,  
        "perRetryTimeout" : {  
          "unit" : "s",  
          "value" : 15  
        },  
        "tcpRetryEvents" : [ "connection-error" ]  
      }  
    },  
    "priority" : 100  
  }  
}
```



```

    },
    "virtualRouterName" : "serviceBgrpc"
  }

```

**Salida:**

```

{
  "route": {
    "meshName": "apps",
    "metadata": {
      "arn": "arn:aws:appmesh:us-west-2:123456789012:mesh/apps/virtualRouter/
serviceBgrpc/route/grpcRoute",
      "createdAt": 1572010806.008,
      "lastUpdatedAt": 1572010806.008,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "routeName": "grpcRoute",
    "spec": {
      "grpcRoute": {
        "action": {
          "weightedTargets": [
            {
              "virtualNode": "serviceBgrpc",
              "weight": 100
            }
          ]
        },
        "match": {
          "metadata": [
            {
              "invert": false,
              "match": {
                "prefix": "123"
              },
              "name": "mymetadata"
            }
          ],
          "methodName": "GetColor",
          "serviceName": "com.amazonaws.services.ColorService"
        },
        "retryPolicy": {
          "grpcRetryEvents": [

```

```

        "deadline-exceeded"
    ],
    "httpRetryEvents": [
        "server-error",
        "gateway-error"
    ],
    "maxRetries": 3,
    "perRetryTimeout": {
        "unit": "s",
        "value": 15
    },
    "tcpRetryEvents": [
        "connection-error"
    ]
    }
    },
    "priority": 100
},
"status": {
    "status": "ACTIVE"
},
"virtualRouterName": "serviceBgrpc"
}
}

```

Para crear una ruta nueva HTTP o HTTP /2

En el siguiente `create-route` ejemplo, se utiliza un archivo de JSON entrada para crear una ruta HTTP /2. Para crear una HTTP ruta, sustituya `Http2route` por `httpRoute` under `spec`. Todo el tráfico HTTP /2 dirigido a cualquier URL prefijo que tenga un valor de encabezado que comience por 123 se enruta a un nodo virtual denominado 2. `serviceBhttp` Si hay errores específicos HTTP o TCP errores al intentar comunicarse con el destino de la ruta, la ruta se vuelve a intentar tres veces. Hay un retraso de 15 segundos entre cada intento.

```

aws appmesh create-route \
  --cli-input-json file://create-route-http2.json

```

Contenidos de `create-route-http2.json`:

```

{
  "meshName": "apps",
  "routeName": "http2Route",

```

```
"spec": {
  "http2Route": {
    "action": {
      "weightedTargets": [
        {
          "virtualNode": "serviceBhttp2",
          "weight": 100
        }
      ]
    },
    "match": {
      "headers": [
        {
          "invert": false,
          "match": {
            "prefix": "123"
          },
          "name": "clientRequestId"
        }
      ],
      "method": "POST",
      "prefix": "/",
      "scheme": "http"
    },
    "retryPolicy": {
      "httpRetryEvents": [
        "server-error",
        "gateway-error"
      ],
      "maxRetries": 3,
      "perRetryTimeout": {
        "unit": "s",
        "value": 15
      },
      "tcpRetryEvents": [
        "connection-error"
      ]
    }
  },
  "priority": 200
},
"virtualRouterName": "serviceBhttp2"
}
```

## Salida:

```
{
  "route": {
    "meshName": "apps",
    "metadata": {
      "arn": "arn:aws:appmesh:us-west-2:123456789012:mesh/apps/virtualRouter/
serviceBhttp2/route/http2Route",
      "createdAt": 1572011008.352,
      "lastUpdatedAt": 1572011008.352,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "routeName": "http2Route",
    "spec": {
      "http2Route": {
        "action": {
          "weightedTargets": [
            {
              "virtualNode": "serviceBhttp2",
              "weight": 100
            }
          ]
        },
        "match": {
          "headers": [
            {
              "invert": false,
              "match": {
                "prefix": "123"
              },
              "name": "clientRequestId"
            }
          ],
          "method": "POST",
          "prefix": "/",
          "scheme": "http"
        },
        "retryPolicy": {
          "httpRetryEvents": [
            "server-error",
            "gateway-error"
          ],
          "maxRetries": 3,

```

```

        "perRetryTimeout": {
            "unit": "s",
            "value": 15
        },
        "tcpRetryEvents": [
            "connection-error"
        ]
    },
    "priority": 200
},
"status": {
    "status": "ACTIVE"
},
"virtualRouterName": "serviceBhttp2"
}
}

```

Para crear una ruta nueva TCP

El siguiente `create-route` ejemplo utiliza un archivo JSON de entrada para crear una TCP ruta. El 75 por ciento del tráfico se enruta a un nodo virtual denominado `serviceBtcp` y el 25 por ciento del tráfico se enruta a un nodo virtual denominado `serviceBv2tcp`. Especificar diferentes ponderaciones para distintos objetivos es una forma eficaz de implementar una nueva versión de una aplicación. Puede ajustar las ponderaciones para que, finalmente, el 100 por ciento de todo el tráfico se dirija a un destino que tenga la nueva versión de una aplicación.

```

aws appmesh create-route \
  --cli-input-json file://create-route-tcp.json

```

Contenido de `create-route-tcp.json`:

```

{
  "meshName": "apps",
  "routeName": "tcpRoute",
  "spec": {
    "priority": 300,
    "tcpRoute": {
      "action": {
        "weightedTargets": [
          {
            "virtualNode": "serviceBtcp",

```

```

        "weight": 75
      },
      {
        "virtualNode": "serviceBv2tcp",
        "weight": 25
      }
    ]
  }
},
"virtualRouterName": "serviceBtcp"
}

```

**Salida:**

```

{
  "route": {
    "meshName": "apps",
    "metadata": {
      "arn": "arn:aws:appmesh:us-west-2:123456789012:mesh/apps/virtualRouter/
serviceBtcp/route/tcpRoute",
      "createdAt": 1572011436.26,
      "lastUpdatedAt": 1572011436.26,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "routeName": "tcpRoute",
    "spec": {
      "priority": 300,
      "tcpRoute": {
        "action": {
          "weightedTargets": [
            {
              "virtualNode": "serviceBtcp",
              "weight": 75
            },
            {
              "virtualNode": "serviceBv2tcp",
              "weight": 25
            }
          ]
        }
      }
    }
  }
}

```

```

    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualRouterName": "serviceBtcp"
  }
}

```

Para obtener más información, consulta [Rutas](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [CreateRoute](#) la Referencia de AWS CLI comandos.

## create-virtual-gateway

En el siguiente ejemplo de código, se muestra cómo usar `create-virtual-gateway`.

### AWS CLI

Para crear una nueva puerta de enlace virtual

En el siguiente `create-virtual-gateway` ejemplo, se utiliza un archivo JSON de entrada para crear una puerta de enlace virtual con un detector para HTTP utilizar el puerto 9080.

```

aws appmesh create-virtual-gateway \
  --mesh-name meshName \
  --virtual-gateway-name virtualGatewayName \
  --cli-input-json file://create-virtual-gateway.json

```

Contenidos de `create-virtual-gateway.json`:

```

{
  "spec": {
    "listeners": [
      {
        "portMapping": {
          "port": 9080,
          "protocol": "http"
        }
      }
    ]
  }
}

```

Salida:

```
{
  "virtualGateway": {
    "meshName": "meshName",
    "metadata": {
      "arn": "arn:aws:appmesh:us-west-2:123456789012:mesh/meshName/
virtualGateway/virtualGatewayName",
      "createdAt": "2022-04-06T10:42:42.015000-05:00",
      "lastUpdatedAt": "2022-04-06T10:42:42.015000-05:00",
      "meshOwner": "123456789012",
      "resourceOwner": "123456789012",
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "spec": {
      "listeners": [
        {
          "portMapping": {
            "port": 9080,
            "protocol": "http"
          }
        }
      ]
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualGatewayName": "virtualGatewayName"
  }
}
```

Para obtener más información, consulte [Virtual Gateways](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [CreateVirtualGateway](#) la Referencia de AWS CLI comandos.

## create-virtual-node

En el siguiente ejemplo de código, se muestra cómo usar `create-virtual-node`.



## AWS CLI

Ejemplo 1: Para crear un nuevo nodo virtual que se utilice DNS para la detección

El siguiente `create-virtual-node` ejemplo utiliza un archivo JSON de entrada para crear un nodo virtual que se utiliza DNS para la detección de servicios.

```
aws appmesh create-virtual-node \  
  --cli-input-json file://create-virtual-node-dns.json
```

Contenidos de `create-virtual-node-dns.json`:

```
{  
  "meshName": "app1",  
  "spec": {  
    "listeners": [  
      {  
        "portMapping": {  
          "port": 80,  
          "protocol": "http"  
        }  
      }  
    ],  
    "serviceDiscovery": {  
      "dns": {  
        "hostname": "serviceBv1.svc.cluster.local"  
      }  
    }  
  },  
  "virtualNodeName": "vnServiceBv1"  
}
```

Salida:

```
{  
  "virtualNode": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/  
vnServiceBv1",  
      "createdAt": 1563810019.874,  
      "lastUpdatedAt": 1563810019.874,  
    }  
  }  
}
```

```

        "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "version": 1
    },
    "spec": {
        "listeners": [
            {
                "portMapping": {
                    "port": 80,
                    "protocol": "http"
                }
            }
        ],
        "serviceDiscovery": {
            "dns": {
                "hostname": "serviceBv1.svc.cluster.local"
            }
        }
    },
    "status": {
        "status": "ACTIVE"
    },
    "virtualNodeName": "vnServiceBv1"
}
}

```

Ejemplo 2: Para crear un nuevo nodo virtual que utilice AWS Cloud Map para la detección

En el siguiente `create-virtual-node` ejemplo, se usa un archivo de JSON entrada para crear un nodo virtual que usa AWS Cloud Map para la detección de servicios.

```

aws appmesh create-virtual-node \
  --cli-input-json file://create-virtual-node-cloud-map.json

```

Contenidos de `create-virtual-node-cloud-map.json`:

```

{
  "meshName": "app1",
  "spec": {
    "backends": [
      {
        "virtualService": {
          "virtualServiceName": "serviceA.svc.cluster.local"
        }
      }
    ]
  }
}

```

```

    }
  ],
  "listeners": [
    {
      "portMapping": {
        "port": 80,
        "protocol": "http"
      }
    }
  ],
  "serviceDiscovery": {
    "awsCloudMap": {
      "attributes": [
        {
          "key": "Environment",
          "value": "Testing"
        }
      ],
      "namespaceName": "namespace1",
      "serviceName": "serviceA"
    }
  },
  "virtualNodeName": "vnServiceA"
}

```

Salida:

```

{
  "virtualNode": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/vnServiceA",
      "createdAt": 1563810859.465,
      "lastUpdatedAt": 1563810859.465,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "spec": {
      "backends": [
        {
          "virtualService": {

```

```
        "virtualServiceName": "serviceA.svc.cluster.local"
      }
    },
    "listeners": [
      {
        "portMapping": {
          "port": 80,
          "protocol": "http"
        }
      }
    ],
    "serviceDiscovery": {
      "awsCloudMap": {
        "attributes": [
          {
            "key": "Environment",
            "value": "Testing"
          }
        ]
      },
      "namespaceName": "namespace1",
      "serviceName": "serviceA"
    }
  },
  "status": {
    "status": "ACTIVE"
  },
  "virtualNodeName": "vnServiceA"
}
}
```

Para obtener más información, consulte [Nodos virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [CreateVirtualNode](#) la Referencia de AWS CLI comandos.

## **create-virtual-router**

En el siguiente ejemplo de código, se muestra cómo usar `create-virtual-router`.

## AWS CLI

Para crear un nuevo router virtual

El siguiente `create-virtual-router` ejemplo usa un archivo JSON de entrada para crear un enrutador virtual con un listener para HTTP usar el puerto 80.

```
aws appmesh create-virtual-router \  
  --cli-input-json file://create-virtual-router.json
```

Contenidos de `create-virtual-router.json`:

```
{  
  "meshName": "app1",  
  "spec": {  
    "listeners": [  
      {  
        "portMapping": {  
          "port": 80,  
          "protocol": "http"  
        }  
      }  
    ]  
  },  
  "virtualRouterName": "vrServiceB"  
}
```

Salida:

```
{  
  "virtualRouter": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/  
vrServiceB",  
      "createdAt": 1563810546.59,  
      "lastUpdatedAt": 1563810546.59,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 1  
    },  
    "spec": {  
      "listeners": [  
        {
```

```

        "portMapping": {
            "port": 80,
            "protocol": "http"
        }
    ]
},
"status": {
    "status": "ACTIVE"
},
"virtualRouterName": "vrServiceB"
}
}

```

Para obtener más información, consulte [Enrutadores virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [CreateVirtualRouter](#) la Referencia de AWS CLI comandos.

## create-virtual-service

En el siguiente ejemplo de código, se muestra cómo usar `create-virtual-service`.

### AWS CLI

Ejemplo 1: Para crear un nuevo servicio virtual con un proveedor de nodos virtuales

El siguiente `create-virtual-service` ejemplo utiliza un archivo JSON de entrada para crear un servicio virtual con un proveedor de nodos virtuales.

```

aws appmesh create-virtual-service \
  --cli-input-json file://create-virtual-service-virtual-node.json

```

Contenidos de `create-virtual-service-virtual-node.json`:

```

{
  "meshName": "app1",
  "spec": {
    "provider": {
      "virtualNode": {
        "virtualNodeName": "vnServiceA"
      }
    }
  }
}

```

```

    }
  },
  "virtualServiceName": "serviceA.svc.cluster.local"
}

```

Salida:

```

{
  "virtualService": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceA.svc.cluster.local",
      "createdAt": 1563810859.474,
      "lastUpdatedAt": 1563810967.179,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 2
    },
    "spec": {
      "provider": {
        "virtualNode": {
          "virtualNodeName": "vnServiceA"
        }
      }
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualServiceName": "serviceA.svc.cluster.local"
  }
}

```

Para obtener más información, consulte [Virtual Node](#) en la Guía del usuario de AWS App Mesh.

Ejemplo 2: Para crear un nuevo servicio virtual con un proveedor de enrutadores virtuales

El siguiente `create-virtual-service` ejemplo utiliza un archivo JSON de entrada para crear un servicio virtual con un proveedor de enrutadores virtuales.

```

aws appmesh create-virtual-service \
  --cli-input-json file://create-virtual-service-virtual-router.json

```

## Contenidos de create-virtual-service-virtual-router.json:

```
{
  "meshName": "app1",
  "spec": {
    "provider": {
      "virtualRouter": {
        "virtualRouterName": "vrServiceB"
      }
    }
  },
  "virtualServiceName": "serviceB.svc.cluster.local"
}
```

## Salida:

```
{
  "virtualService": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceB.svc.cluster.local",
      "createdAt": 1563908363.999,
      "lastUpdatedAt": 1563908363.999,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "spec": {
      "provider": {
        "virtualRouter": {
          "virtualRouterName": "vrServiceB"
        }
      }
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualServiceName": "serviceB.svc.cluster.local"
  }
}
```

Para obtener más información, consulte Virtual Services < [https://docs.aws.amazon.com/app-mesh/latest/userguide/virtual\\_services.html](https://docs.aws.amazon.com/app-mesh/latest/userguide/virtual_services.html) > en la Guía del usuario de App Mesh AWS



- Para obtener API AWS CLI más información, [CreateVirtualService](#) consulte la Referencia de comandos.

## delete-mesh

En el siguiente ejemplo de código, se muestra cómo usar `delete-mesh`.

### AWS CLI

Para eliminar una malla de servicios

En el siguiente `delete-mesh` ejemplo, se elimina la malla de servicios especificada.

```
aws appmesh delete-mesh \  
  --mesh-name app1
```

Salida:

```
{  
  "mesh": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",  
      "createdAt": 1563809909.282,  
      "lastUpdatedAt": 1563824981.248,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 2  
    },  
    "spec": {  
      "egressFilter": {  
        "type": "ALLOW_ALL"  
      }  
    },  
    "status": {  
      "status": "DELETED"  
    }  
  }  
}
```

Para obtener más información, consulte [Service Mesh en la](#) Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [DeleteMesh](#) la Referencia de AWS CLI comandos.

## delete-route

En el siguiente ejemplo de código, se muestra cómo usar delete-route.

### AWS CLI

Para eliminar una ruta

En el siguiente delete-route ejemplo, se elimina la ruta especificada.

```
aws appmesh delete-route \  
  --mesh-name app1 \  
  --virtual-router-name vrServiceB \  
  --route-name toVnServiceB-weighted
```

Salida:

```
{  
  "route": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/  
vrServiceB/route/toVnServiceB-weighted",  
      "createdAt": 1563811384.015,  
      "lastUpdatedAt": 1563823915.936,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 3  
    },  
    "routeName": "toVnServiceB-weighted",  
    "spec": {  
      "httpRoute": {  
        "action": {  
          "weightedTargets": [  
            {  
              "virtualNode": "vnServiceBv1",  
              "weight": 80  
            },  
            {  
              "virtualNode": "vnServiceBv2",  
              "weight": 20  
            }  
          ]  
        }  
      }  
    },  
  },  
}
```

```

        "match": {
            "prefix": "/"
        }
    },
    "status": {
        "status": "DELETED"
    },
    "virtualRouterName": "vrServiceB"
}
}

```

Para obtener más información, consulta [Rutas](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [DeleteRoute](#) la Referencia de AWS CLI comandos.

## delete-virtual-node

En el siguiente ejemplo de código, se muestra cómo usar `delete-virtual-node`.

### AWS CLI

Para eliminar un nodo virtual

En el siguiente `delete-virtual-node` ejemplo, se elimina el nodo virtual especificado.

```

aws appmesh delete-virtual-node \
  --mesh-name app1 \
  --virtual-node-name vnServiceBv2

```

Salida:

```

{
  "virtualNode": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/vnServiceBv2",
      "createdAt": 1563810117.297,
      "lastUpdatedAt": 1563824700.678,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 2
    }
  },
}

```

```

    "spec": {
      "backends": [],
      "listeners": [
        {
          "portMapping": {
            "port": 80,
            "protocol": "http"
          }
        }
      ],
      "serviceDiscovery": {
        "dns": {
          "hostname": "serviceBv2.svc.cluster.local"
        }
      }
    },
    "status": {
      "status": "DELETED"
    },
    "virtualNodeName": "vnServiceBv2"
  }
}

```

Para obtener más información, consulte [Nodos virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [DeleteVirtualNode](#) la Referencia de AWS CLI comandos.

## delete-virtual-router

En el siguiente ejemplo de código, se muestra cómo usar `delete-virtual-router`.

### AWS CLI

Para eliminar un router virtual

En el siguiente `delete-virtual-router` ejemplo, se elimina el enrutador virtual especificado.

```

aws appmesh delete-virtual-router \
  --mesh-name app1 \
  --virtual-router-name vrServiceB

```

Salida:

```
{
  "virtualRouter": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/vrServiceB",
      "createdAt": 1563810546.59,
      "lastUpdatedAt": 1563824253.467,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 3
    },
    "spec": {
      "listeners": [
        {
          "portMapping": {
            "port": 80,
            "protocol": "http"
          }
        }
      ]
    },
    "status": {
      "status": "DELETED"
    },
    "virtualRouterName": "vrServiceB"
  }
}
```

Para obtener más información, consulte [Enrutadores virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [DeleteVirtualRouter](#) la Referencia de AWS CLI comandos.

## delete-virtual-service

En el siguiente ejemplo de código, se muestra cómo usar `delete-virtual-service`.

### AWS CLI

Para eliminar un servicio virtual

En el siguiente `delete-virtual-service` ejemplo, se elimina el servicio virtual especificado.

```
aws appmesh delete-virtual-service \  
  --mesh-name app1 \  
  --virtual-service-name serviceB.svc.cluster.local
```

Salida:

```
{  
  "virtualService": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/  
serviceB.svc.cluster.local",  
      "createdAt": 1563908363.999,  
      "lastUpdatedAt": 1563913940.866,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 3  
    },  
    "spec": {},  
    "status": {  
      "status": "DELETED"  
    },  
    "virtualServiceName": "serviceB.svc.cluster.local"  
  }  
}
```

Para obtener más información, consulte [Virtual Service](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [DeleteVirtualService](#) la Referencia de AWS CLI comandos.

## **describe-mesh**

En el siguiente ejemplo de código, se muestra cómo usar `describe-mesh`.

AWS CLI

Para describir una malla de servicios

El siguiente `describe-mesh` ejemplo devuelve detalles sobre la malla de servicios especificada.

```
aws appmesh describe-mesh \  
  --mesh-name app1
```

Salida:

```
{  
  "mesh": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",  
      "createdAt": 1563809909.282,  
      "lastUpdatedAt": 1563809909.282,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 1  
    },  
    "spec": {},  
    "status": {  
      "status": "ACTIVE"  
    }  
  }  
}
```

Para obtener más información, consulte [Service Mesh en la](#) Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [DescribeMesh](#) la Referencia de AWS CLI comandos.

## describe-route

En el siguiente ejemplo de código, se muestra cómo usar `describe-route`.

AWS CLI

Para describir una ruta

El siguiente `describe-route` ejemplo devuelve detalles sobre la ruta especificada.

```
aws appmesh describe-route \  
  --mesh-name app1 \  
  --virtual-router-name vrServiceB \  
  --route-name toVnServiceB-weighted
```

**Salida:**

```
{
  "route": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/vrServiceB/route/toVnServiceB-weighted",
      "createdAt": 1563811384.015,
      "lastUpdatedAt": 1563811384.015,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    },
    "routeName": "toVnServiceB-weighted",
    "spec": {
      "httpRoute": {
        "action": {
          "weightedTargets": [
            {
              "virtualNode": "vnServiceBv1",
              "weight": 90
            },
            {
              "virtualNode": "vnServiceBv2",
              "weight": 10
            }
          ]
        },
        "match": {
          "prefix": "/"
        }
      }
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualRouterName": "vrServiceB"
  }
}
```

Para obtener más información, consulta [Rutas](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [DescribeRoute](#) la Referencia de AWS CLI comandos.



## describe-virtual-node

En el siguiente ejemplo de código, se muestra cómo usar `describe-virtual-node`.

### AWS CLI

Para describir un nodo virtual

El siguiente `describe-virtual-node` ejemplo devuelve detalles sobre el nodo virtual especificado.

```
aws appmesh describe-virtual-node \  
  --mesh-name app1 \  
  --virtual-node-name vnServiceBv1
```

Salida:

```
{  
  "virtualNode": {  
    "meshName": "app1",  
    "metadata": {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/  
vnServiceBv1",  
      "createdAt": 1563810019.874,  
      "lastUpdatedAt": 1563810019.874,  
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
      "version": 1  
    },  
    "spec": {  
      "backends": [],  
      "listeners": [  
        {  
          "portMapping": {  
            "port": 80,  
            "protocol": "http"  
          }  
        }  
      ],  
      "serviceDiscovery": {  
        "dns": {  
          "hostname": "serviceBv1.svc.cluster.local"  
        }  
      }  
    }  
  }  
}
```

```

    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualNodeName": "vnServiceBv1"
  }
}

```

Para obtener más información, consulte [Nodos virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [DescribeVirtualNode](#) la Referencia de AWS CLI comandos.

## describe-virtual-router

En el siguiente ejemplo de código, se muestra cómo usar `describe-virtual-router`.

### AWS CLI

Para describir un router virtual

El siguiente `describe-virtual-router` ejemplo devuelve detalles sobre el enrutador virtual especificado.

```

aws appmesh describe-virtual-router \
  --mesh-name app1 \
  --virtual-router-name vrServiceB

```

Salida:

```

{
  "virtualRouter": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/vrServiceB",
      "createdAt": 1563810546.59,
      "lastUpdatedAt": 1563810546.59,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 1
    }
  }
}

```

```
    },
    "spec": {
      "listeners": [
        {
          "portMapping": {
            "port": 80,
            "protocol": "http"
          }
        }
      ]
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualRouterName": "vrServiceB"
  }
}
```

Para obtener más información, consulte [Enrutadores virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [DescribeVirtualRouter](#) la Referencia de AWS CLI comandos.

## describe-virtual-service

En el siguiente ejemplo de código, se muestra cómo usar `describe-virtual-service`.

### AWS CLI

Para describir un servicio virtual

El siguiente `describe-virtual-service` ejemplo devuelve detalles sobre el servicio virtual especificado.

```
aws appmesh describe-virtual-service \
  --mesh-name app1 \
  --virtual-service-name serviceB.svc.cluster.local
```

Salida:

```
{
```

```
"virtualService": {
  "meshName": "app1",
  "metadata": {
    "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceB.svc.cluster.local",
    "createdAt": 1563908363.999,
    "lastUpdatedAt": 1563908363.999,
    "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "version": 1
  },
  "spec": {
    "provider": {
      "virtualRouter": {
        "virtualRouterName": "vrServiceB"
      }
    }
  },
  "status": {
    "status": "ACTIVE"
  },
  "virtualServiceName": "serviceB.svc.cluster.local"
}
}
```

Para obtener más información, consulte [Servicios virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [DescribeVirtualService](#) la Referencia de AWS CLI comandos.

## list-meshes

En el siguiente ejemplo de código, se muestra cómo usar `list-meshes`.

### AWS CLI

Para enumerar las mallas de servicio

En el siguiente `list-meshes` ejemplo, se enumeran todas las mallas de servicio de la región actual AWS .

```
aws appmesh list-meshes
```

Salida:

```
{
  "meshes": [
    {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",
      "meshName": "app1"
    }
  ]
}
```

Para obtener más información, consulte [Service Mesh en la](#) Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [ListMeshes](#) la Referencia de AWS CLI comandos.

## list-routes

En el siguiente ejemplo de código, se muestra cómo usar `list-routes`.

AWS CLI

Para enumerar las rutas

En el siguiente `list-routes` ejemplo, se enumeran todas las rutas del enrutador virtual especificado.

```
aws appmesh list-routes \
  --mesh-name app1 \
  --virtual-router-name vrServiceB
```

Salida:

```
{
  "routes": [
    {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB/route/toVnServiceB",
      "meshName": "app1",
      "routeName": "toVnServiceB-weighted",
      "virtualRouterName": "vrServiceB"
    }
  ]
}
```

```
}
```

Para obtener más información, consulta [Rutas](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [ListRoutes](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas asignadas al recurso especificado.

```
aws appmesh list-tags-for-resource \
  --resource-arn arn:aws:appmesh:us-east-1:123456789012:mesh/app1
```

Salida:

```
{
  "tags": [
    {
      "key": "key1",
      "value": "value1"
    },
    {
      "key": "key2",
      "value": "value2"
    },
    {
      "key": "key3",
      "value": "value3"
    }
  ]
}
```

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## list-virtual-nodes

En el siguiente ejemplo de código, se muestra cómo usar `list-virtual-nodes`.

### AWS CLI

Para enumerar los nodos virtuales

En el siguiente `list-virtual-nodes` ejemplo, se enumeran todos los nodos virtuales de la malla de servicios especificada.

```
aws appmesh list-virtual-nodes \  
  --mesh-name app1
```

Salida:

```
{  
  "virtualNodes": [  
    {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/  
vnServiceBv1",  
      "meshName": "app1",  
      "virtualNodeName": "vnServiceBv1"  
    },  
    {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/  
vnServiceBv2",  
      "meshName": "app1",  
      "virtualNodeName": "vnServiceBv2"  
    }  
  ]  
}
```

Para obtener más información, consulte [Nodos virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [ListVirtualNodes](#) la Referencia de AWS CLI comandos.

## list-virtual-routers

En el siguiente ejemplo de código, se muestra cómo usar `list-virtual-routers`.

## AWS CLI

Para enumerar los enrutadores virtuales

En el siguiente `list-virtual-routers` ejemplo, se enumeran todos los enrutadores virtuales de la malla de servicios especificada.

```
aws appmesh list-virtual-routers \  
  --mesh-name app1
```

Salida:

```
{  
  "virtualRouters": [  
    {  
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/  
vrServiceB",  
      "meshName": "app1",  
      "virtualRouterName": "vrServiceB"  
    }  
  ]  
}
```

Para obtener más información, consulte [Enrutadores virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [ListVirtualRouters](#) la Referencia de AWS CLI comandos.

## `list-virtual-services`

En el siguiente ejemplo de código, se muestra cómo usar `list-virtual-services`.

## AWS CLI

Para enumerar los servicios virtuales

En el siguiente `list-virtual-services` ejemplo, se enumeran todos los servicios virtuales de la malla de servicios especificada.

```
aws appmesh list-virtual-services \  
  --mesh-name app1
```



**Salida:**

```
{
  "virtualServices": [
    {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceA.svc.cluster.local",
      "meshName": "app1",
      "virtualServiceName": "serviceA.svc.cluster.local"
    },
    {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceB.svc.cluster.local",
      "meshName": "app1",
      "virtualServiceName": "serviceB.svc.cluster.local"
    }
  ]
}
```

Para obtener más información, consulte [Servicios virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [ListVirtualServices](#) la Referencia de AWS CLI comandos.

**tag-resource**

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

**AWS CLI**

Para etiquetar un recurso

En el siguiente `tag-resource` ejemplo, se agrega la etiqueta `key1` con el valor `value1` al recurso especificado.

```
aws appmesh tag-resource \
  --resource-arn arn:aws:appmesh:us-east-1:123456789012:mesh/app1 \
  --tags key=key1,value=value1
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para quitar la etiqueta de un recurso

En el siguiente `untag-resource` ejemplo, se quita una etiqueta con la clave `key1` del recurso especificado.

```
aws appmesh untag-resource \  
  --resource-arn arn:aws:appmesh:us-east-1:123456789012:mesh/app1 \  
  --tag-keys key1
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-mesh

En el siguiente ejemplo de código, se muestra cómo usar `update-mesh`.

### AWS CLI

Para actualizar una malla de servicios

El siguiente `update-mesh` ejemplo utiliza un archivo JSON de entrada para actualizar una malla de servicios y permitir que todo el tráfico de salida externo se reenvíe intacto a través del proxy de Envoy.

```
aws appmesh update-mesh \  
  --cli-input-json file://update-mesh.json
```

Contenidos de `update-mesh.json`:

```
{
```

```
"meshName": "app1",
"spec": {
  "egressFilter": {
    "type": "ALLOW_ALL"
  }
}
```

Salida:

```
{
  "mesh": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",
      "createdAt": 1563809909.282,
      "lastUpdatedAt": 1563812829.687,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 2
    },
    "spec": {
      "egressFilter": {
        "type": "ALLOW_ALL"
      }
    },
    "status": {
      "status": "ACTIVE"
    }
  }
}
```

Para obtener más información, consulte [Service Mesh en la](#) Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [UpdateMesh](#) la Referencia de AWS CLI comandos.

## update-route

En el siguiente ejemplo de código, se muestra cómo usar `update-route`.

### AWS CLI

Para actualizar una ruta

En el siguiente `update-route` ejemplo, se utiliza un archivo de JSON entrada para actualizar los pesos de una ruta.

```
aws appmesh update-route \
  --cli-input-json file://update-route-weighted.json
```

Contenidos de `update-route-weighted.json`:

```
{
  "meshName": "app1",
  "routeName": "toVnServiceB-weighted",
  "spec": {
    "httpRoute": {
      "action": {
        "weightedTargets": [
          {
            "virtualNode": "vnServiceBv1",
            "weight": 80
          },
          {
            "virtualNode": "vnServiceBv2",
            "weight": 20
          }
        ]
      },
      "match": {
        "prefix": "/"
      }
    }
  },
  "virtualRouterName": "vrServiceB"
}
```

Salida:

```
{
  "route": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/vrServiceB/route/toVnServiceB-weighted",
      "createdAt": 1563811384.015,

```

```
    "lastUpdatedAt": 1563819600.022,
    "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "version": 2
  },
  "routeName": "toVnServiceB-weighted",
  "spec": {
    "httpRoute": {
      "action": {
        "weightedTargets": [
          {
            "virtualNode": "vnServiceBv1",
            "weight": 80
          },
          {
            "virtualNode": "vnServiceBv2",
            "weight": 20
          }
        ]
      },
      "match": {
        "prefix": "/"
      }
    }
  },
  "status": {
    "status": "ACTIVE"
  },
  "virtualRouterName": "vrServiceB"
}
}
```

Para obtener más información, consulta [Rutas](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [UpdateRoute](#) la Referencia de AWS CLI comandos.

## update-virtual-node

En el siguiente ejemplo de código, se muestra cómo usar `update-virtual-node`.

### AWS CLI

Para actualizar un nodo virtual

En el siguiente `update-virtual-node` ejemplo, se utiliza un archivo de JSON entrada para añadir una comprobación de estado a un nodo virtual.

```
aws appmesh update-virtual-node \  
--cli-input-json file://update-virtual-node.json
```

Contenidos de `update-virtual-node.json`:

```
{  
  "clientToken": "500",  
  "meshName": "app1",  
  "spec": {  
    "listeners": [  
      {  
        "healthCheck": {  
          "healthyThreshold": 5,  
          "intervalMillis": 10000,  
          "path": "/",  
          "port": 80,  
          "protocol": "http",  
          "timeoutMillis": 3000,  
          "unhealthyThreshold": 3  
        },  
        "portMapping": {  
          "port": 80,  
          "protocol": "http"  
        }  
      }  
    ],  
    "serviceDiscovery": {  
      "dns": {  
        "hostname": "serviceBv1.svc.cluster.local"  
      }  
    }  
  },  
  "virtualNodeName": "vnServiceBv1"  
}
```

Salida:

```
{  
  "virtualNode": {
```

```
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/
vnServiceBv1",
      "createdAt": 1563810019.874,
      "lastUpdatedAt": 1563819234.825,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 2
    },
    "spec": {
      "listeners": [
        {
          "healthCheck": {
            "healthyThreshold": 5,
            "intervalMillis": 10000,
            "path": "/",
            "port": 80,
            "protocol": "http",
            "timeoutMillis": 3000,
            "unhealthyThreshold": 3
          },
          "portMapping": {
            "port": 80,
            "protocol": "http"
          }
        }
      ],
      "serviceDiscovery": {
        "dns": {
          "hostname": "serviceBv1.svc.cluster.local"
        }
      }
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualNodeName": "vnServiceBv1"
  }
}
```

Para obtener más información, consulte [Nodos virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [UpdateVirtualRouter](#) la Referencia de AWS CLI comandos.

## update-virtual-router

En el siguiente ejemplo de código, se muestra cómo usar `update-virtual-router`.

### AWS CLI

Para actualizar un router virtual

El siguiente `update-virtual-router` ejemplo utiliza un archivo JSON de entrada para actualizar el puerto de escucha de un router virtual.

```
aws appmesh update-virtual-router \  
  --cli-input-json file://update-virtual-router.json
```

Contenidos de `update-virtual-router.json`:

```
{  
  "meshName": "app1",  
  "spec": {  
    "listeners": [  
      {  
        "portMapping": {  
          "port": 8080,  
          "protocol": "http"  
        }  
      }  
    ]  
  },  
  "virtualRouterName": "vrServiceB"  
}
```

Salida:

```
{  
  "virtualRouter": {  
    "meshName": "app1",  
    "metadata": {
```



```

    "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB",
    "createdAt": 1563810546.59,
    "lastUpdatedAt": 1563819431.352,
    "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "version": 2
  },
  "spec": {
    "listeners": [
      {
        "portMapping": {
          "port": 8080,
          "protocol": "http"
        }
      }
    ]
  },
  "status": {
    "status": "ACTIVE"
  },
  "virtualRouterName": "vrServiceB"
}
}

```

Para obtener más información, consulte [Enrutadores virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [UpdateVirtualRouter](#) la Referencia de AWS CLI comandos.

## update-virtual-service

En el siguiente ejemplo de código, se muestra cómo usar `update-virtual-service`.

### AWS CLI

Para actualizar un servicio virtual

En el siguiente `update-virtual-service` ejemplo, se utiliza un archivo de JSON entrada para actualizar un servicio virtual y utilizar un proveedor de enrutadores virtuales.

```
aws appmesh update-virtual-service \
```

```
--cli-input-json file://update-virtual-service.json
```

Contenidos de `update-virtual-service.json`:

```
{
  "meshName": "app1",
  "spec": {
    "provider": {
      "virtualRouter": {
        "virtualRouterName": "vrServiceA"
      }
    }
  },
  "virtualServiceName": "serviceA.svc.cluster.local"
}
```

Salida:

```
{
  "virtualService": {
    "meshName": "app1",
    "metadata": {
      "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceA.svc.cluster.local",
      "createdAt": 1563810859.474,
      "lastUpdatedAt": 1563820257.411,
      "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "version": 3
    },
    "spec": {
      "provider": {
        "virtualRouter": {
          "virtualRouterName": "vrServiceA"
        }
      }
    },
    "status": {
      "status": "ACTIVE"
    },
    "virtualServiceName": "serviceA.svc.cluster.local"
  }
}
```

Para obtener más información, consulte [Servicios virtuales](#) en la Guía del usuario de AWS App Mesh.

- Para API obtener más información, consulte [UpdateVirtualService](#) la Referencia de AWS CLI comandos.

## Ejemplos de App Runner que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con App Runner.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puedes encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **associate-custom-domain**

En el siguiente ejemplo de código, se muestra cómo usar `associate-custom-domain`.

AWS CLI

Para asociar un nombre de dominio y el subdominio `www` a un servicio

El siguiente `associate-custom-domain` ejemplo asocia un nombre de dominio personalizado que usted controla a un servicio de App Runner. El nombre de dominio es el dominio raíz `example.com`, incluido el subdominio en mayúsculas y minúsculas. `www.example.com`

```
aws apprunner associate-custom-domain \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
  "DomainName": "example.com",
  "EnableWWWSubdomain": true
}
```

Salida:

```
{
  "CustomDomain": {
    "CertificateValidationRecords": [
      {
        "Name": "_70d3f50a94f7c72dc28784cf55db2f6b.example.com",
        "Status": "PENDING_VALIDATION",
        "Type": "CNAME",
        "Value": "_1270c137383c6307b6832db02504c4b0.bsgbmzkfwj.acm-
validations.aws."
      },
      {
        "Name": "_287870d3f50a94f7c72dc4cf55db2f6b.www.example.com",
        "Status": "PENDING_VALIDATION",
        "Type": "CNAME",
        "Value": "_832db01270c137383c6307b62504c4b0.mzkbsgbfwj.acm-
validations.aws."
      }
    ],
    "DomainName": "example.com",
    "EnableWWWSubdomain": true,
    "Status": "CREATING"
  },
  "DNSTarget": "psbqam834h.us-east-1.awsapprunner.com",
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

- Para API obtener más información, consulte la Referencia de [AssociateCustomDomain](#) comandos AWS CLI .

## create-auto-scaling-configuration

En el siguiente ejemplo de código, se muestra cómo usar `create-auto-scaling-configuration`.

### AWS CLI

Para crear una configuración de escalado automático de alta disponibilidad

En el siguiente `create-auto-scaling-configuration` ejemplo, se crea una configuración de escalado automático optimizada para una alta disponibilidad `MinSize` con un valor de 5. Con esta configuración, App Runner intenta distribuir las instancias de servicio en el mayor número posible de zonas de disponibilidad, hasta cinco, según la AWS región.

La llamada devuelve un `AutoScalingConfiguration` objeto con el resto de la configuración establecida en sus valores predeterminados. En el ejemplo, esta es la primera llamada para crear una configuración denominada `high-availability`. La revisión se establece en 1 y es la última revisión.

```
aws apprunner create-auto-scaling-configuration \  
--cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "AutoScalingConfigurationName": "high-availability",  
  "MinSize": 5  
}
```

Salida:

```
{  
  "AutoScalingConfiguration": {  
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-  
east-1:123456789012:autoscalingconfiguration/high-  
availability/1/2f50e7656d7819fead0f59672e68042e",  
    "AutoScalingConfigurationName": "high-availability",  
    "AutoScalingConfigurationRevision": 1,  
    "CreatedAt": "2020-11-03T00:29:17Z",  
    "Latest": true,  
    "Status": "ACTIVE",
```

```
    "MaxConcurrency": 100,  
    "MaxSize": 50,  
    "MinSize": 5  
  }  
}
```

- Para API obtener más información, consulte [CreateAutoScalingConfiguration](#) la Referencia de AWS CLI comandos.

## create-connection

En el siguiente ejemplo de código, se muestra cómo usar `create-connection`.

### AWS CLI

Para crear una GitHub conexión

El siguiente `create-connection` ejemplo crea una conexión a un repositorio de GitHub código privado. El estado de la conexión después de una llamada exitosa es `PENDING_HANDSHAKE`. Esto se debe a que aún no se ha realizado un apretón de manos de autenticación con el proveedor. Completa el apretón de manos con la consola de App Runner.

```
aws apprunner create-connection \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "ConnectionName": "my-github-connection",  
  "ProviderType": "GITHUB"  
}
```

Salida:

```
{  
  "Connection": {  
    "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-github-connection",  
    "ConnectionName": "my-github-connection",  
    "Status": "PENDING_HANDSHAKE",  
    "CreatedAt": "2020-11-03T00:32:51Z",
```

```

    "ProviderType": "GITHUB"
  }
}

```

Para obtener más información, consulte [Administrar las conexiones de App Runner](#) en la Guía para desarrolladores de AWS App Runner.

- Para API obtener más información, consulte [CreateConnection](#) la Referencia de AWS CLI comandos.

## create-service

En el siguiente ejemplo de código, se muestra cómo usar `create-service`.

### AWS CLI

Ejemplo 1: Para crear un servicio de repositorio de código fuente

En el siguiente `create-service` ejemplo, se crea un servicio de App Runner basado en un repositorio de código fuente de Python.

```

aws apprunner create-service \
  --cli-input-json file://input.json

```

Contenidos de `input.json`:

```

{
  "ServiceName": "python-app",
  "SourceConfiguration": {
    "AuthenticationConfiguration": {
      "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-github-connection/e7656250f67242d7819feade6800f59e"
    },
    "AutoDeploymentsEnabled": true,
    "CodeRepository": {
      "RepositoryUrl": "https://github.com/my-account/python-hello",
      "SourceCodeVersion": {
        "Type": "BRANCH",
        "Value": "main"
      },
    },
    "CodeConfiguration": {
      "ConfigurationSource": "API",
    }
  }
}

```

```

        "CodeConfigurationValues": {
            "Runtime": "PYTHON_3",
            "BuildCommand": "pip install -r requirements.txt",
            "StartCommand": "python server.py",
            "Port": "8080",
            "RuntimeEnvironmentVariables": [
                {
                    "NAME": "Jane"
                }
            ]
        }
    },
    "InstanceConfiguration": {
        "CPU": "1 vCPU",
        "Memory": "3 GB"
    }
}

```

**Salida:**

```

{
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
  "Service": {
    "CreatedAt": "2020-11-20T19:05:25Z",
    "UpdatedAt": "2020-11-20T19:05:25Z",
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceName": "python-app",
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
    "SourceConfiguration": {
      "AuthenticationConfiguration": {
        "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
      },
      "AutoDeploymentsEnabled": true,
      "CodeRepository": {
        "CodeConfiguration": {
          "CodeConfigurationValues": {
            "BuildCommand": "pip install -r requirements.txt",

```



```

        "Port": "8080",
        "Runtime": "PYTHON_3",
        "RuntimeEnvironmentVariables": [
            {
                "NAME": "Jane"
            }
        ],
        "StartCommand": "python server.py"
    },
    "ConfigurationSource": "Api"
},
"RepositoryUrl": "https://github.com/my-account/python-hello",
"SourceCodeVersion": {
    "Type": "BRANCH",
    "Value": "main"
}
}
},
"Status": "OPERATION_IN_PROGRESS",
"InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "3 GB"
}
}
}

```

Ejemplo 2: Para crear un servicio de repositorio de código fuente

En el siguiente `create-service` ejemplo, se crea un servicio de App Runner basado en un repositorio de código fuente de Python.

```

aws apprunner create-service \
  --cli-input-json file://input.json

```

Contenidos de `input.json`:

```

{
  "ServiceName": "python-app",
  "SourceConfiguration": {
    "AuthenticationConfiguration": {
      "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-github-connection/e7656250f67242d7819feade6800f59e"
    },

```

```

    "AutoDeploymentsEnabled": true,
    "CodeRepository": {
      "RepositoryUrl": "https://github.com/my-account/python-hello",
      "SourceCodeVersion": {
        "Type": "BRANCH",
        "Value": "main"
      },
    },
    "CodeConfiguration": {
      "ConfigurationSource": "API",
      "CodeConfigurationValues": {
        "Runtime": "PYTHON_3",
        "BuildCommand": "pip install -r requirements.txt",
        "StartCommand": "python server.py",
        "Port": "8080",
        "RuntimeEnvironmentVariables": [
          {
            "NAME": "Jane"
          }
        ]
      }
    },
  },
  "InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "3 GB"
  }
}

```

**Salida:**

```

{
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
  "Service": {
    "CreatedAt": "2020-11-20T19:05:25Z",
    "UpdatedAt": "2020-11-20T19:05:25Z",
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceName": "python-app",
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
    "SourceConfiguration": {
      "AuthenticationConfiguration": {

```

```

    "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
  },
  "AutoDeploymentsEnabled": true,
  "CodeRepository": {
    "CodeConfiguration": {
      "CodeConfigurationValues": {
        "BuildCommand": "pip install -r requirements.txt",
        "Port": "8080",
        "Runtime": "PYTHON_3",
        "RuntimeEnvironmentVariables": [
          {
            "NAME": "Jane"
          }
        ],
        "StartCommand": "python server.py"
      },
      "ConfigurationSource": "Api"
    },
    "RepositoryUrl": "https://github.com/my-account/python-hello",
    "SourceCodeVersion": {
      "Type": "BRANCH",
      "Value": "main"
    }
  }
},
"Status": "OPERATION_IN_PROGRESS",
"InstanceConfiguration": {
  "CPU": "1 vCPU",
  "Memory": "3 GB"
}
}
}

```

Ejemplo 3: Para crear un servicio de repositorio de imágenes de origen

En el siguiente `create-service` ejemplo, se crea un servicio de App Runner basado en una imagen almacenada en Elastic Container Registry (ECR).

```

aws apprunner create-service \
  --cli-input-json file://input.json

```

## Contenidos de `input.json`:

```
{
  "ServiceName": "golang-container-app",
  "SourceConfiguration": {
    "AuthenticationConfiguration": {
      "AccessRoleArn": "arn:aws:iam::123456789012:role/my-ecr-role"
    },
    "AutoDeploymentsEnabled": true,
    "ImageRepository": {
      "ImageIdentifier": "123456789012.dkr.ecr.us-east-1.amazonaws.com/golang-
app:latest",
      "ImageConfiguration": {
        "Port": "8080",
        "RuntimeEnvironmentVariables": [
          {
            "NAME": "Jane"
          }
        ]
      },
      "ImageRepositoryType": "ECR"
    }
  },
  "InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "3 GB"
  }
}
```

## Salida:

```
{
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
  "Service": {
    "CreatedAt": "2020-11-06T23:15:30Z",
    "UpdatedAt": "2020-11-06T23:15:30Z",
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/golang-
container-app/51728f8a20ce46d39b25398a6c8e9d1a",
    "ServiceId": "51728f8a20ce46d39b25398a6c8e9d1a",
    "ServiceName": "golang-container-app",
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
    "SourceConfiguration": {
      "AuthenticationConfiguration": {
```

```

        "AccessRoleArn": "arn:aws:iam::123456789012:role/my-ecr-role"
    },
    "AutoDeploymentsEnabled": true,
    "ImageRepository": {
        "ImageIdentifier": "123456789012.dkr.ecr.us-east-1.amazonaws.com/
golang-app:latest",
        "ImageConfiguration": {
            "Port": "8080",
            "RuntimeEnvironmentVariables": [
                {
                    "NAME": "Jane"
                }
            ]
        },
        "ImageRepositoryType": "ECR"
    }
},
"Status": "OPERATION_IN_PROGRESS",
"InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "3 GB"
}
}
}

```

- Para API obtener más información, consulte [CreateService](#) la Referencia de AWS CLI comandos.

## delete-auto-scaling-configuration

En el siguiente ejemplo de código, se muestra cómo usar `delete-auto-scaling-configuration`.

### AWS CLI

Ejemplo 1: Para eliminar la última revisión activa de una configuración de autoescalado

En el siguiente `delete-auto-scaling-configuration` ejemplo, se elimina la última revisión activa de una configuración de autoescalado de App Runner. Para eliminar la última revisión activa, especifique un nombre de recurso de Amazon (ARN) que termine con el nombre de la configuración, sin el componente de revisión.

En el ejemplo, existen dos revisiones antes de esta acción. Por lo tanto, se suprime la revisión 2 (la más reciente). Sin embargo, ahora se muestra porque "Latest": false, después de ser eliminada, ya no es la última revisión activa.

```
aws apprunner delete-auto-scaling-configuration \  
  --cli-input-json file://input.json
```

Contenidos de input.json:

```
{  
  "AutoScalingConfigurationArn": "arn:aws:apprunner:us-  
east-1:123456789012:autoscalingconfiguration/high-availability"  
}
```

Salida:

```
{  
  "AutoScalingConfiguration": {  
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-  
east-1:123456789012:autoscalingconfiguration/high-availability/2/  
e76562f50d78042e819fead0f59672e6",  
    "AutoScalingConfigurationName": "high-availability",  
    "AutoScalingConfigurationRevision": 2,  
    "CreatedAt": "2021-02-25T17:42:59Z",  
    "DeletedAt": "2021-03-02T08:07:06Z",  
    "Latest": false,  
    "Status": "INACTIVE",  
    "MaxConcurrency": 30,  
    "MaxSize": 90,  
    "MinSize": 5  
  }  
}
```

Ejemplo 2: Para eliminar una revisión específica de una configuración de autoescalado

En el siguiente delete-auto-scaling-configuration ejemplo, se elimina una revisión específica de una configuración de autoescalado de App Runner. Para eliminar una revisión específica, especifique una ARN que incluya el número de revisión.

En el ejemplo, existen varias revisiones antes de esta acción. La acción elimina la revisión1.

```
aws apprunner delete-auto-scaling-configuration \  
  --cli-input-json file://input.json
```

```
--cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability/1"
}
```

Salida:

```
{
  "AutoScalingConfiguration": {
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-
availability/1/2f50e7656d7819fead0f59672e68042e",
    "AutoScalingConfigurationName": "high-availability",
    "AutoScalingConfigurationRevision": 1,
    "CreatedAt": "2020-11-03T00:29:17Z",
    "DeletedAt": "2021-03-02T08:07:06Z",
    "Latest": false,
    "Status": "INACTIVE",
    "MaxConcurrency": 100,
    "MaxSize": 50,
    "MinSize": 5
  }
}
```

- Para API obtener más información, consulte [DeleteAutoScalingConfiguration](#) la Referencia de AWS CLI comandos.

## **delete-connection**

En el siguiente ejemplo de código, se muestra cómo usar `delete-connection`.

### AWS CLI

Para eliminar una conexión

En el siguiente `delete-connection` ejemplo, se elimina una conexión de App Runner. El estado de la conexión después de una llamada exitosa es `DELETED`. Esto se debe a que la conexión ya no está disponible.

```
aws apprunner delete-connection \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-github-  
connection"  
}
```

Salida:

```
{  
  "Connection": {  
    "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-  
github-connection",  
    "ConnectionName": "my-github-connection",  
    "Status": "DELETED",  
    "CreatedAt": "2020-11-03T00:32:51Z",  
    "ProviderType": "GITHUB"  
  }  
}
```

- Para API obtener más información, consulte [DeleteConnection](#) la Referencia de AWS CLI comandos.

## delete-service

En el siguiente ejemplo de código, se muestra cómo usar `delete-service`.

### AWS CLI

Eliminación de un servicio

En el siguiente `delete-service` ejemplo, se elimina un servicio de App Runner.

```
aws apprunner delete-service \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:



```
{
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

**Salida:**

```
{
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
  "Service": {
    "CreatedAt": "2020-11-20T19:05:25Z",
    "UpdatedAt": "2020-11-20T19:05:25Z",
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceName": "python-app",
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
    "SourceConfiguration": {
      "AuthenticationConfiguration": {
        "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
      },
      "AutoDeploymentsEnabled": true,
      "CodeRepository": {
        "CodeConfiguration": {
          "CodeConfigurationValues": {
            "BuildCommand": "pip install -r requirements.txt",
            "Port": "8080",
            "Runtime": "PYTHON_3",
            "RuntimeEnvironmentVariables": [
              {
                "NAME": "Jane"
              }
            ],
            "StartCommand": "python server.py"
          },
          "ConfigurationSource": "Api"
        },
        "RepositoryUrl": "https://github.com/my-account/python-hello",
        "SourceCodeVersion": {
          "Type": "BRANCH",
          "Value": "main"
        }
      }
    }
  }
}
```

```
    }
  },
  "Status": "OPERATION_IN_PROGRESS",
  "InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "3 GB"
  }
}
```

- Para API obtener más información, consulte [DeleteService](#) la Referencia de AWS CLI comandos.

## describe-auto-scaling-configuration

En el siguiente ejemplo de código, se muestra cómo usar `describe-auto-scaling-configuration`.

### AWS CLI

Ejemplo 1: Para describir la última revisión activa de una configuración de autoescalado

En el siguiente `describe-auto-scaling-configuration` ejemplo, se obtiene una descripción de la última revisión activa de una configuración de autoescalado de App Runner. Para describir la última revisión activa, especifique una ARN que termine con el nombre de la configuración, sin el componente de revisión.

En el ejemplo, existen dos revisiones. Por lo tanto, se describe la revisión 2 (la más reciente). Se muestra el objeto resultante "Latest": true.

```
aws apprunner describe-auto-scaling-configuration \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability"
}
```

Salida:

```
{
  "AutoScalingConfiguration": {
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability/2/
e76562f50d78042e819fead0f59672e6",
    "AutoScalingConfigurationName": "high-availability",
    "AutoScalingConfigurationRevision": 2,
    "CreatedAt": "2021-02-25T17:42:59Z",
    "Latest": true,
    "Status": "ACTIVE",
    "MaxConcurrency": 30,
    "MaxSize": 90,
    "MinSize": 5
  }
}
```

Ejemplo 2: Para describir una revisión específica de una configuración de autoescalado

En el siguiente `describe-auto-scaling-configuration` ejemplo, se obtiene una descripción de una revisión específica de una configuración de autoescalado de App Runner. Para describir una revisión específica, especifique una ARN que incluya el número de revisión.

En el ejemplo, existen varias revisiones y 1 se consulta la revisión. Se muestra `"Latest": false` el objeto resultante.

```
aws apprunner describe-auto-scaling-configuration \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability/1"
}
```

Salida:

```
{
  "AutoScalingConfiguration": {
```

```

    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-
availability/1/2f50e7656d7819fead0f59672e68042e",
    "AutoScalingConfigurationName": "high-availability",
    "AutoScalingConfigurationRevision": 1,
    "CreatedAt": "2020-11-03T00:29:17Z",
    "Latest": false,
    "Status": "ACTIVE",
    "MaxConcurrency": 100,
    "MaxSize": 50,
    "MinSize": 5
  }
}

```

- Para API obtener más información, consulte [DescribeAutoScalingConfiguration](#) la Referencia de AWS CLI comandos.

## describe-custom-domains

En el siguiente ejemplo de código, se muestra cómo usar `describe-custom-domains`.

### AWS CLI

Para obtener descripciones de los nombres de dominio personalizados asociados a un servicio

En el siguiente `describe-custom-domains` ejemplo, se obtienen las descripciones y el estado de los nombres de dominio personalizados asociados a un servicio de App Runner.

```

aws apprunner describe-custom-domains \
  --cli-input-json file://input.json

```

Contenidos de `input.json`:

```

{
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
  "DomainName": "example.com",
  "EnableWWWSubdomain": true
}

```

Salida:

```
{
  "CustomDomains": [
    {
      "CertificateValidationRecords": [
        {
          "Name": "_70d3f50a94f7c72dc28784cf55db2f6b.example.com",
          "Status": "PENDING_VALIDATION",
          "Type": "CNAME",
          "Value": "_1270c137383c6307b6832db02504c4b0.bsgbmzkfwj.acm-
validations.aws."
        },
        {
          "Name": "_287870d3f50a94f7c72dc4cf55db2f6b.www.example.com",
          "Status": "PENDING_VALIDATION",
          "Type": "CNAME",
          "Value": "_832db01270c137383c6307b62504c4b0.mzkbsgbfwj.acm-
validations.aws."
        }
      ],
      "DomainName": "example.com",
      "EnableWWWSubdomain": true,
      "Status": "PENDING_CERTIFICATE_DNS_VALIDATION"
    },
    {
      "CertificateValidationRecords": [
        {
          "Name": "_a94f784c70d3f507c72dc28f55db2f6b.deals.example.com",
          "Status": "SUCCESS",
          "Type": "CNAME",
          "Value": "_2db02504c1270c137383c6307b6834b0.bsgbmzkfwj.acm-
validations.aws."
        }
      ],
      "DomainName": "deals.example.com",
      "EnableWWWSubdomain": false,
      "Status": "ACTIVE"
    }
  ],
  "DNSTarget": "psbqam834h.us-east-1.awsapprunner.com",
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

- Para API obtener más información, consulte [DescribeCustomDomains](#) la Referencia de AWS CLI comandos.

## describe-service

En el siguiente ejemplo de código, se muestra cómo usar `describe-service`.

### AWS CLI

Para describir un servicio

En el siguiente `describe-service` ejemplo, se obtiene una descripción de un servicio de App Runner.

```
aws apprunner describe-service \  
--cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa"  
}
```

Salida:

```
{  
  "Service": {  
    "CreatedAt": "2020-11-20T19:05:25Z",  
    "UpdatedAt": "2020-11-20T19:05:25Z",  
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",  
    "ServiceName": "python-app",  
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",  
    "SourceConfiguration": {  
      "AuthenticationConfiguration": {  
        "ConnectionArn": "arn:aws:apprunner:us-  
east-1:123456789012:connection/my-github-connection/  
e7656250f67242d7819feade6800f59e"  
      }  
    }  
  }  
}
```

```

    },
    "AutoDeploymentsEnabled": true,
    "CodeRepository": {
      "CodeConfiguration": {
        "CodeConfigurationValues": {
          "BuildCommand": "pip install -r requirements.txt",
          "Port": "8080",
          "Runtime": "PYTHON_3",
          "RuntimeEnvironmentVariables": [
            {
              "NAME": "Jane"
            }
          ],
          "StartCommand": "python server.py"
        },
        "ConfigurationSource": "Api"
      },
      "RepositoryUrl": "https://github.com/my-account/python-hello",
      "SourceCodeVersion": {
        "Type": "BRANCH",
        "Value": "main"
      }
    }
  },
  "Status": "RUNNING",
  "InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "3 GB"
  }
}
}

```

- Para API obtener más información, consulte [DescribeService](#) la Referencia de AWS CLI comandos.

## disassociate-custom-domain

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-custom-domain`.

### AWS CLI

Para desasociar un nombre de dominio de un servicio

En el siguiente `disassociate-custom-domain` ejemplo, se desvincula el dominio `example.com` de un servicio de App Runner. La llamada también disocia el subdominio `www.example.com` que estaba asociado al dominio raíz.

```
aws apprunner disassociate-custom-domain \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
  "DomainName": "example.com"  
}
```

Salida:

```
{  
  "CustomDomain": {  
    "CertificateValidationRecords": [  
      {  
        "Name": "_70d3f50a94f7c72dc28784cf55db2f6b.example.com",  
        "Status": "PENDING_VALIDATION",  
        "Type": "CNAME",  
        "Value": "_1270c137383c6307b6832db02504c4b0.bsgbmzkfwj.acm-  
validations.aws."  
      },  
      {  
        "Name": "_287870d3f50a94f7c72dc4cf55db2f6b.www.example.com",  
        "Status": "PENDING_VALIDATION",  
        "Type": "CNAME",  
        "Value": "_832db01270c137383c6307b62504c4b0.mzkbsgbfwj.acm-  
validations.aws."  
      }  
    ],  
    "DomainName": "example.com",  
    "EnableWWWSubdomain": true,  
    "Status": "DELETING"  
  },  
  "DNSTarget": "psbqam834h.us-east-1.awsapprunner.com",  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa"
```



```
}
```

- Para API obtener más información, consulte la Referencia [DisassociateCustomDomain](#) de AWS CLI comandos.

## list-auto-scaling-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-auto-scaling-configurations`.

### AWS CLI

Para obtener una lista paginada de las configuraciones de escalado automático de App Runner

En el siguiente `list-auto-scaling-configurations` ejemplo, se enumeran todas las configuraciones de escalado automático de App Runner de tu AWS cuenta.

En cada respuesta se enumeran hasta cinco configuraciones de escalado automático.

`AutoScalingConfigurationName` y `LatestOnly` no están especificadas. Sus valores predeterminados hacen que aparezca la última revisión de todas las configuraciones activas.

En este ejemplo, la respuesta incluye dos resultados y no hay otros adicionales, por lo que no `NextToken` se devuelve ningún resultado.

```
aws apprunner list-auto-scaling-configurations \  
--cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "MaxResults": 5  
}
```

Salida:

```
{  
  "AutoScalingConfigurationSummaryList": [  
    {  
      "AutoScalingConfigurationArn": "arn:aws:apprunner:us-  
east-1:123456789012:autoscalingconfiguration/high-availability/2/  
e76562f50d78042e819fead0f59672e6",
```

```
        "AutoScalingConfigurationName": "high-availability",
        "AutoScalingConfigurationRevision": 2
    },
    {
        "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/low-
cost/1/50d7804e7656fead0f59672e62f2e819",
        "AutoScalingConfigurationName": "low-cost",
        "AutoScalingConfigurationRevision": 1
    }
]
}
```

- Para API obtener más información, consulte [ListAutoScalingConfigurations](#) la Referencia de AWS CLI comandos.

## list-connections

En el siguiente ejemplo de código, se muestra cómo usar `list-connections`.

### AWS CLI

Ejemplo 1: Para enumerar todas las conexiones

En el siguiente `list-connections` ejemplo, se enumeran todas las conexiones de App Runner de la AWS cuenta.

```
aws apprunner list-connections
```

Salida:

```
{
  "ConnectionSummaryList": [
    {
      "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/
my-github-connection",
      "ConnectionName": "my-github-connection",
      "Status": "AVAILABLE",
      "CreatedAt": "2020-11-03T00:32:51Z",
      "ProviderType": "GITHUB"
    },
  ],
}
```

```

    {
      "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/
my-github-org-connection",
      "ConnectionName": "my-github-org-connection",
      "Status": "AVAILABLE",
      "CreatedAt": "2020-11-03T02:54:17Z",
      "ProviderType": "GITHUB"
    }
  ]
}

```

Ejemplo 2: Para enumerar una conexión por nombre

En el siguiente `list-connections` ejemplo, se muestra una conexión por su nombre.

```

aws apprunner list-connections \
  --cli-input-json file://input.json

```

Contenidos de `input.json`:

```

{
  "ConnectionName": "my-github-org-connection"
}

```

Salida:

```

{
  "ConnectionSummaryList": [
    {
      "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/
my-github-org-connection",
      "ConnectionName": "my-github-org-connection",
      "Status": "AVAILABLE",
      "CreatedAt": "2020-11-03T02:54:17Z",
      "ProviderType": "GITHUB"
    }
  ]
}

```

- Para API obtener más información, consulte [ListConnections](#) la Referencia de AWS CLI comandos.

## list-operations

En el siguiente ejemplo de código, se muestra cómo usar `list-operations`.

### AWS CLI

Para enumerar las operaciones que se produjeron en un servicio

En el siguiente `list-operations` ejemplo, se enumeran todas las operaciones que se han producido en un servicio de App Runner hasta el momento. En este ejemplo, el servicio es nuevo y solo se `CREATE_SERVICE` ha realizado una operación de este tipo.

```
aws apprunner list-operations \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa"  
}
```

Salida:

```
{  
  "OperationSummaryList": [  
    {  
      "EndedAt": 1606156217,  
      "Id": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",  
      "StartedAt": 1606156014,  
      "Status": "SUCCEEDED",  
      "TargetArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
      "Type": "CREATE_SERVICE",  
      "UpdatedAt": 1606156217  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListOperations](#) la Referencia de AWS CLI comandos.



```

        "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/golang-
container-app/ab8f94cfe29a460fb8760afd2ee87555",
        "ServiceId": "ab8f94cfe29a460fb8760afd2ee87555",
        "ServiceName": "golang-container-app",
        "ServiceUrl": "e2m8rrrx33.us-east-1.awsapprunner.com",
        "Status": "RUNNING"
    }
]
}

```

- Para API obtener más información, consulte [ListServices](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para ver una lista de las etiquetas asociadas a un servicio de App Runner

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas asociadas a un servicio de App Runner.

```

aws apprunner list-tags-for-resource \
  --cli-input-json file://input.json

```

Contenidos de `input.json`:

```

{
  "ResourceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}

```

Salida:

```

{
  "Tags": [
    {
      "Key": "Department",
      "Value": "Retail"
    },
  ],
}

```

```
{
  "Key": "CustomerId",
  "Value": "56439872357912"
}
]
```

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## pause-service

En el siguiente ejemplo de código, se muestra cómo usar `pause-service`.

### AWS CLI

Para pausar un servicio

En el siguiente `pause-service` ejemplo, se pausa un servicio de App Runner.

```
aws apprunner pause-service \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

Salida:

```
{
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
  "Service": {
    "CreatedAt": "2020-11-20T19:05:25Z",
    "UpdatedAt": "2020-11-23T12:41:37Z",
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
    "ServiceName": "python-app",
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
```

```

    "SourceConfiguration": {
      "AuthenticationConfiguration": {
        "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
      },
      "AutoDeploymentsEnabled": true,
      "CodeRepository": {
        "CodeConfiguration": {
          "CodeConfigurationValues": {
            "BuildCommand": "pip install -r requirements.txt",
            "Port": "8080",
            "Runtime": "PYTHON_3",
            "RuntimeEnvironmentVariables": [
              {
                "NAME": "Jane"
              }
            ],
            "StartCommand": "python server.py"
          },
          "ConfigurationSource": "Api"
        },
        "RepositoryUrl": "https://github.com/my-account/python-hello",
        "SourceCodeVersion": {
          "Type": "BRANCH",
          "Value": "main"
        }
      }
    },
    "Status": "OPERATION_IN_PROGRESS",
    "InstanceConfiguration": {
      "CPU": "1 vCPU",
      "Memory": "3 GB"
    }
  }
}

```

- Para API obtener más información, consulte [PauseService](#) la Referencia de AWS CLI comandos.

## resume-service

En el siguiente ejemplo de código, se muestra cómo usar `resume-service`.



## AWS CLI

Para reanudar un servicio

En el siguiente `resume-service` ejemplo, se reanuda un servicio de App Runner.

```
aws apprunner resume-service \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa"  
}
```

Salida:

```
{  
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",  
  "Service": {  
    "CreatedAt": "2020-11-20T19:05:25Z",  
    "UpdatedAt": "2020-11-23T12:41:37Z",  
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",  
    "ServiceName": "python-app",  
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",  
    "SourceConfiguration": {  
      "AuthenticationConfiguration": {  
        "ConnectionArn": "arn:aws:apprunner:us-  
east-1:123456789012:connection/my-github-connection/  
e7656250f67242d7819feade6800f59e"  
      },  
      "AutoDeploymentsEnabled": true,  
      "CodeRepository": {  
        "CodeConfiguration": {  
          "CodeConfigurationValues": {  
            "BuildCommand": "pip install -r requirements.txt",  
            "Port": "8080",  
            "Runtime": "PYTHON_3",  
            "RuntimeEnvironmentVariables": [  
              {
```

```

        "NAME": "Jane"
      }
    ],
    "StartCommand": "python server.py"
  },
  "ConfigurationSource": "Api"
},
"RepositoryUrl": "https://github.com/my-account/python-hello",
"SourceCodeVersion": {
  "Type": "BRANCH",
  "Value": "main"
}
}
},
"Status": "OPERATION_IN_PROGRESS",
"InstanceConfiguration": {
  "CPU": "1 vCPU",
  "Memory": "3 GB"
}
}
}
}

```

- Para API obtener más información, consulte [ResumeService](#) la Referencia de AWS CLI comandos.

## start-deployment

En el siguiente ejemplo de código, se muestra cómo usar `start-deployment`.

### AWS CLI

Para iniciar un despliegue manual

En el siguiente `start-deployment` ejemplo, se realiza una implementación manual en un servicio de App Runner.

```
aws apprunner start-deployment \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
```

```
"ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

Salida:

```
{
  "OperationId": "853a7d5b-fc9f-4730-831b-fd8037ab832a"
}
```

- Para API obtener más información, consulte [StartDeployment](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir etiquetas a un servicio de App Runner

En el siguiente `tag-resource` ejemplo, se agregan dos etiquetas a un servicio de App Runner.

```
aws apprunner tag-resource \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "ResourceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
  "Tags": [
    {
      "Key": "Department",
      "Value": "Retail"
    },
    {
      "Key": "CustomerId",
      "Value": "56439872357912"
    }
  ]
}
```

```
}
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar etiquetas de un servicio de App Runner

En el siguiente `untag-resource` ejemplo, se eliminan dos etiquetas de un servicio de App Runner.

```
aws apprunner untag-resource \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "ResourceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
  "TagKeys": [  
    "Department",  
    "CustomerId"  
  ]  
}
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-service

En el siguiente ejemplo de código, se muestra cómo usar `update-service`.

## AWS CLI

Para actualizar el tamaño de la memoria

En el siguiente `update-service` ejemplo, se actualiza el tamaño de memoria de las instancias (unidades de escalado) de un servicio de App Runner a 2048 MiB.

Cuando la llamada se realiza correctamente, App Runner inicia un proceso de actualización asíncrono. La `Service` estructura que devuelve la llamada refleja el nuevo valor de memoria que aplica esta llamada.

```
aws apprunner update-service \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
  "InstanceConfiguration": {  
    "Memory": "4 GB"  
  }  
}
```

Salida:

```
{  
  "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",  
  "Service": {  
    "CreatedAt": "2020-11-20T19:05:25Z",  
    "UpdatedAt": "2020-11-23T12:41:37Z",  
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-  
app/8fe1e10304f84fd2b0df550fe98a71fa",  
    "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",  
    "ServiceName": "python-app",  
    "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",  
    "SourceConfiguration": {  
      "AuthenticationConfiguration": {  
        "ConnectionArn": "arn:aws:apprunner:us-  
east-1:123456789012:connection/my-github-connection/  
e7656250f67242d7819feade6800f59e"  
      },  
    },  
  },  
}
```

```
    "AutoDeploymentsEnabled": true,
    "CodeRepository": {
      "CodeConfiguration": {
        "CodeConfigurationValues": {
          "BuildCommand": "pip install -r requirements.txt",
          "Port": "8080",
          "Runtime": "PYTHON_3",
          "RuntimeEnvironmentVariables": [
            {
              "NAME": "Jane"
            }
          ],
          "StartCommand": "python server.py"
        },
        "ConfigurationSource": "Api"
      },
      "RepositoryUrl": "https://github.com/my-account/python-hello",
      "SourceCodeVersion": {
        "Type": "BRANCH",
        "Value": "main"
      }
    }
  },
  "Status": "OPERATION_IN_PROGRESS",
  "InstanceConfiguration": {
    "CPU": "1 vCPU",
    "Memory": "4 GB"
  }
}
```

- Para API obtener más información, consulte [UpdateService](#) la Referencia de AWS CLI comandos.

## AWS AppConfig ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS AppConfig.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **create-application**

En el siguiente ejemplo de código, se muestra cómo usar `create-application`.

#### AWS CLI

Para crear una aplicación

El siguiente `create-application` ejemplo crea una aplicación en AWS AppConfig.

```
aws appconfig create-application \  
  --name "example-application" \  
  --description "An application used for creating an example."
```

Salida:

```
{  
  "Description": "An application used for creating an example.",  
  "Id": "339ohji",  
  "Name": "example-application"  
}
```

Para obtener más información, consulte el [paso 1: Crear una AWS AppConfig aplicación](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [CreateApplication](#) la Referencia de AWS CLI comandos.

### **create-configuration-profile**

En el siguiente ejemplo de código, se muestra cómo usar `create-configuration-profile`.

## AWS CLI

Para crear un perfil de configuración

En el siguiente `create-configuration-profile` ejemplo, se crea un perfil de configuración mediante una configuración almacenada en Parameter Store, una función de Systems Manager.

```
aws appconfig create-configuration-profile \  
  --application-id "339ohji" \  
  --name "Example-Configuration-Profile" \  
  --location-uri "ssm-parameter://Example-Parameter" \  
  --retrieval-role-arn "arn:aws:iam::111122223333:role/Example-App-Config-Role"
```

Salida:

```
{  
  "ApplicationId": "339ohji",  
  "Description": null,  
  "Id": "ur8hx2f",  
  "LocationUri": "ssm-parameter://Example-Parameter",  
  "Name": "Example-Configuration-Profile",  
  "RetrievalRoleArn": "arn:aws:iam::111122223333:role/Example-App-Config-Role",  
  "Type": null,  
  "Validators": null  
}
```

Para obtener más información, consulte el [paso 3: Crear una configuración y un perfil de configuración](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [CreateConfigurationProfile](#) la Referencia de AWS CLI comandos.

## **create-environment**

En el siguiente ejemplo de código, se muestra cómo usar `create-environment`.

## AWS CLI

Para crear un entorno

El siguiente `create-environment` ejemplo crea un AWS AppConfig entorno denominado `Example-Environment` mediante la aplicación que creó con `create-application`.



```
aws appconfig create-environment \  
  --application-id "339ohji" \  
  --name "Example-Environment"
```

Salida:

```
{  
  "ApplicationId": "339ohji",  
  "Description": null,  
  "Id": "54j1r29",  
  "Monitors": null,  
  "Name": "Example-Environment",  
  "State": "ReadyForDeployment"  
}
```

Para obtener más información, consulte el [paso 2: Creación de un entorno](#) en la Guía del usuario.AWS AppConfig

- Para API obtener más información, consulte [CreateEnvironment](#)la Referencia de AWS CLI comandos.

## create-extension-association

En el siguiente ejemplo de código, se muestra cómo usar create-extension-association.

AWS CLI

Para crear una asociación de extensiones

El siguiente create-extension-association ejemplo crea una nueva asociación de extensiones en AWS AppConfig.

```
aws appconfig create-extension-association \  
  --region us-west-2 \  
  --extension-identifier S3-backup-extension \  
  --resource-identifier "arn:aws:appconfig:us-west-2:123456789012:application/  
Finance" \  
  --parameters S3bucket=FinanceConfigurationBackup
```

Salida:

```
{
  "Id": "a1b2c3d4",
  "ExtensionArn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-backup-extension/1",
  "ResourceArn": "arn:aws:appconfig:us-west-2:123456789012:application/Finance",
  "Parameters": {
    "S3bucket": "FinanceConfigurationBackup"
  },
  "ExtensionVersionNumber": 1
}
```

Para obtener más información, consulte [Trabajar con AWS AppConfig extensiones](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [CreateExtensionAssociation](#) la Referencia de AWS CLI comandos.

## create-extension

En el siguiente ejemplo de código, se muestra cómo usar create-extension.

### AWS CLI

Para crear una extensión

En el create-extension ejemplo siguiente se crea una nueva extensión en AWS AppConfig.

```
aws appconfig create-extension \
  --region us-west-2 \
  --name S3-backup-extension \
  --
actions PRE_CREATE_HOSTED_CONFIGURATION_VERSION=[{Name=S3backup,Uri=arn:aws:lambda:us-west-2:123456789012:function:s3backupfunction,RoleArn=arn:aws:iam::123456789012:role/appconfigextensionrole}] \
  --parameters S3bucket={Required=true}
```

Salida:

```
{
  "Id": "1A2B3C4D",
  "Name": "S3-backup-extension",
  "VersionNumber": 1,
```

```

    "Arn": "arn:aws:appconfig:us-west-2:123456789012:extension/1A2B3C4D/1",
    "Actions": {
      "PRE_CREATE_HOSTED_CONFIGURATION_VERSION": [
        {
          "Name": "S3backup",
          "Uri": "arn:aws:lambda:us-
west-2:123456789012:function:s3backupfunction",
          "RoleArn": "arn:aws:iam::123456789012:role/appconfigextensionrole"
        }
      ]
    },
    "Parameters": {
      "S3bucket": {
        "Required": true
      }
    }
  }
}

```

Para obtener más información, consulte [Trabajar con AWS AppConfig extensiones](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [CreateExtension](#) la Referencia de AWS CLI comandos.

## create-hosted-configuration-version

En el siguiente ejemplo de código, se muestra cómo usar `create-hosted-configuration-version`.

### AWS CLI

Para crear una versión de configuración alojada

El siguiente `create-hosted-configuration-version` ejemplo crea una nueva configuración en el almacén de configuración AWS AppConfig hospedado. El contenido de la configuración debe convertirse primero a base64.

```

aws appconfig create-hosted-configuration-version \
  --application-id "339ohji" \
  --configuration-profile-id "ur8hx2f" \
  --
content eyAiTmFtZSI6ICJFeGFtcGxlQXBwbGljYXRpb24iLCAiSWQiOiBFcGFtcGxlSUQsICJSYW5rIjogNyB9
\

```

```
--content-type "application/json" \  
configuration_version_output_file
```

Contenidos de configuration\_version\_output\_file:

```
{ "Name": "ExampleApplication", "Id": ExampleID, "Rank": 7 }
```

Salida:

```
{  
  "ApplicationId": "339ohji",  
  "ConfigurationProfileId": "ur8hx2f",  
  "VersionNumber": "1",  
  "ContentType": "application/json"  
}
```

Para obtener más información, consulte [Acerca del almacén de configuración AWS AppConfig hospedado](#) en la Guía del usuario de AWS AppConfig.

- Para API obtener más información, consulte [CreateHostedConfigurationVersion](#) la Referencia de AWS CLI comandos.

## delete-application

En el siguiente ejemplo de código, se muestra cómo usar delete-application.

AWS CLI

Eliminación de una aplicación

En el siguiente delete-application ejemplo, se elimina la aplicación especificada.

```
aws appconfig delete-application \  
--application-id 339ohji
```

Este comando no genera ninguna salida.

Para obtener más información, consulte el [paso 1: Crear una AWS AppConfig aplicación](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [DeleteApplication](#) la Referencia de AWS CLI comandos.

## delete-configuration-profile

En el siguiente ejemplo de código, se muestra cómo usar `delete-configuration-profile`.

### AWS CLI

Para eliminar un perfil de configuración

En el siguiente `delete-configuration-profile` ejemplo, se elimina el perfil de configuración especificado.

```
aws appconfig delete-configuration-profile \  
  --application-id 339ohji \  
  --configuration-profile-id ur8hx2f
```

Este comando no genera ninguna salida.

Para obtener más información, consulte el [paso 3: Crear una configuración y un perfil de configuración](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [DeleteConfigurationProfile](#) la Referencia de AWS CLI comandos.

## delete-deployment-strategy

En el siguiente ejemplo de código, se muestra cómo usar `delete-deployment-strategy`.

### AWS CLI

Para eliminar una estrategia de despliegue

En el siguiente `delete-deployment-strategy` ejemplo, se elimina la estrategia de despliegue especificada.

```
aws appconfig delete-deployment-strategy \  
  --deployment-strategy-id 1225qzk
```

Este comando no genera ninguna salida.

Para obtener más información, consulte el [paso 4: Crear una estrategia de despliegue](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [DeleteDeploymentStrategy](#) la Referencia de AWS CLI comandos.

## delete-environment

En el siguiente ejemplo de código, se muestra cómo usar delete-environment.

### AWS CLI

Para eliminar un entorno

En el siguiente delete-environment ejemplo, se elimina el entorno de aplicación especificado.

```
aws appconfig delete-environment \  
  --application-id 339ohji \  
  --environment-id 54j1r29
```

Este comando no genera ninguna salida.

Para obtener más información, consulte el [paso 2: Creación de un entorno](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [DeleteEnvironment](#) la Referencia de AWS CLI comandos.

## delete-extension-association

En el siguiente ejemplo de código, se muestra cómo usar delete-extension-association.

### AWS CLI

Para eliminar una asociación de extensiones

En el siguiente delete-extension-association ejemplo, se elimina una asociación de extensiones de AWS AppConfig.

```
aws appconfig delete-extension-association \  
  --region us-west-2 \  
  --extension-association-id a1b2c3d4
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con AWS AppConfig extensiones](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [DeleteExtensionAssociation](#) la Referencia de AWS CLI comandos.

## **delete-extension**

En el siguiente ejemplo de código, se muestra cómo usar `delete-extension`.

### AWS CLI

Para eliminar una extensión

En el siguiente `delete-extension` ejemplo, se elimina una extensión de AWS AppConfig.

```
aws appconfig delete-extension \  
  --region us-west-2 \  
  --extension-identifier S3-backup-extension
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con AWS AppConfig extensiones](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [DeleteExtension](#) la Referencia de AWS CLI comandos.

## **delete-hosted-configuration-version**

En el siguiente ejemplo de código, se muestra cómo usar `delete-hosted-configuration-version`.

### AWS CLI

Para eliminar una versión de configuración alojada

En el siguiente `delete-hosted-configuration-version` ejemplo, se elimina una versión de configuración alojada en el almacén de configuración AWS AppConfig hospedado.

```
aws appconfig delete-hosted-configuration-version \  
  --application-id 339ohji \  
  --configuration-version-id 339ohji
```

```
--configuration-profile-id ur8hx2f \  
--version-number 1
```

Resultado: este comando no produce ningún resultado.

Para obtener más información, consulte el [paso 3: Crear una configuración y un perfil de configuración](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [DeleteHostedConfigurationVersion](#) la Referencia de AWS CLI comandos.

## get-application

En el siguiente ejemplo de código, se muestra cómo usar `get-application`.

### AWS CLI

Para enumerar los detalles de una aplicación

En el siguiente `get-application` ejemplo, se muestran los detalles de la aplicación especificada.

```
aws appconfig get-application \  
--application-id 339ohji
```

Salida:

```
{  
  "Description": "An application used for creating an example.",  
  "Id": "339ohji",  
  "Name": "example-application"  
}
```

Para obtener más información, consulte [Cómo AWS AppConfig funciona](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [GetApplication](#) la Referencia de AWS CLI comandos.

## get-configuration-profile

En el siguiente ejemplo de código, se muestra cómo usar `get-configuration-profile`.



## AWS CLI

Para recuperar los detalles del perfil de configuración

El siguiente `get-configuration-profile` ejemplo devuelve los detalles del perfil de configuración especificado.

```
aws appconfig get-configuration-profile \  
  --application-id 339ohji \  
  --configuration-profile-id ur8hx2f
```

Salida:

```
{  
  "ApplicationId": "339ohji",  
  "Id": "ur8hx2f",  
  "Name": "Example-Configuration-Profile",  
  "LocationUri": "ssm-parameter://Example-Parameter",  
  "RetrievalRoleArn": "arn:aws:iam::111122223333:role/Example-App-Config-Role"  
}
```

Para obtener más información, consulte el [paso 3: Crear una configuración y un perfil de configuración](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [GetConfigurationProfile](#) la Referencia de AWS CLI comandos.

## get-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-configuration`.

## AWS CLI

Para recuperar los detalles de configuración

El siguiente `get-configuration` ejemplo devuelve los detalles de configuración de la aplicación de ejemplo. En llamadas posteriores a `get-configuration`, utilice el `client-configuration-version` parámetro para actualizar la configuración de la aplicación únicamente si la versión ha cambiado. Si solo se actualiza la configuración cuando la versión ha cambiado, se evitan los cargos excesivos que se generen al llamar a `get-configuration`.

```
aws appconfig get-configuration \  
  --application-id 339ohji \  
  --client-configuration-version 1
```

```
--application "example-application" \  
--environment "Example-Environment" \  
--configuration "Example-Configuration-Profile" \  
--client-id "test-id" \  
configuration-output-file
```

Contenidos de configuration-output-file:

```
{ "Name": "ExampleApplication", "Id": ExampleID, "Rank": 7 }
```

Salida:

```
{  
  "ConfigurationVersion": "1",  
  "ContentType": "application/json"  
}
```

Para obtener más información, consulte el [paso 6: Recibir la configuración](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [GetConfiguration](#) la Referencia de AWS CLI comandos.

## get-deployment-strategy

En el siguiente ejemplo de código, se muestra cómo usar get-deployment-strategy.

AWS CLI

Para recuperar los detalles de una estrategia de despliegue

En el siguiente get-deployment-strategy ejemplo, se enumeran los detalles de la estrategia de despliegue especificada.

```
aws appconfig get-deployment-strategy \  
--deployment-strategy-id 1225qzk
```

Salida:

```
{  
  "Id": "1225qzk",  
  "Name": "Example-Deployment",
```

```
"DeploymentDurationInMinutes": 15,  
"GrowthType": "LINEAR",  
"GrowthFactor": 25.0,  
"FinalBakeTimeInMinutes": 0,  
"ReplicateTo": "SSM_DOCUMENT"  
}
```

Para obtener más información, consulte el [paso 4: Crear una estrategia de despliegue](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [GetDeploymentStrategy](#) la Referencia de AWS CLI comandos.

## get-deployment

En el siguiente ejemplo de código, se muestra cómo usar `get-deployment`.

### AWS CLI

Para recuperar los detalles de la implementación

En el siguiente `get-deployment` ejemplo, se enumeran los detalles de la implementación en la aplicación en el entorno y la implementación especificados.

```
aws appconfig get-deployment \  
  --application-id 339ohji \  
  --environment-id 54j1r29 \  
  --deployment-number 1
```

Salida:

```
{  
  "ApplicationId": "339ohji",  
  "EnvironmentId": "54j1r29",  
  "DeploymentStrategyId": "1225qzk",  
  "ConfigurationProfileId": "ur8hx2f",  
  "DeploymentNumber": 1,  
  "ConfigurationName": "Example-Configuration-Profile",  
  "ConfigurationLocationUri": "ssm-parameter://Example-Parameter",  
  "ConfigurationVersion": "1",  
  "DeploymentDurationInMinutes": 15,  
  "GrowthType": "LINEAR",  
  "GrowthFactor": 25.0,  
}
```

```
"FinalBakeTimeInMinutes": 0,
"State": "COMPLETE",
"EventLog": [
  {
    "EventType": "DEPLOYMENT_COMPLETED",
    "TriggeredBy": "APPCONFIG",
    "Description": "Deployment completed",
    "OccurredAt": "2021-09-17T21:59:03.888000+00:00"
  },
  {
    "EventType": "BAKE_TIME_STARTED",
    "TriggeredBy": "APPCONFIG",
    "Description": "Deployment bake time started",
    "OccurredAt": "2021-09-17T21:58:57.722000+00:00"
  },
  {
    "EventType": "PERCENTAGE_UPDATED",
    "TriggeredBy": "APPCONFIG",
    "Description": "Configuration available to 100.00% of clients",
    "OccurredAt": "2021-09-17T21:55:56.816000+00:00"
  },
  {
    "EventType": "PERCENTAGE_UPDATED",
    "TriggeredBy": "APPCONFIG",
    "Description": "Configuration available to 75.00% of clients",
    "OccurredAt": "2021-09-17T21:52:56.567000+00:00"
  },
  {
    "EventType": "PERCENTAGE_UPDATED",
    "TriggeredBy": "APPCONFIG",
    "Description": "Configuration available to 50.00% of clients",
    "OccurredAt": "2021-09-17T21:49:55.737000+00:00"
  },
  {
    "EventType": "PERCENTAGE_UPDATED",
    "TriggeredBy": "APPCONFIG",
    "Description": "Configuration available to 25.00% of clients",
    "OccurredAt": "2021-09-17T21:46:55.187000+00:00"
  },
  {
    "EventType": "DEPLOYMENT_STARTED",
    "TriggeredBy": "USER",
    "Description": "Deployment started",
    "OccurredAt": "2021-09-17T21:43:54.205000+00:00"
  }
]
```

```
    }
  ],
  "PercentageComplete": 100.0,
  "StartedAt": "2021-09-17T21:43:54.205000+00:00",
  "CompletedAt": "2021-09-17T21:59:03.888000+00:00"
}
```

Para obtener más información, consulte el [paso 5: Implementar una configuración](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [GetDeployment](#) la Referencia de AWS CLI comandos.

## get-environment

En el siguiente ejemplo de código, se muestra cómo usar `get-environment`.

### AWS CLI

Para recuperar los detalles del entorno

El siguiente `get-environment` ejemplo devuelve los detalles y el estado del entorno especificado.

```
aws appconfig get-environment \
  --application-id 339ohji \
  --environment-id 54j1r29
```

Salida:

```
{
  "ApplicationId": "339ohji",
  "Id": "54j1r29",
  "Name": "Example-Environment",
  "State": "ReadyForDeployment"
}
```

Para obtener más información, consulte el [paso 2: Creación de un entorno](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [GetEnvironment](#) la Referencia de AWS CLI comandos.

## get-extension-association

En el siguiente ejemplo de código, se muestra cómo usar `get-extension-association`.

### AWS CLI

Para obtener los detalles de la asociación de extensiones

En el siguiente `get-extension-association` ejemplo, se muestra información sobre una asociación de extensiones.

```
aws appconfig get-extension-association \
  --region us-west-2 \
  --extension-association-id a1b2c3d4
```

Salida:

```
{
  "Id": "a1b2c3d4",
  "ExtensionArn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-backup-extension/1",
  "ResourceArn": "arn:aws:appconfig:us-west-2:123456789012:application/Finance",
  "Parameters": {
    "S3bucket": "FinanceConfigurationBackup"
  },
  "ExtensionVersionNumber": 1
}
```

Para obtener más información, consulte [Trabajar con AWS AppConfig extensiones](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [GetExtensionAssociation](#) la Referencia de AWS CLI comandos.

## get-extension

En el siguiente ejemplo de código, se muestra cómo usar `get-extension`.

### AWS CLI

Para obtener los detalles de la extensión

En el siguiente `get-extension` ejemplo, se muestra información sobre una extensión.

```
aws appconfig get-extension \  
  --region us-west-2 \  
  --extension-identifier S3-backup-extension
```

Salida:

```
{  
  "Id": "1A2B3C4D",  
  "Name": "S3-backup-extension",  
  "VersionNumber": 1,  
  "Arn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-backup-  
extension/1",  
  "Actions": {  
    "PRE_CREATE_HOSTED_CONFIGURATION_VERSION": [  
      {  
        "Name": "S3backup",  
        "Uri": "arn:aws:lambda:us-  
west-2:123456789012:function:S3backupfunction",  
        "RoleArn": "arn:aws:iam::123456789012:role/appconfigextensionrole"  
      }  
    ]  
  },  
  "Parameters": {  
    "S3bucket": {  
      "Required": true  
    }  
  }  
}
```

Para obtener más información, consulte [Trabajar con AWS AppConfig extensiones](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [GetExtension](#) la Referencia de AWS CLI comandos.

## get-hosted-configuration-version

En el siguiente ejemplo de código, se muestra cómo usar `get-hosted-configuration-version`.

## AWS CLI

Para recuperar los detalles de la configuración alojada

En el siguiente `get-hosted-configuration-version` ejemplo, se recuperan los detalles de configuración de la AWS AppConfig configuración alojada.

```
aws appconfig get-hosted-configuration-version \  
  --application-id 339ohji \  
  --configuration-profile-id ur8hx2f \  
  --version-number 1 \  
hosted-configuration-version-output
```

Contenidos de `hosted-configuration-version-output`:

```
{ "Name": "ExampleApplication", "Id": ExampleID, "Rank": 7 }
```

Salida:

```
{  
  "ApplicationId": "339ohji",  
  "ConfigurationProfileId": "ur8hx2f",  
  "VersionNumber": "1",  
  "ContentType": "application/json"  
}
```

Para obtener más información, consulte [Acerca del almacén de configuración AWS AppConfig hospedado](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [GetHostedConfigurationVersion](#) la Referencia de AWS CLI comandos.

## **list-applications**

En el siguiente ejemplo de código, se muestra cómo usar `list-applications`.

## AWS CLI

Para ver una lista de las aplicaciones disponibles

En el siguiente `list-applications` ejemplo, se enumeran las aplicaciones disponibles en su AWS cuenta.



```
aws appconfig list-applications
```

Salida:

```
{
  "Items": [
    {
      "Id": "339ohji",
      "Name": "test-application",
      "Description": "An application used for creating an example."
    },
    {
      "Id": "rwalwu7",
      "Name": "Test-Application"
    }
  ]
}
```

Para obtener más información, consulte el [paso 1: Crear una AWS AppConfig aplicación](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [ListApplications](#) la Referencia de AWS CLI comandos.

## list-configuration-profiles

En el siguiente ejemplo de código, se muestra cómo usar `list-configuration-profiles`.

AWS CLI

Para ver una lista de los perfiles de configuración disponibles

El siguiente `list-configuration-profiles` ejemplo muestra los perfiles de configuración disponibles para la aplicación especificada.

```
aws appconfig list-configuration-profiles \
  --application-id 339ohji
```

Salida:

```
{
```

```
"Items": [  
  {  
    "ApplicationId": "339ohji",  
    "Id": "ur8hx2f",  
    "Name": "Example-Configuration-Profile",  
    "LocationUri": "ssm-parameter://Example-Parameter"  
  }  
]  
}
```

Para obtener más información, consulte el [paso 3: Crear una configuración y un perfil de configuración](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [ListConfigurationProfiles](#) la Referencia de AWS CLI comandos.

## list-deployment-strategies

En el siguiente ejemplo de código, se muestra cómo usar `list-deployment-strategies`.

AWS CLI

Para enumerar las estrategias de despliegue disponibles

En el siguiente `list-deployment-strategies` ejemplo, se enumeran las estrategias de despliegue disponibles en su AWS cuenta.

```
aws appconfig list-deployment-strategies
```

Salida:

```
{  
  "Items": [  
    {  
      "Id": "1225qzk",  
      "Name": "Example-Deployment",  
      "DeploymentDurationInMinutes": 15,  
      "GrowthType": "LINEAR",  
      "GrowthFactor": 25.0,  
      "FinalBakeTimeInMinutes": 0,  
      "ReplicateTo": "SSM_DOCUMENT"  
    }  
  ]  
}
```

```

    },
    {
      "Id": "AppConfig.AllAtOnce",
      "Name": "AppConfig.AllAtOnce",
      "Description": "Quick",
      "DeploymentDurationInMinutes": 0,
      "GrowthType": "LINEAR",
      "GrowthFactor": 100.0,
      "FinalBakeTimeInMinutes": 10,
      "ReplicateTo": "NONE"
    },
    {
      "Id": "AppConfig.Linear50PercentEvery30Seconds",
      "Name": "AppConfig.Linear50PercentEvery30Seconds",
      "Description": "Test/Demo",
      "DeploymentDurationInMinutes": 1,
      "GrowthType": "LINEAR",
      "GrowthFactor": 50.0,
      "FinalBakeTimeInMinutes": 1,
      "ReplicateTo": "NONE"
    },
    {
      "Id": "AppConfig.Canary10Percent20Minutes",
      "Name": "AppConfig.Canary10Percent20Minutes",
      "Description": "AWS Recommended",
      "DeploymentDurationInMinutes": 20,
      "GrowthType": "EXPONENTIAL",
      "GrowthFactor": 10.0,
      "FinalBakeTimeInMinutes": 10,
      "ReplicateTo": "NONE"
    }
  ]
}

```

Para obtener más información, consulte el [paso 4: Crear una estrategia de despliegue](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [ListDeploymentStrategies](#) la Referencia de AWS CLI comandos.

## list-deployments

En el siguiente ejemplo de código, se muestra cómo usar `list-deployments`.

## AWS CLI

Para ver una lista de las implementaciones disponibles

En el siguiente `list-deployments` ejemplo, se enumeran las implementaciones disponibles en su AWS cuenta para la aplicación y el entorno especificados.

```
aws appconfig list-deployments \  
  --application-id 339ohji \  
  --environment-id 54j1r29
```

Salida:

```
{  
  "Items": [  
    {  
      "DeploymentNumber": 1,  
      "ConfigurationName": "Example-Configuration-Profile",  
      "ConfigurationVersion": "1",  
      "DeploymentDurationInMinutes": 15,  
      "GrowthType": "LINEAR",  
      "GrowthFactor": 25.0,  
      "FinalBakeTimeInMinutes": 0,  
      "State": "COMPLETE",  
      "PercentageComplete": 100.0,  
      "StartedAt": "2021-09-17T21:43:54.205000+00:00",  
      "CompletedAt": "2021-09-17T21:59:03.888000+00:00"  
    }  
  ]  
}
```

Para obtener más información, consulte el [paso 5: Implementar una configuración](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [ListDeployments](#) la Referencia de AWS CLI comandos.

## **list-environments**

En el siguiente ejemplo de código, se muestra cómo usar `list-environments`.

## AWS CLI

Para ver una lista de los entornos disponibles

En el siguiente `list-environments` ejemplo, se enumeran los entornos disponibles en su AWS cuenta para la aplicación especificada.

```
aws appconfig list-environments \
  --application-id 339ohji
```

Salida:

```
{
  "Items": [
    {
      "ApplicationId": "339ohji",
      "Id": "54j1r29",
      "Name": "Example-Environment",
      "State": "ReadyForDeployment"
    }
  ]
}
```

Para obtener más información, consulte el [paso 2: Creación de un entorno](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [ListEnvironments](#) la Referencia de AWS CLI comandos.

## **list-extension-associations**

En el siguiente ejemplo de código, se muestra cómo usar `list-extension-associations`.

## AWS CLI

Para ver una lista de todas las asociaciones de AWS AppConfig extensiones de una AWS región en tu AWS cuenta

En el siguiente `list-extension-associations` ejemplo, se enumeran todas las asociaciones de AWS AppConfig extensiones de la AWS cuenta corriente de una AWS región específica.

```
aws appconfig list-extension-associations \  
--region us-west-2
```

Salida:

```
{  
  "Items": [  
    {  
      "Id": "a1b2c3d4",  
      "ExtensionArn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-  
backup-extension/1",  
      "ResourceArn": "arn:aws:appconfig:us-west-2:123456789012:application/  
Finance"  
    }  
  ]  
}
```

Para obtener más información, consulte [Trabajar con AWS AppConfig extensiones](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [ListExtensionAssociations](#) la Referencia de AWS CLI comandos.

## list-extensions

En el siguiente ejemplo de código, se muestra cómo usar `list-extensions`.

### AWS CLI

Para ver una lista AWS AppConfig de todas las extensiones de tu AWS cuenta para una AWS región

En el siguiente `list-extensions` ejemplo, se muestran todas las AWS AppConfig extensiones de la AWS cuenta corriente de una AWS región específica. El comando devuelve extensiones personalizadas y AWS creadas.

```
aws appconfig list-extensions \  
--region us-west-2
```

Salida:

```

{
  "Items": [
    {
      "Id": "1A2B3C4D",
      "Name": "S3-backup-extension",
      "VersionNumber": 1,
      "Arn": "arn:aws:appconfig:us-west-2:123456789012:extension/1A2B3C4D/1"
    },
    {
      "Id": "AWS.AppConfig.FeatureFlags",
      "Name": "AppConfig Feature Flags Helper",
      "VersionNumber": 1,
      "Arn": "arn:aws:appconfig:us-west-2::extension/
AWS.AppConfig.FeatureFlags/1",
      "Description": "Validates AppConfig feature flag data automatically
against a JSON schema that includes structure and constraints. Also transforms
feature flag data prior to sending to the client. This extension is automatically
associated to configuration profiles with type \"AWS.AppConfig.FeatureFlags\"."
    },
    {
      "Id": "AWS.AppConfig.JiraIntegration",
      "Name": "AppConfig integration with Atlassian Jira",
      "VersionNumber": 1,
      "Arn": "arn:aws:appconfig:us-west-2::extension/
AWS.AppConfig.JiraIntegration/1",
      "Description": "Exports feature flag data from AWS AppConfig into
Jira. The lifecycle of each feature flag in AppConfig is tracked in Jira as an
individual issue. Customers can see in Jira when flags are updated, turned on or
off. Works in conjunction with the AppConfig app in the Atlassian Marketplace and
is automatically associated to configuration profiles configured within that app."
    },
    {
      "Id": "AWS.AppConfig.DeploymentNotificationsToEventBridge",
      "Name": "AppConfig deployment events to Amazon EventBridge",
      "VersionNumber": 1,
      "Arn": "arn:aws:appconfig:us-west-2::extension/
AWS.AppConfig.DeploymentNotificationsToEventBridge/1",
      "Description": "Sends events to Amazon EventBridge when a deployment
of configuration data in AppConfig is started, completed, or rolled back. Can
be associated to the following resources in AppConfig: Application, Environment,
Configuration Profile."
    },
    {

```

```

        "Id": "AWS.AppConfig.DeploymentNotificationsToSqs",
        "Name": "AppConfig deployment events to Amazon SQS",
        "VersionNumber": 1,
        "Arn": "arn:aws:appconfig:us-west-2::extension/
AWS.AppConfig.DeploymentNotificationsToSqs/1",
        "Description": "Sends messages to the configured Amazon SQS queue when
a deployment of configuration data in AppConfig is started, completed, or rolled
back. Can be associated to the following resources in AppConfig: Application,
Environment, Configuration Profile."
    },
    {
        "Id": "AWS.AppConfig.DeploymentNotificationsToSns",
        "Name": "AppConfig deployment events to Amazon SNS",
        "VersionNumber": 1,
        "Description": "Sends events to the configured Amazon SNS topic when
a deployment of configuration data in AppConfig is started, completed, or rolled
back. Can be associated to the following resources in AppConfig: Application,
Environment, Configuration Profile."
    }
]
}

```

Para obtener más información, consulte [Trabajar con AWS AppConfig extensiones](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [ListExtensions](#) la Referencia de AWS CLI comandos.

## list-hosted-configuration-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-hosted-configuration-versions`.

### AWS CLI

Para ver una lista de las versiones de configuración alojada disponibles

En el siguiente `list-hosted-configuration-versions` ejemplo, se enumeran las versiones de configuración alojadas en el almacén de configuración AWS AppConfig hospedado para la aplicación y el perfil de configuración especificados.

```
aws appconfig list-hosted-configuration-versions \
```



```
--application-id 339ohji \  
--configuration-profile-id ur8hx2f
```

Salida:

```
{  
  "Items": [  
    {  
      "ApplicationId": "339ohji",  
      "ConfigurationProfileId": "ur8hx2f",  
      "VersionNumber": 1,  
      "ContentType": "application/json"  
    }  
  ]  
}
```

Para obtener más información, consulte [Acerca del almacén de configuración AWS AppConfig hospedado](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [ListHostedConfigurationVersions](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

AWS CLI

Para enumerar las etiquetas de una aplicación

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas de una aplicación específica.

```
aws appconfig list-tags-for-resource \  
--resource-arn arn:aws:appconfig:us-east-1:682428703967:application/339ohji
```

Salida:

```
{  
  "Tags": {  
    "group1": "1"  }  
}
```

```
}  
}
```

Para obtener más información, consulte el [paso 1: Crear una AWS AppConfig aplicación](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## start-deployment

En el siguiente ejemplo de código, se muestra cómo usar `start-deployment`.

### AWS CLI

Para iniciar un despliegue de configuración

En el siguiente `start-deployment` ejemplo, se inicia una implementación en la aplicación mediante el entorno, la estrategia de implementación y el perfil de configuración especificados.

```
aws AppConfig start-deployment \  
  --application-id 339ohji \  
  --environment-id 54j1r29 \  
  --deployment-strategy-id 1225qzk \  
  --configuration-profile-id ur8hx2f \  
  --configuration-version 1
```

Salida:

```
{  
  "ApplicationId": "339ohji",  
  "EnvironmentId": "54j1r29",  
  "DeploymentStrategyId": "1225qzk",  
  "ConfigurationProfileId": "ur8hx2f",  
  "DeploymentNumber": 1,  
  "ConfigurationName": "Example-Configuration-Profile",  
  "ConfigurationLocationUri": "ssm-parameter://Example-Parameter",  
  "ConfigurationVersion": "1",  
  "DeploymentDurationInMinutes": 15,  
  "GrowthType": "LINEAR",  
  "GrowthFactor": 25.0,  
  "FinalBakeTimeInMinutes": 0,
```

```

    "State": "DEPLOYING",
    "EventLog": [
      {
        "EventType": "DEPLOYMENT_STARTED",
        "TriggeredBy": "USER",
        "Description": "Deployment started",
        "OccurredAt": "2021-09-17T21:43:54.205000+00:00"
      }
    ],
    "PercentageComplete": 0.0,
    "StartedAt": "2021-09-17T21:43:54.205000+00:00"
  }
}

```

Para obtener más información, consulte el [paso 5: Implementar una configuración](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [StartDeployment](#) la Referencia de AWS CLI comandos.

## stop-deployment

En el siguiente ejemplo de código, se muestra cómo usar stop-deployment.

### AWS CLI

Para detener el despliegue de la configuración

El siguiente stop-deployment ejemplo detiene el despliegue de una configuración de aplicación en el entorno especificado.

```

aws appconfig stop-deployment \
  --application-id 339ohji \
  --environment-id 54j1r29 \
  --deployment-number 2

```

Salida:

```

{
  "DeploymentNumber": 0,
  "DeploymentDurationInMinutes": 0,
  "GrowthFactor": 0.0,
  "FinalBakeTimeInMinutes": 0,

```

```
"PercentageComplete": 0.0  
}
```

Para obtener más información, consulte el [paso 5: Implementar una configuración](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [StopDeployment](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar una aplicación

El siguiente `tag-resource` ejemplo etiqueta un recurso de aplicación.

```
aws appconfig tag-resource \  
  --resource-arn arn:aws:appconfig:us-east-1:682428703967:application/339ohji \  
  --tags '{"group1" : "1"}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte el [paso 1: Crear una AWS AppConfig aplicación](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar una etiqueta de una aplicación

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta `group1` de la aplicación especificada.

```
aws appconfig untag-resource \  
  --resource-arn arn:aws:appconfig:us-east-1:111122223333:application/339ohji \  
  --tag-keys '["group1"]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte el [paso 1: Crear una AWS AppConfig aplicación](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-application

En el siguiente ejemplo de código, se muestra cómo usar `update-application`.

### AWS CLI

Para actualizar una aplicación

En el siguiente `update-application` ejemplo, se actualiza el nombre de la aplicación especificada.

```
aws appconfig update-application \  
  --application-id 339ohji \  
  --name "Example-Application"
```

Salida:

```
{  
  "Id": "339ohji",  
  "Name": "Example-Application",  
  "Description": "An application used for creating an example."  
}
```

Para obtener más información, consulte el [paso 1: Crear una AWS AppConfig aplicación](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [UpdateApplication](#) la Referencia de AWS CLI comandos.

## update-configuration-profile

En el siguiente ejemplo de código, se muestra cómo usar `update-configuration-profile`.

### AWS CLI

Para actualizar un perfil de configuración

El siguiente `update-configuration-profile` ejemplo actualiza la descripción del perfil de configuración especificado.

```
aws appconfig update-configuration-profile \  
  --application-id 339ohji \  
  --configuration-profile-id ur8hx2f \  
  --description "Configuration profile used for examples."
```

Salida:

```
{  
  "ApplicationId": "339ohji",  
  "Id": "ur8hx2f",  
  "Name": "Example-Configuration-Profile",  
  "Description": "Configuration profile used for examples.",  
  "LocationUri": "ssm-parameter://Example-Parameter",  
  "RetrievalRoleArn": "arn:aws:iam::111122223333:role/Example-App-Config-Role"  
}
```

Para obtener más información, consulte el [paso 3: Crear una configuración y un perfil de configuración](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [UpdateConfigurationProfile](#) la Referencia de AWS CLI comandos.

## update-deployment-strategy

En el siguiente ejemplo de código, se muestra cómo usar `update-deployment-strategy`.

### AWS CLI

Para actualizar una estrategia de despliegue

En el siguiente `update-deployment-strategy` ejemplo, se actualiza el tiempo de horneado final a 20 minutos en la estrategia de despliegue especificada.

```
aws appconfig update-deployment-strategy \  
  --deployment-strategy-id 1225qzk \  
  --final-bake-time-in-minutes 20
```

Salida:

```
{  
  "Id": "1225qzk",  
  "Name": "Example-Deployment",  
  "DeploymentDurationInMinutes": 15,  
  "GrowthType": "LINEAR",  
  "GrowthFactor": 25.0,  
  "FinalBakeTimeInMinutes": 20,  
  "ReplicateTo": "SSM_DOCUMENT"  
}
```

Para obtener más información, consulte el [paso 4: Crear una estrategia de despliegue](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [UpdateDeploymentStrategy](#) la Referencia de AWS CLI comandos.

## update-environment

En el siguiente ejemplo de código, se muestra cómo usar `update-environment`.

AWS CLI

Para actualizar un entorno

El siguiente `update-environment` ejemplo actualiza la descripción de un entorno.

```
aws appconfig update-environment \  
  --application-id 339ohji \  
  --environment-id 54j1r29 \  
  --description "An environment for examples."
```

Salida:

```
{
```

```
"ApplicationId": "339ohji",
  "Id": "54j1r29",
  "Name": "Example-Environment",
  "Description": "An environment for examples.",
  "State": "RolledBack"
}
```

Para obtener más información, consulte el [paso 2: Creación de un entorno](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [UpdateEnvironment](#) la Referencia de AWS CLI comandos.

## update-extension-association

En el siguiente ejemplo de código, se muestra cómo usar `update-extension-association`.

### AWS CLI

Para actualizar una asociación AWS AppConfig de extensiones

El siguiente `update-extension-association` ejemplo agrega un nuevo valor de parámetro a una asociación de extensiones en AWS AppConfig.

```
aws appconfig update-extension-association \
  --region us-west-2 \
  --extension-association-id a1b2c3d4 \
  --parameters S3bucket=FinanceMobileApp
```

Salida:

```
{
  "Id": "a1b2c3d4",
  "ExtensionArn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-backup-extension/1",
  "ResourceArn": "arn:aws:appconfig:us-west-2:123456789012:application/Finance",
  "Parameters": {
    "S3bucket": "FinanceMobileApp"
  },
  "ExtensionVersionNumber": 1
}
```



Para obtener más información, consulte [Trabajar con AWS AppConfig extensiones](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [UpdateExtensionAssociation](#) la Referencia de AWS CLI comandos.

## update-extension

En el siguiente ejemplo de código, se muestra cómo usar `update-extension`.

### AWS CLI

Para actualizar una AWS AppConfig extensión

En el siguiente `update-extension` ejemplo, se agrega una clave de parámetro adicional a una extensión en AWS AppConfig.

```
aws appconfig update-extension \  
  --region us-west-2 \  
  --extension-identifier S3-backup-extension \  
  --parameters S3bucket={Required=true}, CampaignID={Required=false}
```

Salida:

```
{  
  "Id": "1A2B3C4D",  
  "Name": "S3-backup-extension",  
  "VersionNumber": 1,  
  "Arn": "arn:aws:appconfig:us-west-2:123456789012:extension/1A2B3C4D/1",  
  "Actions": {  
    "PRE_CREATE_HOSTED_CONFIGURATION_VERSION": [  
      {  
        "Name": "S3backup",  
        "Uri": "arn:aws:lambda:us-  
west-2:123456789012:function:S3backupfunction",  
        "RoleArn": "arn:aws:iam::123456789012:role/appconfigextensionrole"  
      }  
    ]  
  },  
  "Parameters": {  
    "CampaignID": {  
      "Required": false  
    }  
  }  
}
```

```
    },
    "S3bucket": {
      "Required": true
    }
  }
}
```

Para obtener más información, consulte [Trabajar con AWS AppConfig extensiones](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [UpdateExtension](#) la Referencia de AWS CLI comandos.

## validate-configuration

En el siguiente ejemplo de código, se muestra cómo usar `validate-configuration`.

### AWS CLI

Para validar una configuración

En el siguiente `validate-configuration` ejemplo, se utilizan los validadores de un perfil de configuración para validar una configuración.

```
aws appconfig validate-configuration \
  --application-id abc1234 \
  --configuration-profile-id ur8hx2f \
  --configuration-version 1
```

El comando no produce ningún resultado.

Para obtener más información, consulte el [paso 3: Crear una configuración y un perfil de configuración](#) en la Guía del AWS AppConfig usuario.

- Para API obtener más información, consulte [ValidateConfiguration](#) la Referencia de AWS CLI comandos.

## Ejemplos de Application Auto Scaling utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante Application Auto Scaling. AWS Command Line Interface

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **delete-scaling-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-scaling-policy`.

AWS CLI

Para eliminar una política de escalado

En este ejemplo, se elimina una política de escalado para la aplicación web de Amazon ECS Service que se ejecuta en el clúster predeterminado.

Comando:

```
aws application-autoscaling delete-scaling-policy --policy-name web-app-cpu-lt-25 --scalable-dimension ecs:service:DesiredCount --resource-id service/default/web-app --service-namespace ecs
```

- Para API obtener más información, consulte la Referencia [DeleteScalingPolicy](#) de AWS CLI comandos.

### **delete-scheduled-action**

En el siguiente ejemplo de código, se muestra cómo usar `delete-scheduled-action`.

AWS CLI

Para eliminar una acción programada

El siguiente `delete-scheduled-action` ejemplo elimina la acción programada especificada de la flota de Amazon AppStream 2.0 especificada:

```
aws application-autoscaling delete-scheduled-action \  
  --service-namespace appstream \  
  --scalable-dimension appstream:fleet:DesiredCapacity \  
  --resource-id fleet/sample-fleet \  
  --scheduled-action-name my-recurring-action
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Escalado programado](#) en la Guía del usuario de Auto Scaling de aplicaciones.

- Para API obtener más información, consulte la Referencia [DeleteScheduledAction](#) de AWS CLI comandos.

## deregister-scalable-target

En el siguiente ejemplo de código, se muestra cómo usar `deregister-scalable-target`.

### AWS CLI

Para anular el registro de un objetivo escalable

En este ejemplo se anula el registro de un objetivo escalable para un ECS servicio de Amazon denominado `web-app` que se ejecuta en el clúster predeterminado.

Comando:

```
aws application-autoscaling deregister-scalable-target --service-namespace ecs --  
scalable-dimension ecs:service:DesiredCount --resource-id service/default/web-app
```

En este ejemplo se anula el registro de un objetivo escalable para un recurso personalizado. El `custom-resource-id` archivo.txt contiene una cadena que identifica el ID del recurso, que, en el caso de un recurso personalizado, es la ruta al recurso personalizado a través del punto de enlace de Amazon API Gateway.

Comando:

```
aws application-autoscaling deregister-scalable-target --service-namespace custom-resource --scalable-dimension custom-resource:ResourceType:Property --resource-id file://~/custom-resource-id.txt
```

Contenido del custom-resource-id archivo.txt:

```
https://example.execute-api.us-west-2.amazonaws.com/prod/  
scalableTargetDimensions/1-23456789
```

- Para API obtener más información, consulte [DeregisterScalableTarget](#) la Referencia de AWS CLI comandos.

## describe-scalable-targets

En el siguiente ejemplo de código, se muestra cómo usar describe-scalable-targets.

### AWS CLI

Para describir los objetivos escalables

El siguiente describe-scalable-targets ejemplo describe los destinos escalables del espacio de nombres del ecs servicio.

```
aws application-autoscaling describe-scalable-targets \  
--service-namespace ecs
```

Salida:

```
{  
  "ScalableTargets": [  
    {  
      "ServiceNamespace": "ecs",  
      "ScalableDimension": "ecs:service:DesiredCount",  
      "ResourceId": "service/default/web-app",  
      "MinCapacity": 1,  
      "MaxCapacity": 10,  
      "RoleARN": "arn:aws:iam::123456789012:role/  
aws-service-role/ecs.application-autoscaling.amazonaws.com/  
AWSServiceRoleForApplicationAutoScaling_ECSService",  
      "CreationTime": 1462558906.199,  
      "SuspendedState": {
```

```

        "DynamicScalingOutSuspended": false,
        "ScheduledScalingSuspended": false,
        "DynamicScalingInSuspended": false
    },
    "ScalableTargetARN": "arn:aws:application-autoscaling:us-
west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
    }
]
}

```

Para obtener más información, consulte [AWS los servicios que puede usar con Application Auto Scaling](#) en la Guía del usuario de Application Auto Scaling.

- Para API obtener más información, consulte [DescribeScalableTargets](#) la Referencia de AWS CLI comandos.

## describe-scaling-activities

En el siguiente ejemplo de código, se muestra cómo usar `describe-scaling-activities`.

### AWS CLI

Ejemplo 1: Para describir las actividades de escalado para el ECS servicio de Amazon especificado

El siguiente `describe-scaling-activities` ejemplo describe las actividades de escalado de un ECS servicio llamado Amazon web-app que se ejecuta en el default clúster. El resultado muestra una actividad de escalado iniciada por una política de escalado.

```

aws application-autoscaling describe-scaling-activities \
  --service-namespace ecs \
  --resource-id service/default/web-app

```

Salida:

```

{
  "ScalingActivities": [
    {
      "ScalableDimension": "ecs:service:DesiredCount",
      "Description": "Setting desired count to 1.",
      "ResourceId": "service/default/web-app",
      "ActivityId": "e6c5f7d1-dbbb-4a3f-89b2-51f33e766399",

```

```

        "StartTime": 1462575838.171,
        "ServiceNamespace": "ecs",
        "EndTime": 1462575872.111,
        "Cause": "monitor alarm web-app-cpu-1t-25 in state ALARM triggered
policy web-app-cpu-1t-25",
        "StatusMessage": "Successfully set desired count to 1. Change
successfully fulfilled by ecs.",
        "StatusCode": "Successful"
    }
]
}

```

Para obtener más información, consulte [las actividades de escalado de Application Auto Scaling](#) en la Guía del usuario de Application Auto Scaling.

Ejemplo 2: Para describir las actividades de escalado de la tabla de DynamoDB especificada

El siguiente `describe-scaling-activities` ejemplo describe las actividades de escalado de una tabla de DynamoDB llamada. `TestTable` El resultado muestra las actividades de escalado iniciadas por dos acciones programadas diferentes.

```

aws application-autoscaling describe-scaling-activities \
  --service-namespace dynamodb \
  --resource-id table/TestTable

```

Salida:

```

{
  "ScalingActivities": [
    {
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
      "Description": "Setting write capacity units to 10.",
      "ResourceId": "table/my-table",
      "ActivityId": "4d1308c0-bbcf-4514-a673-b0220ae38547",
      "StartTime": 1561574415.086,
      "ServiceNamespace": "dynamodb",
      "EndTime": 1561574449.51,
      "Cause": "maximum capacity was set to 10",
      "StatusMessage": "Successfully set write capacity units to 10. Change
successfully fulfilled by dynamodb.",
      "StatusCode": "Successful"
    },
    {

```

```

    "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
    "Description": "Setting min capacity to 5 and max capacity to 10",
    "ResourceId": "table/my-table",
    "ActivityId": "f2b7847b-721d-4e01-8ef0-0c8d3bacc1c7",
    "StartTime": 1561574414.644,
    "ServiceNamespace": "dynamodb",
    "Cause": "scheduled action name my-second-scheduled-action was
triggered",
    "StatusMessage": "Successfully set min capacity to 5 and max capacity to
10",
    "StatusCode": "Successful"
  },
  {
    "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
    "Description": "Setting write capacity units to 15.",
    "ResourceId": "table/my-table",
    "ActivityId": "d8ea4de6-9eaa-499f-b466-2cc5e681ba8b",
    "StartTime": 1561574108.904,
    "ServiceNamespace": "dynamodb",
    "EndTime": 1561574140.255,
    "Cause": "minimum capacity was set to 15",
    "StatusMessage": "Successfully set write capacity units to 15. Change
successfully fulfilled by dynamodb.",
    "StatusCode": "Successful"
  },
  {
    "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
    "Description": "Setting min capacity to 15 and max capacity to 20",
    "ResourceId": "table/my-table",
    "ActivityId": "3250fd06-6940-4e8e-bb1f-d494db7554d2",
    "StartTime": 1561574108.512,
    "ServiceNamespace": "dynamodb",
    "Cause": "scheduled action name my-first-scheduled-action was
triggered",
    "StatusMessage": "Successfully set min capacity to 15 and max capacity
to 20",
    "StatusCode": "Successful"
  }
]
}

```

Para obtener más información, consulte [las actividades de escalado de Application Auto Scaling](#) en la Guía del usuario de Application Auto Scaling.



- Para API obtener más información, consulte [DescribeScalingActivities](#) la Referencia de AWS CLI comandos.

## describe-scaling-policies

En el siguiente ejemplo de código, se muestra cómo usar `describe-scaling-policies`.

### AWS CLI

Para describir las políticas de escalado

Este comando de ejemplo describe las políticas de escalado del espacio de nombres del servicio `ecs`.

Comando:

```
aws application-autoscaling describe-scaling-policies --service-namespace ecs
```

Salida:

```
{
  "ScalingPolicies": [
    {
      "PolicyName": "web-app-cpu-gt-75",
      "ScalableDimension": "ecs:service:DesiredCount",
      "ResourceId": "service/default/web-app",
      "CreationTime": 1462561899.23,
      "StepScalingPolicyConfiguration": {
        "Cooldown": 60,
        "StepAdjustments": [
          {
            "ScalingAdjustment": 200,
            "MetricIntervalLowerBound": 0.0
          }
        ],
        "AdjustmentType": "PercentChangeInCapacity"
      },
      "PolicyARN": "arn:aws:autoscaling:us-
west-2:012345678910:scalingPolicy:6d8972f3-efc8-437c-92d1-6270f29a66e7:resource/ecs/
service/default/web-app:policyName/web-app-cpu-gt-75",
      "PolicyType": "StepScaling",
    }
  ]
}
```

```

    "Alarms": [
      {
        "AlarmName": "web-app-cpu-gt-75",
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:web-app-cpu-gt-75"
      }
    ],
    "ServiceNamespace": "ecs"
  },
  {
    "PolicyName": "web-app-cpu-lt-25",
    "ScalableDimension": "ecs:service:DesiredCount",
    "ResourceId": "service/default/web-app",
    "CreationTime": 1462562575.099,
    "StepScalingPolicyConfiguration": {
      "Cooldown": 1,
      "StepAdjustments": [
        {
          "ScalingAdjustment": -50,
          "MetricIntervalUpperBound": 0.0
        }
      ],
      "AdjustmentType": "PercentChangeInCapacity"
    },
    "PolicyARN": "arn:aws:autoscaling:us-
west-2:012345678910:scalingPolicy:6d8972f3-efc8-437c-92d1-6270f29a66e7:resource/ecs/
service/default/web-app:policyName/web-app-cpu-lt-25",
    "PolicyType": "StepScaling",
    "Alarms": [
      {
        "AlarmName": "web-app-cpu-lt-25",
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:web-app-cpu-lt-25"
      }
    ],
    "ServiceNamespace": "ecs"
  }
]
}

```

- Para API obtener más información, consulte la Referencia [DescribeScalingPolicies](#) de AWS CLI comandos.

## describe-scheduled-actions

En el siguiente ejemplo de código, se muestra cómo usar `describe-scheduled-actions`.

### AWS CLI

Para describir las acciones programadas

En el siguiente `describe-scheduled-actions` ejemplo, se muestran los detalles de las acciones programadas para el espacio de nombres del servicio especificado:

```
aws application-autoscaling describe-scheduled-actions \  
--service-namespace dynamodb
```

Salida:

```
{  
  "ScheduledActions": [  
    {  
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",  
      "Schedule": "at(2019-05-20T18:35:00)",  
      "ResourceId": "table/my-table",  
      "CreationTime": 1561571888.361,  
      "ScheduledActionARN": "arn:aws:autoscaling:us-  
west-2:123456789012:scheduledAction:2d36aa3b-cdf9-4565-b290-81db519b227d:resource/  
dynamodb/table/my-table:scheduledActionName/my-first-scheduled-action",  
      "ScalableTargetAction": {  
        "MinCapacity": 15,  
        "MaxCapacity": 20  
      },  
      "ScheduledActionName": "my-first-scheduled-action",  
      "ServiceNamespace": "dynamodb"  
    },  
    {  
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",  
      "Schedule": "at(2019-05-20T18:40:00)",  
      "ResourceId": "table/my-table",  
      "CreationTime": 1561571946.021,  
      "ScheduledActionARN": "arn:aws:autoscaling:us-  
west-2:123456789012:scheduledAction:2d36aa3b-cdf9-4565-b290-81db519b227d:resource/  
dynamodb/table/my-table:scheduledActionName/my-second-scheduled-action",  
      "ScalableTargetAction": {  
        "MinCapacity": 5,  

```

```
        "MaxCapacity": 10
      },
      "ScheduledActionName": "my-second-scheduled-action",
      "ServiceNamespace": "dynamodb"
    }
  ]
}
```

Para obtener más información, consulte [Escalado programado](#) en la Guía del usuario de Auto Scaling de aplicaciones.

- Para API obtener más información, consulte la Referencia [DescribeScheduledActions](#) de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un objetivo escalable

En el siguiente `list-tags-for-resource` ejemplo, se enumeran los nombres y valores de las claves de las etiquetas que se adjuntan al objetivo escalable especificado por élARN.

```
aws application-autoscaling list-tags-for-resource \
  --resource-arn arn:aws:application-autoscaling:us-west-2:123456789012:scalable-
  target/1234abcd56ab78cd901ef1234567890ab123
```

Salida:

```
{
  "Tags": {
    "environment": "production"
  }
}
```

Para obtener más información, consulte [Soporte de etiquetado para Application Auto Scaling](#) en la Guía del usuario de Application Auto Scaling.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## put-scaling-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-scaling-policy`.

### AWS CLI

Ejemplo 1: aplicar una política de escalado de seguimiento de destino con una especificación de métrica predefinida

En el siguiente `put-scaling-policy` ejemplo, se aplica una política de escalado de seguimiento objetivo con una especificación de métrica predefinida a un ECS servicio de Amazon denominado `web-app` en el clúster predeterminado. La política mantiene la CPU utilización media del servicio en un 75 por ciento, con períodos de recarga de 60 segundos durante el escalado horizontal y el escalado interno. El resultado contiene los nombres ARNs y los nombres de las dos CloudWatch alarmas creadas en su nombre.

```
aws application-autoscaling put-scaling-policy --service-namespace ecs \  
--scalable-dimension ecs:service:DesiredCount \  
--resource-id service/default/web-app \  
--policy-name cpu75-target-tracking-scaling-policy --policy-  
type TargetTrackingScaling \  
--target-tracking-scaling-policy-configuration file://config.json
```

En este ejemplo se supone que tiene un archivo `config.json` en el directorio actual con el siguiente contenido:

```
{  
  "TargetValue": 75.0,  
  "PredefinedMetricSpecification": {  
    "PredefinedMetricType": "ECSServiceAverageCPUUtilization"  
  },  
  "ScaleOutCooldown": 60,  
  "ScaleInCooldown": 60  
}
```

Salida:

```
{  
  "PolicyARN": "arn:aws:autoscaling:us-west-2:012345678910:scalingPolicy:6d8972f3-  
efc8-437c-92d1-6270f29a66e7:resource/ecs/service/default/web-app:policyName/cpu75-  
target-tracking-scaling-policy",
```

```

    "Alarms": [
      {
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca",
        "AlarmName": "TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca"
      },
      {
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:TargetTracking-service/default/web-app-AlarmLow-1b437334-
d19b-4a63-a812-6c67aaf2910d",
        "AlarmName": "TargetTracking-service/default/web-app-AlarmLow-1b437334-
d19b-4a63-a812-6c67aaf2910d"
      }
    ]
  }

```

Ejemplo 2: aplicar una política de escalado de seguimiento de destino con una especificación de métrica personalizada

En el siguiente `put-scaling-policy` ejemplo, se aplica una política de escalado de seguimiento objetivo con una especificación de métrica personalizada a un ECS servicio de Amazon denominado `web-app` en el clúster predeterminado. La política mantiene la utilización media del servicio en un 75 por ciento, con períodos de recarga de 60 segundos durante el escalado horizontal y el escalado interno. El resultado contiene los nombres ARNs y los nombres de las dos CloudWatch alarmas creadas en su nombre.

```

aws application-autoscaling put-scaling-policy --service-namespace ecs \
--scalable-dimension ecs:service:DesiredCount \
--resource-id service/default/web-app \
--policy-name cms75-target-tracking-scaling-policy \
--policy-type TargetTrackingScaling \
--target-tracking-scaling-policy-configuration file://config.json

```

En este ejemplo se supone que tiene un archivo `config.json` en el directorio actual con el siguiente contenido:

```

{
  "TargetValue":75.0,
  "CustomizedMetricSpecification":{
    "MetricName":"MyUtilizationMetric",

```

```

    "Namespace": "MyNamespace",
    "Dimensions": [
      {
        "Name": "MyOptionalMetricDimensionName",
        "Value": "MyOptionalMetricDimensionValue"
      }
    ],
    "Statistic": "Average",
    "Unit": "Percent"
  },
  "ScaleOutCooldown": 60,
  "ScaleInCooldown": 60
}

```

Salida:

```

{
  "PolicyARN": "arn:aws:autoscaling:us-west-2:012345678910:scalingPolicy:8784a896-b2ba-47a1-b08c-27301cc499a1:resource/ecs/service/default/web-app:policyName/cms75-target-tracking-scaling-policy",
  "Alarms": [
    {
      "AlarmARN": "arn:aws:cloudwatch:us-west-2:012345678910:alarm:TargetTracking-service/default/web-app-AlarmHigh-9bc77b56-0571-4276-ba0f-d4178882e0a0",
      "AlarmName": "TargetTracking-service/default/web-app-AlarmHigh-9bc77b56-0571-4276-ba0f-d4178882e0a0"
    },
    {
      "AlarmARN": "arn:aws:cloudwatch:us-west-2:012345678910:alarm:TargetTracking-service/default/web-app-AlarmLow-9b6ad934-6d37-438e-9e05-02836ddcbdc4",
      "AlarmName": "TargetTracking-service/default/web-app-AlarmLow-9b6ad934-6d37-438e-9e05-02836ddcbdc4"
    }
  ]
}

```

**Ejemplo 3:** aplicar una política de escalado de seguimiento de destino solo para el escalado ascendente

En el siguiente `put-scaling-policy` ejemplo, se aplica una política de escalado de seguimiento de objetivos a un ECS servicio de Amazon llamado `web-app` en el

clúster predeterminado. La política se usa para escalar el ECS servicio cuando la RequestCountPerTarget métrica del Application Load Balancer supera el umbral. El resultado contiene el nombre ARN y el nombre de la CloudWatch alarma creada en su nombre.

```
aws application-autoscaling put-scaling-policy \
  --service-namespace ecs \
  --scalable-dimension ecs:service:DesiredCount \
  --resource-id service/default/web-app \
  --policy-name alb-scale-out-target-tracking-scaling-policy \
  --policy-type TargetTrackingScaling \
  --target-tracking-scaling-policy-configuration file://config.json
```

Contenidos de config.json:

```
{
  "TargetValue": 1000.0,
  "PredefinedMetricSpecification": {
    "PredefinedMetricType": "ALBRequestCountPerTarget",
    "ResourceLabel": "app/EC2Co-EcsE1-1TKLTMITMM0E0/f37c06a68c1748aa/
targetgroup/EC2Co-Defau-LDNM7Q3ZH1ZN/6d4ea56ca2d6a18d"
  },
  "ScaleOutCooldown": 60,
  "ScaleInCooldown": 60,
  "DisableScaleIn": true
}
```

Salida:

```
{
  "PolicyARN": "arn:aws:autoscaling:us-west-2:123456789012:scalingPolicy:6d8972f3-
efc8-437c-92d1-6270f29a66e7:resource/ecs/service/default/web-app:policyName/alb-
scale-out-target-tracking-scaling-policy",
  "Alarms": [
    {
      "AlarmName": "TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca",
      "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca"
    }
  ]
}
```



Para obtener más información, consulte [Target Tracking Scaling Policies for Application Auto Scaling](#) en la Guía del usuario de AWS Application Auto Scaling.

- Para API obtener más información, consulte [PutScalingPolicy](#) la Referencia de AWS CLI comandos.

## put-scheduled-action

En el siguiente ejemplo de código, se muestra cómo usar `put-scheduled-action`.

### AWS CLI

Para añadir una acción programada a una tabla de DynamoDB

En este ejemplo, se agrega una acción programada a una tabla de DynamoDB `TestTable` llamada a escalar de forma horizontal según una programación recurrente. Según el programa especificado (todos los días a las 12:15 p.m. UTC), si la capacidad actual está por debajo del valor especificado `MinCapacity`, Application Auto Scaling se amplía hasta el valor especificado por `MinCapacity`.

Comando:

```
aws application-autoscaling put-scheduled-action --service-namespace dynamodb
--scheduled-action-name my-recurring-action --schedule "cron(15 12 * * ? *)" --
resource-id table/TestTable --scalable-dimension dynamodb:table:WriteCapacityUnits
--scalable-target-action MinCapacity=6
```

Para obtener más información, consulte `Scheduled Scaling` en la Guía del usuario de Application Auto Scaling.

- Para API obtener más información, consulte [PutScheduledAction](#) la Referencia de AWS CLI comandos.

## register-scalable-target

En el siguiente ejemplo de código, se muestra cómo usar `register-scalable-target`.

### AWS CLI

Ejemplo 1: Para registrar un ECS servicio como destino escalable

El siguiente `register-scalable-target` ejemplo registra un ECS servicio de Amazon con Application Auto Scaling. También añade una etiqueta con el nombre de la clave `environment` y el valor `production` al objetivo escalable.

```
aws application-autoscaling register-scalable-target \
  --service-namespace ecs \
  --scalable-dimension ecs:service:DesiredCount \
  --resource-id service/default/web-app \
  --min-capacity 1 --max-capacity 10 \
  --tags environment=production
```

Salida:

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:us-
west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

Para ver ejemplos de otros AWS servicios y recursos personalizados, consulte los temas de los [AWS servicios que puede usar con Application Auto Scaling](#) en la Guía del usuario de Application Auto Scaling.

Ejemplo 2: Suspender las actividades de escalado para un objetivo escalable

El siguiente `register-scalable-target` ejemplo suspende las actividades de escalado para un objetivo escalable existente.

```
aws application-autoscaling register-scalable-target \
  --service-namespace dynamodb \
  --scalable-dimension dynamodb:table:ReadCapacityUnits \
  --resource-id table/my-table \
  --suspended-
state DynamicScalingInSuspended=true,DynamicScalingOutSuspended=true,ScheduledScalingSuspenda
```

Salida:

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:us-
west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

Para obtener más información, consulte [Suspender y reanudar el escalado de Application Auto Scaling en la Guía](#) del usuario de Application Auto Scaling.

Ejemplo 3: Para reanudar las actividades de escalado para un objetivo escalable

En el siguiente `register-scalable-target` ejemplo, se reanudan las actividades de escalado para un objetivo escalable existente.

```
aws application-autoscaling register-scalable-target \
  --service-namespace dynamodb \
  --scalable-dimension dynamodb:table:ReadCapacityUnits \
  --resource-id table/my-table \
  --suspended-
state DynamicScalingInSuspended=false,DynamicScalingOutSuspended=false,ScheduledScalingSuspe
```

Salida:

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:us-
west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

Para obtener más información, consulte [Suspender y reanudar el escalado de Application Auto Scaling en la Guía](#) del usuario de Application Auto Scaling.

- Para API obtener más información, consulte la Referencia [RegisterScalableTarget](#) de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

AWS CLI

Para añadir una etiqueta a un objetivo escalable

En el siguiente `tag-resource` ejemplo, se agrega una etiqueta con el nombre de la clave `environment` y el valor `production` al objetivo escalable especificado por su ARN.

```
aws application-autoscaling tag-resource \
  --resource-arn arn:aws:application-autoscaling:us-west-2:123456789012:scalable-
target/1234abcd56ab78cd901ef1234567890ab123 \
```

```
--tags environment=production
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Soporte de etiquetado para Application Auto Scaling](#) en la Guía del usuario de Application Auto Scaling.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar untag-resource.

### AWS CLI

Para eliminar una etiqueta de un objetivo escalable

En el siguiente untag-resource ejemplo, se elimina el par de etiquetas con el nombre `environment` de la clave del objetivo escalable especificado por suARN.

```
aws application-autoscaling untag-resource \  
  --resource-arn arn:aws:application-autoscaling:us-west-2:123456789012:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123 \  
  --tag-keys "environment"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Soporte de etiquetado para Application Auto Scaling](#) en la Guía del usuario de Application Auto Scaling.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## Ejemplos de Application Discovery Service que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante Application Discovery Service. AWS Command Line Interface

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **describe-agents**

En el siguiente ejemplo de código, se muestra cómo usar `describe-agents`.

#### AWS CLI

Describa los agentes con `collectionStatus` estados específicos

Este comando de ejemplo describe los agentes de recopilación con el estado de recopilación `STARTED` «» o «`STOPPED`».

Comando:

```
aws discovery describe-agents --filters
name="collectionStatus",values="STARTED","STOPPED",condition="EQUALS" --max-
results 3
```

Salida:

```
{
  "Snapshots": [
    {
      "version": "1.0.40.0",
      "agentType": "EC2",
      "hostName": "ip-172-31-40-234",
      "collectionStatus": "STOPPED",
      "agentNetworkInfoList": [
        {
          "macAddress": "06:b5:97:14:fc:0d",
          "ipAddress": "172.31.40.234"
        }
      ],
      "health": "UNKNOWN",
```

```

    "agentId": "i-003305c02a776e883",
    "registeredTime": "2016-12-09T19:05:06Z",
    "lastHealthPingTime": "2016-12-09T19:05:10Z"
  },
  {
    "version": "1.0.40.0",
    "agentType": "EC2",
    "hostName": "ip-172-31-39-64",
    "collectionStatus": "STARTED",
    "agentNetworkInfoList": [
      {
        "macAddress": "06:a1:0e:c7:b2:73",
        "ipAddress": "172.31.39.64"
      }
    ],
    "health": "SHUTDOWN",
    "agentId": "i-003a5e5e2b36cf8bd",
    "registeredTime": "2016-11-16T16:36:25Z",
    "lastHealthPingTime": "2016-11-16T16:47:37Z"
  }
]
}

```

- Para API obtener más información, consulte [DescribeAgents](#) la Referencia de AWS CLI comandos.

## describe-configurations

En el siguiente ejemplo de código, se muestra cómo usar `describe-configurations`.

### AWS CLI

Describe las configuraciones de activos seleccionadas

Este comando de ejemplo describe las configuraciones de dos servidores específicos. La acción detecta el tipo de activo a partir del ID de configuración. Solo se permite un tipo de activo por comando.

Comando:

```
aws discovery describe-configurations --configuration-ids "d-  
server-099385097ef9fbcfb" "d-server-0c4f2dd1fee22c6c1"
```

## Salida:

```
{
  "configurations": [
    {
      "server.performance.maxCpuUsagePct": "0.0",
      "server.performance.maxDiskReadIOPS": "0.0",
      "server.performance.avgCpuUsagePct": "0.0",
      "server.type": "EC2",
      "server.performance.maxNetworkReadsPerSecondInKB": "0.19140625",
      "server.hostName": "ip-172-31-35-152",
      "server.configurationId": "d-server-0c4f2dd1fee22c6c1",
      "server.tags.hasMoreValues": "false",
      "server.performance.minFreeRAMInKB": "1543496.0",
      "server.osVersion": "3.14.48-33.39.amzn1.x86_64",
      "server.performance.maxDiskReadsPerSecondInKB": "0.0",
      "server.applications": "[]",
      "server.performance.numDisks": "1",
      "server.performance.numCpus": "1",
      "server.performance.numCores": "1",
      "server.performance.maxDiskWriteIOPS": "0.0",
      "server.performance.maxNetworkWritesPerSecondInKB": "0.82421875",
      "server.performance.avgDiskWritesPerSecondInKB": "0.0",
      "server.networkInterfaceInfo": "[{\"name\":\"eth0\",
\\\"macAddress\\\":\\\"06:A7:7D:3F:54:57\\\",\\\"ipAddress\\\":\\\"172.31.35.152\\\",\\\"netMask\\\":
\\\"255.255.240.0\\\"},{\"name\":\"lo\",\\\"macAddress\\\":\\\"00:00:00:00:00:00\\\",\\\"ipAddress
\\\":\\\"127.0.0.1\\\",\\\"netMask\\\":\\\"255.0.0.0\\\"},{\"name\":\"eth0\",\\\"macAddress\\\":
\\\"06:A7:7D:3F:54:57\\\",\\\"ipAddress\\\":\\\"fe80::4a7:7dff:fe3f:5457\\\"},{\"name\":\"lo\",
\\\"macAddress\\\":\\\"00:00:00:00:00:00\\\",\\\"ipAddress\\\":\\\":::1\\\"}]",
      "server.performance.avgNetworkReadsPerSecondInKB":
"0.049153645833333333",
      "server.tags": "[]",
      "server.applications.hasMoreValues": "false",
      "server.timeOfCreation": "2016-10-28 23:44:00.0",
      "server.agentId": "i-4447bc1b",
      "server.performance.maxDiskWritesPerSecondInKB": "0.0",
      "server.performance.avgDiskReadIOPS": "0.0",
      "server.performance.avgFreeRAMInKB": "1547210.1333333333",
      "server.performance.avgDiskReadsPerSecondInKB": "0.0",
      "server.performance.avgDiskWriteIOPS": "0.0",
      "server.performance.numNetworkCards": "2",
      "server.hypervisor": "xen",
      "server.networkInterfaceInfo.hasMoreValues": "false",
      "server.performance.avgNetworkWritesPerSecondInKB": "0.1380859375",
```

```

"server.osName": "Linux - Amazon Linux AMI release 2015.03",
"server.performance.totalRAMInKB": "1694732.0",
"server.cpuType": "x64"
},
{
  "server.performance.maxCpuUsagePct": "100.0",
  "server.performance.maxDiskReadIOPS": "0.0",
  "server.performance.avgCpuUsagePct": "14.733333333333338",
  "server.type": "EC2",
  "server.performance.maxNetworkReadsPerSecondInKB": "13.400390625",
  "server.hostName": "ip-172-31-42-208",
  "server.configurationId": "d-server-099385097ef9fbcb",
  "server.tags.hasMoreValues": "false",
  "server.performance.minFreeRAMInKB": "1531104.0",
  "server.osVersion": "3.14.48-33.39.amzn1.x86_64",
  "server.performance.maxDiskReadsPerSecondInKB": "0.0",
  "server.applications": "[]",
  "server.performance.numDisks": "1",
  "server.performance.numCpus": "1",
  "server.performance.numCores": "1",
  "server.performance.maxDiskWriteIOPS": "1.0",
  "server.performance.maxNetworkWritesPerSecondInKB": "12.271484375",
  "server.performance.avgDiskWritesPerSecondInKB":
"0.5333333333333334",
  "server.networkInterfaceInfo": "[{"name":"eth0",
  "macAddress":"06:4A:79:60:75:61","ipAddress":"172.31.42.208","netMask
":"255.255.240.0"}, {"name":"eth0", "macAddress":"06:4A:79:60:75:61",
  "ipAddress":"fe80::44a:79ff:fe60:7561"}, {"name":"lo", "macAddress":
"00:00:00:00:00:00","ipAddress":"",":1"}, {"name":"lo", "macAddress":
"00:00:00:00:00:00","ipAddress":"127.0.0.1","netMask":"255.0.0.0"}]",
  "server.performance.avgNetworkReadsPerSecondInKB":
"2.8720052083333334",
  "server.tags": "[]",
  "server.applications.hasMoreValues": "false",
  "server.timeOfCreation": "2016-10-28 23:44:30.0",
  "server.agentId": "i-c142b99e",
  "server.performance.maxDiskWritesPerSecondInKB": "4.0",
  "server.performance.avgDiskReadIOPS": "0.0",
  "server.performance.avgFreeRAMInKB": "1534946.4",
  "server.performance.avgDiskReadsPerSecondInKB": "0.0",
  "server.performance.avgDiskWriteIOPS": "0.13333333333333336",
  "server.performance.numNetworkCards": "2",
  "server.hypervisor": "xen",
  "server.networkInterfaceInfo.hasMoreValues": "false",

```



```

        "server.performance.avgNetworkWritesPerSecondInKB":
"1.7977864583333332",
        "server.osName": "Linux - Amazon Linux AMI release 2015.03",
        "server.performance.totalRAMInKB": "1694732.0",
        "server.cpuType": "x64"
    }
]
}

```

Describe las configuraciones de activos seleccionadas

Este comando de ejemplo describe las configuraciones de dos aplicaciones especificadas. La acción detecta el tipo de activo a partir del ID de configuración. Solo se permite un tipo de activo por comando.

Comando:

```

aws discovery describe-configurations --configuration-ids "d-
application-0ac39bc0e4fad0e42" "d-application-02444a45288013764q"

```

Salida:

```

{
  "configurations": [
    {
      "application.serverCount": "0",
      "application.name": "Application-12345",
      "application.lastModifiedTime": "2016-12-13 23:53:27.0",
      "application.description": "",
      "application.timeOfCreation": "2016-12-13 23:53:27.0",
      "application.configurationId": "d-application-0ac39bc0e4fad0e42"
    },
    {
      "application.serverCount": "0",
      "application.name": "Application-67890",
      "application.lastModifiedTime": "2016-12-13 23:53:33.0",
      "application.description": "",
      "application.timeOfCreation": "2016-12-13 23:53:33.0",
      "application.configurationId": "d-application-02444a45288013764"
    }
  ]
}

```

- Para API obtener más información, consulte [DescribeConfigurations](#) la Referencia de AWS CLI comandos.

## list-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-configurations`.

### AWS CLI

Para ver una lista de todos los servidores detectados que cumplen un conjunto de condiciones de filtrado

Este comando de ejemplo muestra los servidores detectados que coinciden con uno de los dos patrones de nombres de host y que no ejecutan Ubuntu.

Comando:

```
aws discovery list-configurations --configuration-type SERVER --filters
name="server.hostName",values="172-31-35","172-31-42",condition="CONTAINS"
name="server.osName",values="Ubuntu",condition="NOT_CONTAINS"
```

Salida:

```
{
  "configurations": [
    {
      "server.osVersion": "3.14.48-33.39.amzn1.x86_64",
      "server.type": "EC2",
      "server.hostName": "ip-172-31-42-208",
      "server.timeOfCreation": "2016-10-28 23:44:30.0",
      "server.configurationId": "d-server-099385097ef9fbcfb",
      "server.osName": "Linux - Amazon Linux AMI release 2015.03",
      "server.agentId": "i-c142b99e"
    },
    {
      "server.osVersion": "3.14.48-33.39.amzn1.x86_64",
      "server.type": "EC2",
      "server.hostName": "ip-172-31-35-152",
      "server.timeOfCreation": "2016-10-28 23:44:00.0",
      "server.configurationId": "d-server-0c4f2dd1fee22c6c1",
      "server.osName": "Linux - Amazon Linux AMI release 2015.03",

```

```
        "server.agentId": "i-4447bc1b"
      }
    ]
  }
```

- Para API obtener más información, consulte [ListConfigurations](#) la Referencia de AWS CLI comandos.

## AppRegistry ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AppRegistry.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **associate-attribute-group**

En el siguiente ejemplo de código, se muestra cómo usar `associate-attribute-group`.

### AWS CLI

Para asociar un grupo de atributos

El siguiente `associate-attribute-group` ejemplo asocia un grupo de atributos específico de su AWS cuenta a una aplicación específica de su AWS cuenta.

```
aws servicecatalog-appregistry associate-attribute-group \  
  --application "ExampleApplication" \  
  --attribute-group "ExampleAttributeGroup"
```

Salida:

```
{
  "applicationArn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/0ars38r6btoohvpvd9gqrptt91",
  "attributeGroupArn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/01sj5xdwhbw54kejwnt09fnpc1"
}
```

Para obtener más información, consulte [Asociar y desasociar grupos de atributos](#) en la Guía del administrador de AWS Service Catalog AppRegistry .

- Para API obtener más información, consulte la Referencia de [AssociateAttributeGroup](#) comandos AWS CLI .

## create-application

En el siguiente ejemplo de código, se muestra cómo usar create-application.

AWS CLI

Para crear una aplicación

En el siguiente create-application ejemplo, se crea una nueva aplicación en su AWS cuenta.

```
aws servicecatalog-appregistry create-application \
  --name "ExampleApplication"
```

Salida:

```
{
  "application": {
    "id": "0ars38r6btoohvpvd9gqrptt91",
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/0ars38r6btoohvpvd9gqrptt91",
    "name": "ExampleApplication",
    "creationTime": "2023-02-28T21:10:10.820000+00:00",
    "lastUpdateTime": "2023-02-28T21:10:10.820000+00:00",
    "tags": {}
  }
}
```

```
}
```

Para obtener más información, consulte [Creación de aplicaciones](#) en la Guía del AppRegistry administrador de AWS Service Catalog.

- Para API obtener más información, consulte [CreateApplication](#) la Referencia de AWS CLI comandos.

## create-attribute-group

En el siguiente ejemplo de código, se muestra cómo usar `create-attribute-group`.

### AWS CLI

Para crear un grupo de atributos

En el siguiente `create-attribute-group` ejemplo, se crea un nuevo grupo de atributos en su AWS cuenta.

```
aws servicecatalog-appregistry create-attribute-group \  
  --name "ExampleAttributeGroup" \  
  --attributes '{"SomeKey1":"SomeValue1","SomeKey2":"SomeValue2"}'
```

Salida:

```
{  
  "attributeGroup": {  
    "id": "01sj5xdwhbw54kejwnt09fnpc1",  
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-  
groups/01sj5xdwhbw54kejwnt09fnpc1",  
    "name": "ExampleAttributeGroup",  
    "creationTime": "2023-02-28T20:38:01.389000+00:00",  
    "lastUpdateTime": "2023-02-28T20:38:01.389000+00:00",  
    "tags": {}  
  }  
}
```

Para obtener más información, consulte [Creación de grupos de atributos](#) en la Guía del AppRegistry administrador de AWS Service Catalog.

- Para API obtener más información, consulte [CreateAttributeGroup](#) la Referencia de AWS CLI comandos.

## delete-application

En el siguiente ejemplo de código, se muestra cómo usar delete-application.

### AWS CLI

Eliminación de una aplicación

En el siguiente delete-application ejemplo, se elimina una aplicación específica de su AWS cuenta.

```
aws servicecatalog-appregistry delete-application \  
  --application "ExampleApplication3"
```

Salida:

```
{  
  "application": {  
    "id": "055gw7aynr1i5mbv7kjwtzx5945",  
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/  
applications/055gw7aynr1i5mbv7kjwtzx5945",  
    "name": "ExampleApplication3",  
    "creationTime": "2023-02-28T22:06:28.228000+00:00",  
    "lastUpdateTime": "2023-02-28T22:06:28.228000+00:00"  
  }  
}
```

Para obtener más información, consulte [Eliminar aplicaciones](#) en la Guía del AppRegistry administrador de AWS Service Catalog.

- Para API obtener más información, consulte [DeleteApplication](#) la Referencia de AWS CLI comandos.

## delete-attribute-group

En el siguiente ejemplo de código, se muestra cómo usar delete-attribute-group.

### AWS CLI

Ejemplo 8: Para eliminar un grupo de atributos

En el siguiente delete-attribute-group ejemplo, se elimina un grupo de atributos específico de su AWS cuenta.

```
aws servicecatalog-appregistry delete-attribute-group \  
--attribute-group "ExampleAttributeGroup3"
```

Salida:

```
{  
  "attributeGroup": {  
    "id": "011ge6y3emyjijt8dw8jn6r0hv",  
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-  
groups/011ge6y3emyjijt8dw8jn6r0hv",  
    "name": "ExampleAttributeGroup3",  
    "creationTime": "2023-02-28T22:05:35.224000+00:00",  
    "lastUpdateTime": "2023-02-28T22:05:35.224000+00:00"  
  }  
}
```

Para obtener más información, consulte [Eliminar grupos de atributos](#) en la Guía del AppRegistry administrador de AWS Service Catalog.

- Para API obtener más información, consulte [DeleteAttributeGroup](#) la Referencia de AWS CLI comandos.

## get-application

En el siguiente ejemplo de código, se muestra cómo usar `get-application`.

AWS CLI

Para obtener una solicitud

En el siguiente `get-application` ejemplo, se recupera la información de metadatos sobre una aplicación específica de su AWS cuenta.

```
aws servicecatalog-appregistry get-application \  
--application "ExampleApplication"
```

Salida:

```
{  
  "id": "0ars38r6btoohvpvd9gqrptt91",  
  "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/  
applications/0ars38r6btoohvpvd9gqrptt91",
```

```

    "name": "ExampleApplication",
    "creationTime": "2023-02-28T21:10:10.820000+00:00",
    "lastUpdateTime": "2023-02-28T21:10:10.820000+00:00",
    "associatedResourceCount": 0,
    "tags": {
      "aws:servicecatalog:applicationName": "ExampleApplication"
    },
    "integrations": {
      "resourceGroup": {
        "state": "CREATE_COMPLETE",
        "arn": "arn:aws:resource-groups:us-west-2:813737243517:group/
AWS_AppRegistry_Application-ExampleApplication"
      }
    }
  }
}

```

Para obtener más información, consulte [Uso de los detalles de la aplicación](#) en la Guía del AppRegistry administrador de AWS Service Catalog.

- Para API obtener más información, consulte [GetApplication](#) la Referencia de AWS CLI comandos.

## get-attribute-group

En el siguiente ejemplo de código, se muestra cómo usar `get-attribute-group`.

### AWS CLI

Para obtener un grupo de atributos

En el siguiente `get-attribute-group` ejemplo, se recupera un grupo de atributos específico de su AWS cuenta.

```

aws servicecatalog-appregistry get-attribute-group \
  --attribute-group ExampleAttributeGroup

```

Salida:

```

{
  "id": "01sj5xdwhbw54kejwnt09fnpc1",
  "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/01sj5xdwhbw54kejwnt09fnpc1",
  "name": "ExampleAttributeGroup",

```



```
"attributes": "{\"SomeKey1\":\"SomeValue1\", \"SomeKey2\":\"SomeValue2\"}\",
"creationTime": "2023-02-28T20:38:01.389000+00:00",
"lastUpdateTime": "2023-02-28T20:38:01.389000+00:00",
"tags": {
  "aws:servicecatalog:attributeGroupName": "ExampleAttributeGroup"
}
}
```

Para obtener más información, consulte [Administración de metadatos para grupos de atributos](#) en la Guía del AppRegistry administrador de AWS Service Catalog.

- Para API obtener más información, consulte [GetAttributeGroup](#) la Referencia de AWS CLI comandos.

## list-applications

En el siguiente ejemplo de código, se muestra cómo usar `list-applications`.

### AWS CLI

Para enumerar las aplicaciones

En el siguiente `list-applications` ejemplo, se recupera una lista de todas las aplicaciones de su AWS cuenta.

```
aws servicecatalog-appregistry list-applications
```

Salida:

```
{
  "applications": [
    {
      "id": "03axw94pjfj3uan00tcgbrxnkw",
      "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/03axw94pjfj3uan00tcgbrxnkw",
      "name": "ExampleApplication2",
      "creationTime": "2023-02-28T21:59:34.094000+00:00",
      "lastUpdateTime": "2023-02-28T21:59:34.094000+00:00"
    },
    {
      "id": "055gw7aynr1i5mbv7kjwzx5945",
      "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/055gw7aynr1i5mbv7kjwzx5945",
```

```

        "name": "ExampleApplication3",
        "creationTime": "2023-02-28T22:06:28.228000+00:00",
        "lastUpdateTime": "2023-02-28T22:06:28.228000+00:00"
    },
    {
        "id": "0ars38r6btoohvpvd9gqrptt91",
        "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/0ars38r6btoohvpvd9gqrptt91",
        "name": "ExampleApplication",
        "description": "This is an example application",
        "creationTime": "2023-02-28T21:10:10.820000+00:00",
        "lastUpdateTime": "2023-02-28T21:24:19.729000+00:00"
    }
]
}

```

Para obtener más información, consulte [Visualización de los detalles de la aplicación](#) en la Guía del AppRegistry administrador de AWS Service Catalog.

- Para API obtener más información, consulte [ListApplications](#) la Referencia de AWS CLI comandos.

## list-associated-attribute-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-associated-attribute-groups`.

### AWS CLI

Para enumerar los grupos de atributos asociados

El siguiente `list-associated-attribute-groups` ejemplo recupera una lista de todos los grupos de atributos de su AWS cuenta que están asociados a una aplicación específica de su AWS cuenta.

```

aws servicecatalog-appregistry list-associated-attribute-groups \
  --application "ExampleApplication"

```

Salida:

```

{
  "attributeGroups": [

```

```

    "01sj5xdwhbw54kejwnt09fnpc1"
  ]
}

```

Para obtener más información, consulte [Asociar y desasociar grupos de atributos](#) en la Guía del administrador de AWS Service Catalog AppRegistry .

- Para API obtener más información, consulte la Referencia de [ListAssociatedAttributeGroups](#) comandos AWS CLI .

## list-attribute-groups-for-application

En el siguiente ejemplo de código, se muestra cómo usar `list-attribute-groups-for-application`.

### AWS CLI

Para enumerar los grupos de atributos de una aplicación

En el siguiente `list-attribute-groups-for-application` ejemplo, se enumeran los detalles de todos los grupos de atributos de su AWS cuenta que están asociados a una aplicación específica de su AWS cuenta.

```

aws servicecatalog-appregistry list-attribute-groups-for-application \
  --application "ExampleApplication"

```

Salida:

```

{
  "attributeGroupsDetails": [
    {
      "id": "01sj5xdwhbw54kejwnt09fnpc1",
      "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/01sj5xdwhbw54kejwnt09fnpc1",
      "name": "ExampleAttributeGroup"
    }
  ]
}

```

Para obtener más información, consulte [Visualización de los detalles de los grupos de atributos](#) en la Guía del AppRegistry administrador de AWS Service Catalog.

- Para API obtener más información, consulte [ListAttributeGroupsForApplication](#) la Referencia de AWS CLI comandos.

## list-attribute-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-attribute-groups`.

### AWS CLI

Para enumerar los grupos de atributos

En el siguiente `list-attribute-groups` ejemplo, se recupera una lista de todos los grupos de atributos de su AWS cuenta.

```
aws servicecatalog-appregistry list-attribute-groups
```

Salida:

```
{
  "attributeGroups": [
    {
      "id": "011ge6y3emyjijt8dw8jn6r0hv",
      "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/011ge6y3emyjijt8dw8jn6r0hv",
      "name": "ExampleAttributeGroup3",
      "creationTime": "2023-02-28T22:05:35.224000+00:00",
      "lastUpdateTime": "2023-02-28T22:05:35.224000+00:00"
    },
    {
      "id": "01sj5xdwhbw54kejwnt09fnpc1",
      "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/01sj5xdwhbw54kejwnt09fnpc1",
      "name": "ExampleAttributeGroup",
      "description": "This is an example attribute group",
      "creationTime": "2023-02-28T20:38:01.389000+00:00",
      "lastUpdateTime": "2023-02-28T21:02:04.559000+00:00"
    },
    {
      "id": "03n1yffgq6d18vwrzxf0c70nm3",
      "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/03n1yffgq6d18vwrzxf0c70nm3",
      "name": "ExampleAttributeGroup2",
```

```
        "creationTime": "2023-02-28T21:57:30.687000+00:00",
        "lastUpdateTime": "2023-02-28T21:57:30.687000+00:00"
    }
]
}
```

Para obtener más información, consulte [Visualización de los detalles de los grupos de atributos](#) en la Guía del AppRegistry administrador de AWS Service Catalog.

- Para API obtener más información, consulte [ListAttributeGroups](#) la Referencia de AWS CLI comandos.

## update-application

En el siguiente ejemplo de código, se muestra cómo usar `update-application`.

### AWS CLI

Para actualizar una aplicación

En el siguiente `update-application` ejemplo, se actualiza una aplicación específica de su AWS cuenta para incluir una descripción.

```
aws servicecatalog-appregistry update-application \
  --application "ExampleApplication" \
  --description "This is an example application"
```

Salida:

```
{
  "application": {
    "id": "0ars38r6btoohvpvd9gqrptt91",
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/0ars38r6btoohvpvd9gqrptt91",
    "name": "ExampleApplication",
    "description": "This is an example application",
    "creationTime": "2023-02-28T21:10:10.820000+00:00",
    "lastUpdateTime": "2023-02-28T21:24:19.729000+00:00",
    "tags": {
      "aws:servicecatalog:applicationName": "ExampleApplication"
    }
  }
}
```

```
}
```

Para obtener más información, consulte [Edición de aplicaciones](#) en la Guía del AppRegistry administrador de AWS Service Catalog.

- Para API obtener más información, consulte [UpdateApplication](#) la Referencia de AWS CLI comandos.

## update-attribute-group

En el siguiente ejemplo de código, se muestra cómo usar `update-attribute-group`.

### AWS CLI

Para actualizar un grupo de atributos

En el siguiente `update-attribute-group` ejemplo, se actualiza un grupo de atributos específico de su AWS cuenta para incluir una descripción.

```
aws servicecatalog-appregistry update-attribute-group \  
  --attribute-group "ExampleAttributeGroup" \  
  --description "This is an example attribute group"
```

Salida:

```
{  
  "attributeGroup": {  
    "id": "01sj5xdwhbw54kejwnt09fnpc1",  
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-  
groups/01sj5xdwhbw54kejwnt09fnpc1",  
    "name": "ExampleAttributeGroup",  
    "description": "This is an example attribute group",  
    "creationTime": "2023-02-28T20:38:01.389000+00:00",  
    "lastUpdateTime": "2023-02-28T21:02:04.559000+00:00",  
    "tags": {  
      "aws:servicecatalog:attributeGroupName": "ExampleAttributeGroup"  
    }  
  }  
}
```

Para obtener más información, consulte [Edición de grupos de atributos](#) en la Guía del AppRegistry administrador de AWS Service Catalog.

- Para API obtener más información, consulte [UpdateAttributeGroup](#) la Referencia de AWS CLI comandos.

## Ejemplos de Athena usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface Athena.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **batch-get-named-query**

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-named-query`.

AWS CLI

Para devolver información sobre más de una consulta

En el siguiente `batch-get-named-query` ejemplo, se devuelve información sobre las consultas con nombre que tienen los valores especificados IDs.

```
aws athena batch-get-named-query \
  --named-query-ids a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 a1b2c3d4-5678-90ab-cdef-EXAMPLE22222 a1b2c3d4-5678-90ab-cdef-EXAMPLE33333
```

Salida:

```
{
  "NamedQueries": [
```

```

    {
      "Name": "Flights Select Query",
      "Description": "Sample query to get the top 10 airports with the most
number of departures since 2000",
      "Database": "sampledb",
      "QueryString": "SELECT origin, count(*) AS total_departures\nFROM
\nflights_parquet\nWHERE year >= '2000'\nGROUP BY origin\nORDER BY total_departures
DESC\nLIMIT 10;",
      "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "WorkGroup": "primary"
    },
    {
      "Name": "Load flights table partitions",
      "Description": "Sample query to load flights table partitions using MSCK
REPAIR TABLE statement",
      "Database": "sampledb",
      "QueryString": "MSCK REPAIR TABLE flights_parquet;",
      "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "WorkGroup": "primary"
    },
    {
      "Name": "CloudFront Select Query",
      "Description": "Sample query to view requests per operating system
during a particular time frame",
      "Database": "sampledb",
      "QueryString": "SELECT os, COUNT(*) count FROM cloudfront_logs WHERE
date BETWEEN date '2014-07-05' AND date '2014-08-05' GROUP BY os;",
      "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
      "WorkGroup": "primary"
    }
  ],
  "UnprocessedNamedQueryIds": []
}

```

Para obtener más información, consulte [Ejecución de SQL consultas con Amazon Athena](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [BatchGetNamedQuery](#) de AWS CLI comandos.

## batch-get-query-execution

En el siguiente ejemplo de código, se muestra cómo usar batch-get-query-execution.



## AWS CLI

Para devolver información sobre una o más ejecuciones de consultas

En el siguiente `batch-get-query-execution` ejemplo, se devuelve la información de ejecución de las consultas que tienen la consulta especificada IDs.

```
aws athena batch-get-query-execution \  
  --query-execution-ids a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 a1b2c3d4-5678-90ab-  
cdef-EXAMPLE22222
```

Salida:

```
{  
  "QueryExecutions": [  
    {  
      "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "Query": "create database if not exists webdata",  
      "StatementType": "DDL",  
      "ResultConfiguration": {  
        "OutputLocation": "s3://awsdoc-example-bucket/a1b2c3d4-5678-90ab-  
cdef-EXAMPLE11111.txt"  
      },  
      "QueryExecutionContext": {},  
      "Status": {  
        "State": "SUCCEEDED",  
        "SubmissionDateTime": 1593470720.592,  
        "CompletionDateTime": 1593470720.902  
      },  
      "Statistics": {  
        "EngineExecutionTimeInMillis": 232,  
        "DataScannedInBytes": 0,  
        "TotalExecutionTimeInMillis": 310,  
        "ResultConfiguration": {  
          "QueryQueueTimeInMillis": 50,  
          "ServiceProcessingTimeInMillis": 28  
        },  
        "WorkGroup": "AthenaAdmin"  
      },  
      {  
        "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
```

```

    "Query": "select date, location, browser, uri, status from
cloudfront_logs where method = 'GET' and status = 200 and location like 'SF0%'
limit 10",
    "StatementType": "DML",
    "ResultConfiguration": {
        "OutputLocation": "s3://awsdoc-example-bucket/a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222.csv"
    },
    "QueryExecutionContext": {
        "Database": "mydatabase",
        "Catalog": "awsdatacatalog"
    },
    "Status": {
        "State": "SUCCEEDED",
        "SubmissionDateTime": 1593469842.665,
        "CompletionDateTime": 1593469846.486
    },
    "Statistics": {
        "EngineExecutionTimeInMillis": 3600,
        "DataScannedInBytes": 203089,
        "TotalExecutionTimeInMillis": 3821,
        "QueryQueueTimeInMillis": 267,
        "QueryPlanningTimeInMillis": 1175
    },
    "WorkGroup": "AthenaAdmin"
}
],
"UnprocessedQueryExecutionIds": []
}

```

Para obtener más información, consulte [Ejecución de SQL consultas con Amazon Athena](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [BatchGetQueryExecution](#) de AWS CLI comandos.

## create-data-catalog

En el siguiente ejemplo de código, se muestra cómo usar create-data-catalog.

### AWS CLI

Para crear un catálogo de datos

En el siguiente `create-data-catalog` ejemplo, se crea el catálogo de `dynamo_db_catalog` datos.

```
aws athena create-data-catalog \  
  --name dynamo_db_catalog \  
  --type LAMBDA \  
  --description "DynamoDB Catalog" \  
  --parameters function=arn:aws:lambda:us-west-2:111122223333:function:dynamo_db_lambda
```

Este comando no genera ninguna salida. Para ver el resultado, utilice `aws athena get-data-catalog --name dynamo_db_catalog`.

Para obtener más información, consulte [Registrar un catálogo: create-data-catalog](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte [CreateDataCatalog](#) la Referencia de AWS CLI comandos.

## create-named-query

En el siguiente ejemplo de código, se muestra cómo usar `create-named-query`.

### AWS CLI

Para crear una consulta con nombre

En el siguiente `create-named-query` ejemplo, se crea una consulta guardada en el AthenaAdmin grupo de trabajo que consulta la `flights_parquet` tabla sobre los vuelos de Seattle a Nueva York en enero de 2016 cuya salida y llegada se retrasaron más de diez minutos. Como los valores de los códigos de aeropuerto de la tabla son cadenas que incluyen comillas dobles (por ejemplo, "SEA«), aparecen separados por barras invertidas y entre comillas simples.

```
aws athena create-named-query \  
  --name "SEA to JFK delayed flights Jan 2016" \  
  --description "Both arrival and departure delayed more than 10 minutes." \  
  --database sampledb \  
  --query-string "SELECT flightdate, carrier, flightnum, origin, dest, depdelayminutes, arrdelayminutes FROM sampledb.flights_parquet WHERE yr = 2016 AND month = 1 AND origin = \"SEA\" AND dest = \"JFK\" AND depdelayminutes > 10 AND arrdelayminutes > 10" \  
  --
```

```
--work-group AthenaAdmin
```

Salida:

```
{
  "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Para obtener más información, consulte [Ejecución de SQL consultas con Amazon Athena](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [CreateNamedQuery](#) de AWS CLI comandos.

## create-work-group

En el siguiente ejemplo de código, se muestra cómo usar `create-work-group`.

### AWS CLI

Para crear un grupo de trabajo

En el siguiente `create-work-group` ejemplo, se crea un grupo de trabajo denominado `Data_Analyst_Group` que tiene la ubicación de salida de los resultados de la consulta. `s3://awsdoc-example-bucket` El comando crea un grupo de trabajo que anula los ajustes de configuración del cliente, que incluyen la ubicación de salida de los resultados de la consulta. El comando también habilita CloudWatch las métricas y agrega tres pares de etiquetas clave-valor al grupo de trabajo para distinguirlo de otros grupos de trabajo. Tenga en cuenta que el `--configuration` argumento no tiene espacios antes de las comas que separan sus opciones.

```
aws athena create-work-group \
  --name Data_Analyst_Group \
  --configuration ResultConfiguration={OutputLocation="s3://awsdoc-example-
bucket"},EnforceWorkGroupConfiguration="true",PublishCloudWatchMetricsEnabled="true" \
  --description "Workgroup for data analysts" \
  --tags Key=Division,Value=West Key=Location,Value=Seattle Key=Team,Value="Big
Data"
```

Este comando no genera ninguna salida. Para ver los resultados, utilice `aws athena get-work-group --work-group Data_Analyst_Group`.

Para obtener más información, consulte [Administración de grupos de trabajo](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [CreateWorkGroup](#) de AWS CLI comandos.

## delete-data-catalog

En el siguiente ejemplo de código, se muestra cómo usar `delete-data-catalog`.

### AWS CLI

Para eliminar un catálogo de datos

En el siguiente `delete-data-catalog` ejemplo, se elimina el catálogo `UnusedDataCatalog` de datos.

```
aws athena delete-data-catalog \  
  --name UnusedDataCatalog
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un catálogo: delete-data-catalog](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte [DeleteDataCatalog](#) la Referencia de AWS CLI comandos.

## delete-named-query

En el siguiente ejemplo de código, se muestra cómo usar `delete-named-query`.

### AWS CLI

Para eliminar una consulta con nombre

En el siguiente `delete-named-query` ejemplo, se elimina la consulta con nombre asignado que tiene el identificador especificado.

```
aws athena delete-named-query \  
  --named-query-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Ejecución de SQL consultas con Amazon Athena](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [DeleteNamedQuery](#) de AWS CLI comandos.

## **delete-work-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-work-group`.

AWS CLI

Para eliminar un grupo de trabajo

En el siguiente `delete-work-group` ejemplo, se elimina el `TeamB` grupo de trabajo.

```
aws athena delete-work-group \  
  --work-group TeamB
```

Este comando no genera ninguna salida. Para confirmar la eliminación, utilice `aws athena list-work-groups`

Para obtener más información, consulte [Administración de grupos de trabajo](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [DeleteWorkGroup](#) de AWS CLI comandos.

## **get-data-catalog**

En el siguiente ejemplo de código, se muestra cómo usar `get-data-catalog`.

AWS CLI

Para devolver información sobre un catálogo de datos

El siguiente `get-data-catalog` ejemplo devuelve información sobre el catálogo `dynamo_db_catalog` de datos.

```
aws athena get-data-catalog \  
  --catalog-name dynamo_db_catalog
```

```
--name dynamo_db_catalog
```

Salida:

```
{
  "DataCatalog": {
    "Name": "dynamo_db_catalog",
    "Description": "DynamoDB Catalog",
    "Type": "LAMBDA",
    "Parameters": {
      "catalog": "dynamo_db_catalog",
      "metadata-function": "arn:aws:lambda:us-
west-2:111122223333:function:dynamo_db_lambda",
      "record-function": "arn:aws:lambda:us-
west-2:111122223333:function:dynamo_db_lambda"
    }
  }
}
```

Para obtener más información, consulte [Mostrar detalles del catálogo: get-data-catalog](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte [GetDataCatalog](#) la Referencia de AWS CLI comandos.

## get-database

En el siguiente ejemplo de código, se muestra cómo usar `get-database`.

AWS CLI

Para devolver información sobre una base de datos de un catálogo de datos

El siguiente `get-database` ejemplo devuelve información sobre la `samp1edb` base de datos del catálogo `AwsDataCatalog` de datos.

```
aws athena get-database \  
  --catalog-name AwsDataCatalog \  
  --database-name samp1edb
```

Salida:

```
{
  "Database": {
    "Name": "sampledb",
    "Description": "Sample database",
    "Parameters": {
      "CreatedBy": "Athena",
      "EXTERNAL": "TRUE"
    }
  }
}
```

Para obtener más información, consulte [Mostrar detalles de la base de datos: get-database](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia de [GetDatabase](#) comandos AWS CLI .

## get-named-query

En el siguiente ejemplo de código, se muestra cómo usar `get-named-query`.

### AWS CLI

Para devolver una consulta con nombre

En el siguiente `get-named-query` ejemplo, se devuelve información sobre la consulta que tiene el identificador especificado.

```
aws athena get-named-query \
  --named-query-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{
  "NamedQuery": {
    "Name": "CloudFront Logs - SF0",
    "Description": "Shows successful GET request data for SF0",
    "Database": "default",
    "QueryString": "select date, location, browser, uri, status from
cloudfront_logs where method = 'GET' and status = 200 and location like 'SF0%'
limit 10",
    "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
```



```

    "WorkGroup": "AthenaAdmin"
  }
}

```

Para obtener más información, consulte [Ejecución de SQL consultas con Amazon Athena](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [GetNamedQuery](#) de AWS CLI comandos.

## get-query-execution

En el siguiente ejemplo de código, se muestra cómo usar `get-query-execution`.

### AWS CLI

Para devolver información sobre la ejecución de una consulta

El siguiente `get-query-execution` ejemplo devuelve información sobre la consulta que tiene el identificador de consulta especificado.

```

aws athena get-query-execution \
  --query-execution-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

```

Salida:

```

{
  "QueryExecution": {
    "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Query": "select date, location, browser, uri, status from cloudfront_logs
where method = 'GET
' and status = 200 and location like 'SF0%' limit 10",
    "StatementType": "DML",
    "ResultConfiguration": {
      "OutputLocation": "s3://awsdoc-example-bucket/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111.csv"
    },
    "QueryExecutionContext": {
      "Database": "mydatabase",
      "Catalog": "awsdatacatalog"
    },
    "Status": {
      "State": "SUCCEEDED",

```

```

        "SubmissionDateTime": 1593469842.665,
        "CompletionDateTime": 1593469846.486
    },
    "Statistics": {
        "EngineExecutionTimeInMillis": 3600,
        "DataScannedInBytes": 203089,
        "TotalExecutionTimeInMillis": 3821,
        "QueryQueueTimeInMillis": 267,
        "QueryPlanningTimeInMillis": 1175
    },
    "WorkGroup": "AthenaAdmin"
}
}

```

Para obtener más información, consulte [Ejecución de SQL consultas con Amazon Athena](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [GetQueryExecution](#) de AWS CLI comandos.

## get-query-results

En el siguiente ejemplo de código, se muestra cómo usar `get-query-results`.

### AWS CLI

Para devolver los resultados de una consulta

El siguiente `get-query-results` ejemplo devuelve los resultados de la consulta que tiene el identificador de consulta especificado.

```

aws athena get-query-results \
  --query-execution-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

```

Salida:

```

{
  "ResultSet": {
    "Rows": [
      {
        "Data": [
          {
            "VarCharValue": "date"
          }
        ]
      }
    ]
  }
}

```

```
    },
    {
      "VarCharValue": "location"
    },
    {
      "VarCharValue": "browser"
    },
    {
      "VarCharValue": "uri"
    },
    {
      "VarCharValue": "status"
    }
  ]
},
{
  "Data": [
    {
      "VarCharValue": "2014-07-05"
    },
    {
      "VarCharValue": "SF04"
    },
    {
      "VarCharValue": "Safari"
    },
    {
      "VarCharValue": "/test-image-2.jpeg"
    },
    {
      "VarCharValue": "200"
    }
  ]
},
{
  "Data": [
    {
      "VarCharValue": "2014-07-05"
    },
    {
      "VarCharValue": "SF04"
    },
    {
      "VarCharValue": "Opera"
    }
  ]
}
```

```
    },
    {
      "VarCharValue": "/test-image-2.jpeg"
    },
    {
      "VarCharValue": "200"
    }
  ]
},
{
  "Data": [
    {
      "VarCharValue": "2014-07-05"
    },
    {
      "VarCharValue": "SF04"
    },
    {
      "VarCharValue": "Firefox"
    },
    {
      "VarCharValue": "/test-image-3.jpeg"
    },
    {
      "VarCharValue": "200"
    }
  ]
},
{
  "Data": [
    {
      "VarCharValue": "2014-07-05"
    },
    {
      "VarCharValue": "SF04"
    },
    {
      "VarCharValue": "Lynx"
    },
    {
      "VarCharValue": "/test-image-3.jpeg"
    },
    {
      "VarCharValue": "200"
    }
  ]
}
```

```
    }
  ]
},
{
  "Data": [
    {
      "VarCharValue": "2014-07-05"
    },
    {
      "VarCharValue": "SF04"
    },
    {
      "VarCharValue": "IE"
    },
    {
      "VarCharValue": "/test-image-2.jpeg"
    },
    {
      "VarCharValue": "200"
    }
  ]
},
{
  "Data": [
    {
      "VarCharValue": "2014-07-05"
    },
    {
      "VarCharValue": "SF04"
    },
    {
      "VarCharValue": "Opera"
    },
    {
      "VarCharValue": "/test-image-1.jpeg"
    },
    {
      "VarCharValue": "200"
    }
  ]
},
{
  "Data": [
    {
```

```
        "VarCharValue": "2014-07-05"
      },
      {
        "VarCharValue": "SF04"
      },
      {
        "VarCharValue": "Chrome"
      },
      {
        "VarCharValue": "/test-image-3.jpeg"
      },
      {
        "VarCharValue": "200"
      }
    ]
  },
  {
    "Data": [
      {
        "VarCharValue": "2014-07-05"
      },
      {
        "VarCharValue": "SF04"
      },
      {
        "VarCharValue": "Firefox"
      },
      {
        "VarCharValue": "/test-image-2.jpeg"
      },
      {
        "VarCharValue": "200"
      }
    ]
  },
  {
    "Data": [
      {
        "VarCharValue": "2014-07-05"
      },
      {
        "VarCharValue": "SF04"
      },
```

```
        "VarCharValue": "Chrome"
      },
      {
        "VarCharValue": "/test-image-3.jpeg"
      },
      {
        "VarCharValue": "200"
      }
    ]
  },
  {
    "Data": [
      {
        "VarCharValue": "2014-07-05"
      },
      {
        "VarCharValue": "SF04"
      },
      {
        "VarCharValue": "IE"
      },
      {
        "VarCharValue": "/test-image-2.jpeg"
      },
      {
        "VarCharValue": "200"
      }
    ]
  }
],
"ResultSetMetadata": {
  "ColumnInfo": [
    {
      "CatalogName": "hive",
      "SchemaName": "",
      "TableName": "",
      "Name": "date",
      "Label": "date",
      "Type": "date",
      "Precision": 0,
      "Scale": 0,
      "Nullable": "UNKNOWN",
      "CaseSensitive": false
    },

```

```
{
  "CatalogName": "hive",
  "SchemaName": "",
  "TableName": "",
  "Name": "location",
  "Label": "location",
  "Type": "varchar",
  "Precision": 2147483647,
  "Data": [
    {
      "Scale": 0,
      "Nullable": "UNKNOWN",
      "CaseSensitive": true
    },
    {
      "CatalogName": "hive",
      "SchemaName": "",
      "TableName": "",
      "Name": "browser",
      "Label": "browser",
      "Type": "varchar",
      "Precision": 2147483647,
      "Scale": 0,
      "Nullable": "UNKNOWN",
      "CaseSensitive": true
    },
    {
      "CatalogName": "hive",
      "SchemaName": "",
      "TableName": "",
      "Name": "uri",
      "Label": "uri",
      "Type": "varchar",
      "Precision": 2147483647,
      "Scale": 0,
      "Nullable": "UNKNOWN",
      "CaseSensitive": true
    },
    {
      "CatalogName": "hive",
      "SchemaName": "",
      "TableName": "",
      "Name": "status",
      "Label": "status",
```



```

        "Type": "integer",
        "Precision": 10,
        "Scale": 0,
        "Nullable": "UNKNOWN",
        "CaseSensitive": false
      }
    ]
  },
  "UpdateCount": 0
}

```

Para obtener más información, consulte [Trabajar con los resultados de consultas, los archivos de salida y el historial de consultas](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte [GetQueryResults](#) la Referencia de AWS CLI comandos.

## get-table-metadata

En el siguiente ejemplo de código, se muestra cómo usar `get-table-metadata`.

### AWS CLI

Para devolver información de metadatos sobre una tabla

El siguiente `get-table-metadata` ejemplo devuelve información de metadatos sobre la `counties` tabla, incluidos los nombres de las columnas y sus tipos de datos, de la `sampledb` base de datos del catálogo de datos. `AwsDataCatalog`

```

aws athena get-table-metadata \
  --catalog-name AwsDataCatalog \
  --database-name sampledb \
  --table-name counties

```

Salida:

```

{
  "TableMetadata": {
    "Name": "counties",
    "CreateTime": 1593559968.0,

```

```
"LastAccessTime": 0.0,
"TableType": "EXTERNAL_TABLE",
"Columns": [
  {
    "Name": "name",
    "Type": "string",
    "Comment": "from deserializer"
  },
  {
    "Name": "boundaryshape",
    "Type": "binary",
    "Comment": "from deserializer"
  },
  {
    "Name": "motto",
    "Type": "string",
    "Comment": "from deserializer"
  },
  {
    "Name": "population",
    "Type": "int",
    "Comment": "from deserializer"
  }
],
"PartitionKeys": [],
"Parameters": {
  "EXTERNAL": "TRUE",
  "inputformat": "com.esri.json.hadoop.EnclosedJsonInputFormat",
  "location": "s3://awsdoc-example-bucket/json",
  "outputformat":
"org.apache.hadoop.hive ql.io.HiveIgnoreKeyTextOutputFormat",
  "serde.param.serialization.format": "1",
  "serde.serialization.lib": "com.esri.hadoop.hive.serde.JsonSerde",
  "transient_lastDdlTime": "1593559968"
}
}
```

Para obtener más información, consulte [Mostrar detalles de la tabla: get-table-metadata](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte [GetTableMetadata](#) la Referencia de AWS CLI comandos.

## get-work-group

En el siguiente ejemplo de código, se muestra cómo usar `get-work-group`.

### AWS CLI

Para devolver información sobre un grupo de trabajo

En el siguiente `get-work-group` ejemplo, se devuelve información sobre el grupo de AthenaAdmin trabajo.

```
aws athena get-work-group \  
--work-group AthenaAdmin
```

Salida:

```
{  
  "WorkGroup": {  
    "Name": "AthenaAdmin",  
    "State": "ENABLED",  
    "Configuration": {  
      "ResultConfiguration": {  
        "OutputLocation": "s3://awsdoc-example-bucket/"  
      },  
      "EnforceWorkGroupConfiguration": false,  
      "PublishCloudWatchMetricsEnabled": true,  
      "RequesterPaysEnabled": false  
    },  
    "Description": "Workgroup for Athena administrators",  
    "CreationTime": 1573677174.105  
  }  
}
```

Para obtener más información, consulte [Administración de grupos de trabajo](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [GetWorkGroup](#) de AWS CLI comandos.

## list-data-catalogs

En el siguiente ejemplo de código, se muestra cómo usar `list-data-catalogs`.

## AWS CLI

Para enumerar los catálogos de datos registrados en Athena

En el siguiente `list-data-catalogs` ejemplo, se enumeran los catálogos de datos registrados en Athena.

```
aws athena list-data-catalogs
```

Salida:

```
{
  "DataCatalogsSummary": [
    {
      "CatalogName": "AwsDataCatalog",
      "Type": "GLUE"
    },
    {
      "CatalogName": "cw_logs_catalog",
      "Type": "LAMBDA"
    },
    {
      "CatalogName": "cw_metrics_catalog",
      "Type": "LAMBDA"
    }
  ]
}
```

Para obtener más información, consulte [Publicar catálogos registrados: list-data-catalogs](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [ListDataCatalogs](#) de AWS CLI comandos.

## list-databases

En el siguiente ejemplo de código, se muestra cómo usar `list-databases`.

## AWS CLI

Para enumerar las bases de datos de un catálogo de datos

En el siguiente `list-databases` ejemplo, se enumeran las bases de `AwsDataCatalog` datos del catálogo de datos.

```
aws athena list-databases \  
  --catalog-name AwsDataCatalog
```

Salida:

```
{  
  "DatabaseList": [  
    {  
      "Name": "default"  
    },  
    {  
      "Name": "mydatabase"  
    },  
    {  
      "Name": "newdb"  
    },  
    {  
      "Name": "sampledb",  
      "Description": "Sample database",  
      "Parameters": {  
        "CreatedBy": "Athena",  
        "EXTERNAL": "TRUE"  
      }  
    },  
    {  
      "Name": "webdata"  
    }  
  ]  
}
```

Para obtener más información, consulte [Incluir bases de datos en un catálogo: bases de datos de listas](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia de [ListDatabases](#) comandos AWS CLI .

## **list-named-queries**

En el siguiente ejemplo de código, se muestra cómo usar `list-named-queries`.

## AWS CLI

Para enumerar las consultas con nombre asignado a un grupo de trabajo

En el siguiente `list-named-queries` ejemplo, se enumeran las consultas con nombre del grupo de AthenaAdmin trabajo.

```
aws athena list-named-queries \  
  --work-group AthenaAdmin
```

Salida:

```
{  
  "NamedQueryIds": [  
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333"  
  ]  
}
```

Para obtener más información, consulte [Ejecución de SQL consultas con Amazon Athena](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [ListNamedQueries](#) de AWS CLI comandos.

## **list-query-executions**

En el siguiente ejemplo de código, se muestra cómo usar `list-query-executions`.

### AWS CLI

Para enumerar la consulta IDs de las consultas de un grupo de trabajo específico

En el siguiente `list-query-executions` ejemplo, se muestra un máximo de diez consultas del IDs grupo de AthenaAdmin trabajo.

```
aws athena list-query-executions \  
  --work-group AthenaAdmin \  
  --max-items 10
```

Salida:

```
{
  "QueryExecutionIds": [
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11110",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11114",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11115",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11116",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11117",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11118",
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11119"
  ],
  "NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxMH0="
}
```

Para obtener más información, consulte [Trabajar con los resultados de consultas, los archivos de salida y el historial de consultas](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte [ListQueryExecutions](#) la Referencia de AWS CLI comandos.

## list-table-metadata

En el siguiente ejemplo de código, se muestra cómo usar `list-table-metadata`.

### AWS CLI

Para enumerar los metadatos de las tablas de la base de datos especificada de un catálogo de datos

El siguiente `list-table-metadata` ejemplo devuelve la información de metadatos de un máximo de dos tablas de la `geography` base de datos del catálogo de `AwsDataCatalog` datos.

```
aws athena list-table-metadata \
  --catalog-name AwsDataCatalog \
  --database-name geography \
  --max-items 2
```

Salida:

```
{
```

```
"TableMetadataList": [
  {
    "Name": "country_codes",
    "CreateTime": 1586553454.0,
    "TableType": "EXTERNAL_TABLE",
    "Columns": [
      {
        "Name": "country",
        "Type": "string",
        "Comment": "geo id"
      },
      {
        "Name": "alpha-2 code",
        "Type": "string",
        "Comment": "geo id2"
      },
      {
        "Name": "alpha-3 code",
        "Type": "string",
        "Comment": "state name"
      },
      {
        "Name": "numeric code",
        "Type": "bigint",
        "Comment": ""
      },
      {
        "Name": "latitude",
        "Type": "bigint",
        "Comment": "location (latitude)"
      },
      {
        "Name": "longitude",
        "Type": "bigint",
        "Comment": "location (longitude)"
      }
    ],
    "Parameters": {
      "areColumnsQuoted": "false",
      "classification": "csv",
      "columnsOrdered": "true",
      "delimiter": ",",
      "has_encrypted_data": "false",
      "inputformat": "org.apache.hadoop.mapred.TextInputFormat",
    }
  }
]
```



```

        "location": "s3://awsdoc-example-bucket/csv/countrycode",
        "outputformat":
"org.apache.hadoop.hive ql.io.HiveIgnoreKeyTextOutputFormat",
        "serde.param.field.delim": ",",
        "serde.serialization.lib":
"org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe",
        "skip.header.line.count": "1",
        "typeOfData": "file"
    }
},
{
    "Name": "county_populations",
    "CreateTime": 1586553446.0,
    "TableType": "EXTERNAL_TABLE",
    "Columns": [
        {
            "Name": "id",
            "Type": "string",
            "Comment": "geo id"
        },
        {
            "Name": "country",

            "Name": "id2",
            "Type": "string",
            "Comment": "geo id2"
        },
        {
            "Name": "county",
            "Type": "string",
            "Comment": "county name"
        },
        {
            "Name": "state",
            "Type": "string",
            "Comment": "state name"
        },
        {
            "Name": "population estimate 2018",
            "Type": "string",
            "Comment": ""
        }
    ],
    "Parameters": {

```

```

        "areColumnsQuoted": "false",
        "classification": "csv",
        "columnsOrdered": "true",
        "delimiter": ",",
        "has_encrypted_data": "false",
        "inputformat": "org.apache.hadoop.mapred.TextInputFormat",
        "location": "s3://awsdoc-example-bucket/csv/CountyPopulation",
        "outputformat":
"org.apache.hadoop.hive.q1.io.HiveIgnoreKeyTextOutputFormat",
        "serde.param.field.delim": ",",
        "serde.serialization.lib":
"org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe",
        "skip.header.line.count": "1",
        "typeOfData": "file"
    }
}
],
"NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}

```

Para obtener más información, consulte [Mostrar los metadatos de todas las tablas de una base de datos: list-table-metadata](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte [ListTableMetadata](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Ejemplo 1: Para enumerar las etiquetas de un grupo de trabajo

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas del grupo de `Data_Analyst_Group` trabajo.

```

aws athena list-tags-for-resource \
  --resource-arn arn:aws:athena:us-west-2:111122223333:workgroup/
Data_Analyst_Group

```

Salida:

```
{
  "Tags": [
    {
      "Key": "Division",
      "Value": "West"
    },
    {
      "Key": "Team",
      "Value": "Big Data"
    },
    {
      "Key": "Location",
      "Value": "Seattle"
    }
  ]
}
```

Ejemplo 2: Para enumerar las etiquetas de un catálogo de datos

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas del catálogo `dynamo_db_catalog` de datos.

```
aws athena list-tags-for-resource \
  --resource-arn arn:aws:athena:us-west-2:111122223333:datacatalog/
dynamo_db_catalog
```

Salida:

```
{
  "Tags": [
    {
      "Key": "Division",
      "Value": "Mountain"
    },
    {
      "Key": "Organization",
      "Value": "Retail"
    },
    {
      "Key": "Product_Line",
      "Value": "Shoes"
    },
  ],
}
```

```
{
  "Key": "Location",
  "Value": "Denver"
}
]
```

Para obtener más información, consulte [Enumerar las etiquetas de un recurso: list-tags-for-resource](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## list-work-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-work-groups`.

### AWS CLI

Para enumerar los grupos de trabajo

En el siguiente `list-work-groups` ejemplo, se enumeran los grupos de trabajo de la cuenta corriente.

```
aws athena list-work-groups
```

Salida:

```
{
  "WorkGroups": [
    {
      "Name": "Data_Analyst_Group",
      "State": "ENABLED",
      "Description": "",
      "CreationTime": 1578006683.016
    },
    {
      "Name": "AthenaAdmin",
      "State": "ENABLED",
      "Description": "",
      "CreationTime": 1573677174.105
    },
    {
```

```

        "Name": "primary",
        "State": "ENABLED",
        "Description": "",
        "CreationTime": 1567465222.723
    }
]
}

```

Para obtener más información, consulte [Administración de grupos de trabajo](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [ListWorkGroups](#) de AWS CLI comandos.

## start-query-execution

En el siguiente ejemplo de código, se muestra cómo usar `start-query-execution`.

### AWS CLI

Ejemplo 1: Para ejecutar una consulta en un grupo de trabajo en la tabla especificada de la base de datos y el catálogo de datos especificados

En el siguiente `start-query-execution` ejemplo, se utiliza el `AthenaAdmin` grupo de trabajo para ejecutar una consulta en la `cloudfront_logs` tabla del `cflogsdatabase` catálogo de `AwsDataCatalog` datos.

```

aws athena start-query-execution \
  --query-string "select date, location, browser, uri, status from cloudfront_logs
  where method = 'GET' and status = 200 and location like 'SF0%' limit 10" \
  --work-group "AthenaAdmin" \
  --query-execution-context Database=cflogsdatabase,Catalog=AwsDataCatalog

```

Salida:

```

{
  "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}

```

Para obtener más información, consulte [Ejecución de SQL consultas con Amazon Athena](#) en la Guía del usuario de Amazon Athena.

Ejemplo 2: Para ejecutar una consulta que utilice un grupo de trabajo específico para crear una base de datos en el catálogo de datos especificado

En el siguiente `start-query-execution` ejemplo, se utiliza el `AthenaAdmin` grupo de trabajo para crear la base de datos `newdb` en el catálogo de datos predeterminado. `AwsDataCatalog`

```
aws athena start-query-execution \  
  --query-string "create database if not exists newdb" \  
  --work-group "AthenaAdmin"
```

Salida:

```
{  
  "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11112"  
}
```

Para obtener más información, consulte [Ejecución de SQL consultas con Amazon Athena](#) en la Guía del usuario de Amazon Athena.

Ejemplo 3: Para ejecutar una consulta que cree una vista en una tabla de la base de datos y el catálogo de datos especificados

En el siguiente `start-query-execution` ejemplo, se utiliza una `SELECT` instrucción de la `cloudfront_logs` tabla `cflogsdatabase` para crear la `vistacf10`.

```
aws athena start-query-execution \  
  --query-string "CREATE OR REPLACE VIEW cf10 AS SELECT * FROM cloudfront_logs  
  limit 10" \  
  --query-execution-context Database=cflogsdatabase
```

Salida:

```
{  
  "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11113"  
}
```

Para obtener más información, consulte [Ejecución de SQL consultas con Amazon Athena](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [StartQueryExecution](#) de AWS CLI comandos.

## stop-query-execution

En el siguiente ejemplo de código, se muestra cómo usar `stop-query-execution`.

### AWS CLI

Para detener una consulta en ejecución

El siguiente `stop-query-execution` ejemplo detiene la consulta que tiene el identificador de consulta especificado.

```
aws athena stop-query-execution \  
  --query-execution-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Ejecución de SQL consultas con Amazon Athena](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [StopQueryExecution](#) de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Cómo añadir una etiqueta a un recurso

El siguiente `tag-resource` ejemplo agrega tres etiquetas al catálogo `dynamo_db_catalog` de datos.

```
aws athena tag-resource \  
  --resource-arn arn:aws:athena:us-west-2:111122223333:datacatalog/  
dynamo_db_catalog \  
  --  
  tags Key=Organization,Value=Retail Key=Division,Value=Mountain Key=Product_Line,Value=Shoes
```

Este comando no genera ninguna salida. Para ver el resultado, utilice `aws athena list-tags-for-resource --resource-arn arn:aws:athena:us-west-2:111122223333:datacatalog/dynamo_db_catalog`.

Para obtener más información, consulte [Añadir etiquetas a un recurso: tag-resource](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia de [TagResource](#) comandos AWS CLI .

## **untag-resource**

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar una etiqueta de un recurso

En el siguiente `untag-resource` ejemplo, se eliminan las Focus claves `Specialization` y sus valores asociados del recurso del catálogo de `dynamo_db_catalog` datos.

```
aws athena untag-resource \  
  --resource-arn arn:aws:athena:us-west-2:111122223333:datacatalog/  
dynamo_db_catalog \  
  --tag-keys Specialization Focus
```

Este comando no genera ninguna salida. Para ver los resultados, utilice el `list-tags-for-resource` comando.

Para obtener más información, consulte [Eliminar etiquetas de un recurso: untag-resource](#) en la Guía del usuario de Amazon Athena.

- Para obtener API más información, consulte la Referencia de comandos [UntagResource](#).AWS CLI

## **update-data-catalog**

En el siguiente ejemplo de código, se muestra cómo usar `update-data-catalog`.

### AWS CLI

Para actualizar un catálogo de datos

El siguiente `update-data-catalog` ejemplo actualiza la función Lambda y la descripción del catálogo de `cw_logs_catalog` datos.



```
aws athena update-data-catalog \  
  --name cw_logs_catalog \  
  --type LAMBDA \  
  --description "New CloudWatch Logs Catalog" \  
  --function=arn:aws:lambda:us-west-2:111122223333:function:new_cw_logs_lambda
```

Este comando no genera ninguna salida. Para ver el resultado, utilice `aws athena get-data-catalog --name cw_logs_catalog`.

Para obtener más información, consulte [Actualización de un catálogo: update-data-catalog](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte [UpdateDataCatalog](#) la Referencia de AWS CLI comandos.

## update-work-group

En el siguiente ejemplo de código, se muestra cómo usar `update-work-group`.

### AWS CLI

Para actualizar un grupo de trabajo

En el siguiente `update-work-group` ejemplo, se deshabilita el grupo de trabajo `Data_Analyst_Group`. Los usuarios no pueden ejecutar ni crear consultas en el grupo de trabajo deshabilitado, pero sí pueden ver las métricas, los controles de los límites de uso de datos, la configuración del grupo de trabajo, el historial de consultas y las consultas guardadas.

```
aws athena update-work-group \  
  --work-group Data_Analyst_Group \  
  --state DISABLED
```

Este comando no genera ninguna salida. Para comprobar el cambio de estado, utilice `aws athena get-work-group --work-group Data_Analyst_Group` y compruebe la `State` propiedad de la salida.

Para obtener más información, consulte [Administración de grupos de trabajo](#) en la Guía del usuario de Amazon Athena.

- Para API obtener más información, consulte la Referencia [UpdateWorkGroup](#) de AWS CLI comandos.

## Ejemplos de Auto Scaling usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el AWS Command Line Interface uso de Auto Scaling.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

#### **attach-instances**

En el siguiente ejemplo de código, se muestra cómo usar `attach-instances`.

#### AWS CLI

Para adjuntar una instancia a un grupo de Auto Scaling

En este ejemplo, se adjunta la instancia especificada al grupo de Auto Scaling especificado.

```
aws autoscaling attach-instances \  
  --instance-ids i-061c63c5eb45f0416 \  
  --auto-scaling-group-name my-asg
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [AttachInstances](#) la Referencia de AWS CLI comandos.

#### **attach-load-balancer-target-groups**

En el siguiente ejemplo de código, se muestra cómo usar `attach-load-balancer-target-groups`.

## AWS CLI

Asociación de un grupo de destino a un grupo de escalado automático

En este ejemplo, se asocia el grupo de destino especificado al grupo de escalado automático especificado.

```
aws autoscaling attach-load-balancer-target-groups \  
  --auto-scaling-group-name my-asg \  
  --target-group-arns arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Elastic Load Balancing y Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [AttachLoadBalancerTargetGroups](#) la Referencia de AWS CLI comandos.

## **attach-load-balancers**

En el siguiente ejemplo de código, se muestra cómo usar `attach-load-balancers`.

## AWS CLI

Para adjuntar un Classic Load Balancer a un grupo de Auto Scaling

En este ejemplo, se adjunta el Classic Load Balancer especificado al grupo de Auto Scaling especificado.

```
aws autoscaling attach-load-balancers \  
  --load-balancer-names my-load-balancer \  
  --auto-scaling-group-name my-asg
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Elastic Load Balancing y Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [AttachLoadBalancers](#) la Referencia de AWS CLI comandos.

## cancel-instance-refresh

En el siguiente ejemplo de código, se muestra cómo usar `cancel-instance-refresh`.

### AWS CLI

Para cancelar la actualización de una instancia

El siguiente `cancel-instance-refresh` ejemplo cancela una actualización de instancias en curso para el grupo de Auto Scaling especificado.

```
aws autoscaling cancel-instance-refresh \  
  --auto-scaling-group-name my-asg
```

Salida:

```
{  
  "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b"  
}
```

Para obtener más información, consulte [Cancelar una actualización de una instancia](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [CancelInstanceRefresh](#) la Referencia de AWS CLI comandos.

## complete-lifecycle-action

En el siguiente ejemplo de código, se muestra cómo usar `complete-lifecycle-action`.

### AWS CLI

Para completar la acción del ciclo de vida

En este ejemplo, se notifica a Amazon EC2 Auto Scaling que la acción del ciclo de vida especificada se ha completado para que pueda finalizar el lanzamiento o la finalización de la instancia.

```
aws autoscaling complete-lifecycle-action \  
  --lifecycle-hook-name my-launch-hook \  
  --auto-scaling-group-name my-asg \  
  --lifecycle-action-result CONTINUE \  
  --lifecycle-action-configuration-name my-launch-hook
```

```
--lifecycle-action-token bcd2f1b8-9a78-44d3-8a7a-4dd07d7cf635
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [enlaces del ciclo de vida de Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [CompleteLifecycleAction](#) la Referencia de AWS CLI comandos.

## create-auto-scaling-group

En el siguiente ejemplo de código, se muestra cómo usar `create-auto-scaling-group`.

### AWS CLI

Ejemplo 1: Creación de un grupo de escalado automático

En el siguiente ejemplo `create-auto-scaling-group` se crea un grupo de escalado automático en subredes de varias zonas de disponibilidad dentro de una región. Las instancias se lanzan con la versión predeterminada de la plantilla de lanzamiento especificada. Tenga en cuenta que se utilizan valores predeterminados para la mayoría de las demás configuraciones, como las políticas de terminación y la configuración de las comprobaciones de estado.

```
aws autoscaling create-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --launch-template LaunchTemplateId=lt-1234567890abcde12 \  
  --min-size 1 \  
  --max-size 5 \  
  --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [grupos de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Asociación de un equilibrador de carga de aplicación, un equilibrador de carga de red o un equilibrador de carga de puerta de enlace

En este ejemplo se especifica el grupo objetivo ARN de un balanceador de cargas que soporte el tráfico esperado. El tipo de comprobación de estado especifica ELB para que, cuando Elastic Load Balancing informa de una instancia como en mal estado, el grupo de escalado automático

reemplaza la instancia. El comando también define un período de gracia de 600 segundos para la comprobación de estado. El período de gracia ayuda a evitar la finalización prematura de las instancias recién lanzadas.

```
aws autoscaling create-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --launch-template LaunchTemplateId=lt-1234567890abcde12 \  
  --target-group-arns arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/943f017f100becff \  
  --health-check-type ELB \  
  --health-check-grace-period 600 \  
  --min-size 1 \  
  --max-size 5 \  
  --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Elastic Load Balancing y Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 3: Especificación de un grupo con ubicación y utilizar la versión más reciente de la plantilla de lanzamiento

En este ejemplo, se lanzan instancias a un grupo con ubicación en una única zona de disponibilidad. Esto puede resultar útil para grupos de baja latencia con cargas de HPC trabajo. En este ejemplo, también se especifican el tamaño mínimo, el tamaño máximo y la capacidad deseada del grupo.

```
aws autoscaling create-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --launch-template LaunchTemplateId=lt-1234567890abcde12,Version='$Latest' \  
  --min-size 1 \  
  --max-size 5 \  
  --desired-capacity 3 \  
  --placement-group my-placement-group \  
  --vpc-zone-identifier "subnet-6194ea3b"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Grupos de ubicación](#) en la Guía del EC2 usuario de Amazon para instancias de Linux.

Ejemplo 4: Especificación de un grupo de escalado automático de una sola instancia y utilizar una versión específica de la plantilla de lanzamiento

En este ejemplo, se crea un grupo de escalado automático con una capacidad mínima y máxima establecida en 1 para garantizar que se ejecute una instancia. El comando también especifica la versión 1 de una plantilla de lanzamiento en la que ENI se especifica el ID de una plantilla existente. Cuando utilice una plantilla de lanzamiento que especifique una existente ENI para eth0, debe especificar una zona de disponibilidad para el grupo de Auto Scaling que coincida con la interfaz de red, sin especificar también un ID de subred en la solicitud.

```
aws autoscaling create-auto-scaling-group \  
  --auto-scaling-group-name my-asg-single-instance \  
  --launch-template LaunchTemplateName=my-template-for-auto-scaling,Version='1' \  
  --min-size 1 \  
  --max-size 1 \  
  --availability-zones us-west-2a
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [grupos de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 5: Especificación de una política de terminación diferente

En este ejemplo, se crea un grupo de escalado automático mediante una configuración de lanzamiento y se establece la política de terminación para terminar primero las instancias más antiguas. El comando también aplica una etiqueta al grupo y a sus instancias, con una clave de Role y un valor de WebServer.

```
aws autoscaling create-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --launch-configuration-name my-lc \  
  --min-size 1 \  
  --max-size 5 \  
  --termination-policies "OldestInstance" \  
  --tags "ResourceId=my-asg,ResourceType=auto-scaling-  
group,Key=Role,Value=WebServer,PropagateAtLaunch=true" \  
  --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Cómo [trabajar con las políticas de terminación de Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

#### Ejemplo 6: Especificación de un enlace de ciclo de vida de lanzamiento

En este ejemplo, se crea un grupo de escalado automático con un enlace de ciclo de vida que admite una acción personalizada cuando se lanza una instancia.

```
aws autoscaling create-auto-scaling-group \  
--cli-input-json file://~/config.json
```

Contenido del archivo config.json:

```
{  
  "AutoScalingGroupName": "my-asg",  
  "LaunchTemplate": {  
    "LaunchTemplateId": "lt-1234567890abcde12"  
  },  
  "LifecycleHookSpecificationList": [{  
    "LifecycleHookName": "my-launch-hook",  
    "LifecycleTransition": "autoscaling:EC2_INSTANCE_LAUNCHING",  
    "NotificationTargetARN": "arn:aws:sqs:us-west-2:123456789012:my-sqs-queue",  
    "RoleARN": "arn:aws:iam::123456789012:role/my-notification-role",  
    "NotificationMetadata": "SQS message metadata",  
    "HeartbeatTimeout": 4800,  
    "DefaultResult": "ABANDON"  
  }],  
  "MinSize": 1,  
  "MaxSize": 5,  
  "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",  
  "Tags": [{  
    "ResourceType": "auto-scaling-group",  
    "ResourceId": "my-asg",  
    "PropagateAtLaunch": true,  
    "Value": "test",  
    "Key": "environment"  
  }]  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [enlaces del ciclo de vida de Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.



## Ejemplo 7: Especificación de un enlace de ciclo de vida de terminación

En este ejemplo, se crea un grupo de escalado automático con un enlace de ciclo de vida que admite una acción personalizada en la terminación de una instancia.

```
aws autoscaling create-auto-scaling-group \  
--cli-input-json file://~/config.json
```

Contenidos de config.json:

```
{  
  "AutoScalingGroupName": "my-asg",  
  "LaunchTemplate": {  
    "LaunchTemplateId": "lt-1234567890abcde12"  
  },  
  "LifecycleHookSpecificationList": [{  
    "LifecycleHookName": "my-termination-hook",  
    "LifecycleTransition": "autoscaling:EC2_INSTANCE_TERMINATING",  
    "HeartbeatTimeout": 120,  
    "DefaultResult": "CONTINUE"  
  }],  
  "MinSize": 1,  
  "MaxSize": 5,  
  "TargetGroupARNs": [  
    "arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-  
targets/73e2d6bc24d8a067"  
  ],  
  "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [enlaces del ciclo de vida de Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

## Ejemplo 8: Especificación de una política de terminación personalizada

En este ejemplo, se crea un grupo de Auto Scaling que especifica una política de terminación de funciones de Lambda personalizada que indica a Amazon EC2 Auto Scaling qué instancias son seguras de terminar al escalarlas.

```
aws autoscaling create-auto-scaling-group \  
--cli-input-json file://~/config.json
```

```
--auto-scaling-group-name my-asg-single-instance \  
--launch-template LaunchTemplateName=my-template-for-auto-scaling \  
--min-size 1 \  
--max-size 5 \  
--termination-policies "arn:aws:lambda:us-west-2:123456789012:function:HelloFunction:prod" \  
--vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de una política de rescisión personalizada con Lambda](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [CreateAutoScalingGroup](#) la Referencia de AWS CLI comandos.

## create-launch-configuration

En el siguiente ejemplo de código, se muestra cómo usar `create-launch-configuration`.

### AWS CLI

Ejemplo 1: Para crear una configuración de lanzamiento

En este ejemplo se crea una configuración de lanzamiento sencilla.

```
aws autoscaling create-launch-configuration \  
--launch-configuration-name my-lc \  
--image-id ami-04d5cc9b88example \  
--instance-type m5.Large
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de una configuración de lanzamiento](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Para crear una configuración de inicio con un grupo de seguridad, un key pair y un script de bootstrapping

En este ejemplo, se crea una configuración de inicio con un grupo de seguridad, un key pair y un script de bootstrapping incluidos en los datos del usuario.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --security-groups sg-eb2af88example \  
  --key-name my-key-pair \  
  --user-data file://myuserdata.txt
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de una configuración de lanzamiento](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 3: Para crear una configuración de lanzamiento con un IAM rol

En este ejemplo, se crea una configuración de lanzamiento con el nombre del perfil de instancia de un IAM rol.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --iam-instance-profile my-autoscaling-role
```

Este comando no genera ninguna salida.

Para obtener más información, consulte el [IAMrol de las aplicaciones que se ejecutan en EC2 instancias de Amazon](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 4: Para crear una configuración de lanzamiento con la supervisión detallada habilitada

En este ejemplo, se crea una configuración de lanzamiento con la supervisión EC2 detallada habilitada, que envía EC2 las métricas CloudWatch en períodos de 1 minuto.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --instance-monitoring Enabled=true
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de la supervisión de instancias de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 5: Para crear una configuración de lanzamiento que lance instancias puntuales

En este ejemplo, se crea una configuración de lanzamiento que utiliza instancias puntuales como única opción de compra.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --spot-price "0.50"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Solicitud de instancias puntuales](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 6: Para crear una configuración de lanzamiento mediante una EC2 instancia

En este ejemplo, se crea una configuración de lanzamiento basada en los atributos de una instancia existente. Anula la ubicación, el arrendamiento y la configuración de una dirección IP pública al incluir las opciones `--placement-tenancy` y `--no-associate-public-ip-address`.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc-from-instance \  
  --instance-id i-0123a456700123456 \  
  --instance-type m5.large \  
  --no-associate-public-ip-address \  
  --placement-tenancy dedicated
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de una configuración de lanzamiento mediante una EC2 instancia](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 7: Para crear una configuración de lanzamiento con un mapeo de dispositivos de bloques para un EBS volumen de Amazon

En este ejemplo, se crea una configuración de lanzamiento con un mapeo de dispositivos de bloques para un EBS gp3 volumen de Amazon con el nombre del dispositivo `/dev/sdh` y un tamaño de volumen de 20.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --block-device-mappings '[{"DeviceName":"/dev/sdh", "Ebs":  
{"VolumeSize":20, "VolumeType":"gp3"}]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [EBS](#) la API referencia de Amazon EC2 Auto Scaling.

Para obtener información sobre la sintaxis para citar valores JSON de parámetros con formato, consulte [Uso de comillas con cadenas AWS CLI en la Guía del usuario de la](#) interfaz de línea de AWS comandos.

Ejemplo 8: Para crear una configuración de lanzamiento con un mapeo de dispositivos de bloques para un volumen de almacén de instancias

En este ejemplo, se crea una configuración de lanzamiento con `ephemeral1` un volumen de almacén de instancias con el nombre del dispositivo `/dev/sdc`.

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.large \  
  --block-device-mappings '[{"DeviceName":"/dev/sdc", "VirtualName":"ephemeral1"}]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [BlockDeviceMapping](#) la API referencia de Amazon EC2 Auto Scaling.

Para obtener información sobre la sintaxis para citar valores JSON de parámetros con formato, consulte [Uso de comillas con cadenas AWS CLI en la Guía del usuario de la](#) interfaz de línea de AWS comandos.

Ejemplo 9: Para crear una configuración de inicio y evitar que un dispositivo de bloques se conecte en el momento del lanzamiento

En este ejemplo, se crea una configuración de lanzamiento que suprime un dispositivo de bloques especificado en el mapeo de dispositivos de bloques del AMI (por ejemplo, /dev/sdf).

```
aws autoscaling create-launch-configuration \  
  --launch-configuration-name my-lc \  
  --image-id ami-04d5cc9b88example \  
  --instance-type m5.Large \  
  --block-device-mappings ' [{"DeviceName": "/dev/sdf", "NoDevice": ""}] '
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [BlockDeviceMapping](#) la API referencia de Amazon EC2 Auto Scaling.

Para obtener información sobre la sintaxis para citar valores JSON de parámetros con formato, consulte [Uso de comillas con cadenas AWS CLI en la Guía del usuario de la](#) interfaz de línea de AWS comandos.

- Para API obtener más información, consulte la Referencia de [CreateLaunchConfiguration](#) comandos AWS CLI .

## create-or-update-tags

En el siguiente ejemplo de código, se muestra cómo usar `create-or-update-tags`.

### AWS CLI

Para crear o actualizar etiquetas para un grupo de Auto Scaling

Este ejemplo agrega dos etiquetas al grupo Auto Scaling especificado.

```
aws autoscaling create-or-update-tags \  
  --tags ResourceId=my-asg,ResourceType=auto-scaling-  
group,Key=Role,Value=WebServer,PropagateAtLaunch=true ResourceId=my-  
asg,ResourceType=auto-scaling-group,Key=Dept,Value=Research,PropagateAtLaunch=true
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de grupos e instancias de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [CreateOrUpdateTags](#) la Referencia de AWS CLI comandos.

## delete-auto-scaling-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-auto-scaling-group`.

### AWS CLI

Ejemplo 1: Eliminación del grupo de escalado automático especificado

En este ejemplo, se elimina el grupo de escalado automático especificado.

```
aws autoscaling delete-auto-scaling-group \  
  --auto-scaling-group-name my-asg
```

Este comando no genera ninguna salida.

Para obtener más información, [consulte Eliminar la infraestructura de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Forzado de la eliminación del grupo de escalado automático especificado

Para eliminar el grupo de escalado automático sin esperar a que las instancias del grupo terminen, utilice la opción `--force-delete`.

```
aws autoscaling delete-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --force-delete
```

Este comando no genera ninguna salida.

Para obtener más información, [consulte Eliminar la infraestructura de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DeleteAutoScalingGroup](#) la Referencia de AWS CLI comandos.

## delete-launch-configuration

En el siguiente ejemplo de código, se muestra cómo usar `delete-launch-configuration`.

### AWS CLI

Para eliminar una configuración de lanzamiento

En este ejemplo, se elimina la configuración de lanzamiento especificada.

```
aws autoscaling delete-launch-configuration \  
  --launch-configuration-name my-launch-config
```

Este comando no genera ninguna salida.

Para obtener más información, [consulte Eliminar la infraestructura de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DeleteLaunchConfiguration](#) la Referencia de AWS CLI comandos.

## **delete-lifecycle-hook**

En el siguiente ejemplo de código, se muestra cómo usar `delete-lifecycle-hook`.

### AWS CLI

Para eliminar un enlace de ciclo de vida

En este ejemplo, se elimina el enlace del ciclo de vida especificado.

```
aws autoscaling delete-lifecycle-hook \  
  --lifecycle-hook-name my-lifecycle-hook \  
  --auto-scaling-group-name my-asg
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteLifecycleHook](#) la Referencia de AWS CLI comandos.

## **delete-notification-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `delete-notification-configuration`.

### AWS CLI

Para eliminar una notificación de Auto Scaling

En este ejemplo, se elimina la notificación especificada del grupo de Auto Scaling especificado.



```
aws autoscaling delete-notification-configuration \  
  --auto-scaling-group-name my-asg \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:my-sns-topic
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar la configuración de notificaciones](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DeleteNotificationConfiguration](#) la Referencia de AWS CLI comandos.

## delete-policy

En el siguiente ejemplo de código, se muestra cómo usar `delete-policy`.

### AWS CLI

Para eliminar una política de escalado

En este ejemplo, se elimina la política de escalado especificada.

```
aws autoscaling delete-policy \  
  --auto-scaling-group-name my-asg \  
  --policy-name alb1000-target-tracking-scaling-policy
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeletePolicy](#) la Referencia de AWS CLI comandos.

## delete-scheduled-action

En el siguiente ejemplo de código, se muestra cómo usar `delete-scheduled-action`.

### AWS CLI

Para eliminar una acción programada de un grupo de Auto Scaling

En este ejemplo, se elimina la acción programada especificada del grupo de Auto Scaling especificado.

```
aws autoscaling delete-scheduled-action \  
  --auto-scaling-group-name my-asg \  
  --scheduled-action-name my-scheduled-action
```

```
--auto-scaling-group-name my-asg \  
--scheduled-action-name my-scheduled-action
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteScheduledAction](#) la Referencia de AWS CLI comandos.

## delete-tags

En el siguiente ejemplo de código, se muestra cómo usar delete-tags.

AWS CLI

Para eliminar una etiqueta de un grupo de Auto Scaling

En este ejemplo, se elimina la etiqueta especificada del grupo de Auto Scaling especificado.

```
aws autoscaling delete-tags \  
--tags ResourceId=my-asg,ResourceType=auto-scaling-group,Key=Dept,Value=Research
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de grupos e instancias de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DeleteTags](#) la Referencia de AWS CLI comandos.

## delete-warm-pool

En el siguiente ejemplo de código, se muestra cómo usar delete-warm-pool.

AWS CLI

Ejemplo 1: Para eliminar una piscina caliente

El siguiente ejemplo elimina la piscina caliente del grupo de Auto Scaling especificado.

```
aws autoscaling delete-warm-pool \  
--auto-scaling-group-name my-asg
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Warm Pools for Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Para forzar la eliminación de una piscina caliente

Para eliminar la piscina caliente sin esperar a que sus instancias terminen, usa la `--force-delete` opción.

```
aws autoscaling delete-warm-pool \  
  --auto-scaling-group-name my-asg \  
  --force-delete
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Warm Pools for Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DeleteWarmPool](#) la Referencia de AWS CLI comandos.

## **describe-account-limits**

En el siguiente ejemplo de código, se muestra cómo usar `describe-account-limits`.

AWS CLI

Para describir los límites de su cuenta EC2 de Amazon Auto Scaling

En este ejemplo, se describen los límites EC2 de Amazon Auto Scaling para su AWS cuenta.

```
aws autoscaling describe-account-limits
```

Salida:

```
{  
  "NumberOfLaunchConfigurations": 5,  
  "MaxNumberOfLaunchConfigurations": 100,  
  "NumberOfAutoScalingGroups": 3,  
  "MaxNumberOfAutoScalingGroups": 20
```

```
}
```

Para obtener más información, consulte las [cuotas de servicio de Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DescribeAccountLimits](#) la Referencia de AWS CLI comandos.

## **describe-adjustment-types**

En el siguiente ejemplo de código, se muestra cómo usar `describe-adjustment-types`.

AWS CLI

Para describir los tipos de ajustes de escala disponibles

En este ejemplo se describen los tipos de ajuste disponibles.

```
aws autoscaling describe-adjustment-types
```

Salida:

```
{
  "AdjustmentTypes": [
    {
      "AdjustmentType": "ChangeInCapacity"
    },
    {
      "AdjustmentType": "ExactCapacity"
    },
    {
      "AdjustmentType": "PercentChangeInCapacity"
    }
  ]
}
```

Para obtener más información, consulte [Tipos de ajustes de escalado](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DescribeAdjustmentTypes](#) la Referencia de AWS CLI comandos.

## describe-auto-scaling-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-auto-scaling-groups`.

### AWS CLI

Ejemplo 1: Descripción del grupo de escalado automático especificado

En este ejemplo, se describe el grupo de escalado automático especificado.

```
aws autoscaling describe-auto-scaling-groups \  
  --auto-scaling-group-name my-asg
```

Salida:

```
{  
  "AutoScalingGroups": [  
    {  
      "AutoScalingGroupName": "my-asg",  
      "AutoScalingGroupARN": "arn:aws:autoscaling:us-  
west-2:123456789012:autoScalingGroup:930d940e-891e-4781-  
a11a-7b0acd480f03:autoScalingGroupName/my-asg",  
      "LaunchTemplate": {  
        "LaunchTemplateName": "my-launch-template",  
        "Version": "1",  
        "LaunchTemplateId": "lt-1234567890abcde12"  
      },  
      "MinSize": 0,  
      "MaxSize": 1,  
      "DesiredCapacity": 1,  
      "DefaultCooldown": 300,  
      "AvailabilityZones": [  
        "us-west-2a",  
        "us-west-2b",  
        "us-west-2c"  
      ],  
      "LoadBalancerNames": [],  
      "TargetGroupARNs": [],  
      "HealthCheckType": "EC2",  
      "HealthCheckGracePeriod": 0,  
      "Instances": [  
        {  
          "InstanceId": "i-06905f55584de02da",
```

```

        "InstanceType": "t2.micro",
        "AvailabilityZone": "us-west-2a",
        "HealthStatus": "Healthy",
        "LifecycleState": "InService",
        "ProtectedFromScaleIn": false,
        "LaunchTemplate": {
            "LaunchTemplateName": "my-launch-template",
            "Version": "1",
            "LaunchTemplateId": "lt-1234567890abcde12"
        }
    ],
    "CreatedTime": "2023-10-28T02:39:22.152Z",
    "SuspendedProcesses": [],
    "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
    "EnabledMetrics": [],
    "Tags": [],
    "TerminationPolicies": [
        "Default"
    ],
    "NewInstancesProtectedFromScaleIn": false,
    "ServiceLinkedRoleARN": "arn",
    "TrafficSources": []
}
]
}

```

Ejemplo 2: Descripción de los 100 primeros grupos de escalado automático especificados

En este ejemplo, se describen los grupos de escalado automático especificados. Le permite especificar hasta 100 nombres de grupos.

```

aws autoscaling describe-auto-scaling-groups \
  --max-items 100 \
  --auto-scaling-group-name "group1" "group2" "group3" "group4"

```

Consulte el ejemplo 1 para ver una salida de muestra.

Ejemplo 3: Descripción de un grupo de escalado automático en la región especificada

En este ejemplo, se describen los grupos de escalado automático en la región especificada, hasta un máximo de 75 grupos.

```
aws autoscaling describe-auto-scaling-groups \  
  --max-items 75 \  
  --region us-east-1
```

Consulte el ejemplo 1 para ver una salida de muestra.

Ejemplo 4: Descripción del número especificado de grupos de escalado automático

Para devolver un número específico de grupos de escalado automático, utilice la opción `--max-items`.

```
aws autoscaling describe-auto-scaling-groups \  
  --max-items 1
```

Consulte el ejemplo 1 para ver una salida de muestra.

Si la salida incluye un campo `NextToken`, hay más grupos. Para obtener los grupos adicionales, utilice el valor de este campo con la opción `--starting-token` en una llamada posterior de la siguiente manera.

```
aws autoscaling describe-auto-scaling-groups \  
  --starting-token Z3M3LMPEXAMPLE
```

Consulte el ejemplo 1 para ver una salida de muestra.

Ejemplo 5: Para describir los grupos de Auto Scaling que utilizan configuraciones de lanzamiento

En este ejemplo, se usa la `--query` opción para describir los grupos de Auto Scaling que usan configuraciones de lanzamiento.

```
aws autoscaling describe-auto-scaling-groups \  
  --query 'AutoScalingGroups[?LaunchConfigurationName!=`null`]'
```

Salida:

```
[  
  {  
    "AutoScalingGroupName": "my-asg",  
    "AutoScalingGroupARN": "arn:aws:autoscaling:us-  
west-2:123456789012:autoScalingGroup:930d940e-891e-4781-  
a11a-7b0acd480f03:autoScalingGroupName/my-asg",
```

```
"LaunchConfigurationName": "my-lc",
"MinSize": 0,
"MaxSize": 1,
"DesiredCapacity": 1,
"DefaultCooldown": 300,
"AvailabilityZones": [
  "us-west-2a",
  "us-west-2b",
  "us-west-2c"
],
"LoadBalancerNames": [],
"TargetGroupARNs": [],
"HealthCheckType": "EC2",
"HealthCheckGracePeriod": 0,
"Instances": [
  {
    "InstanceId": "i-088c57934a6449037",
    "InstanceType": "t2.micro",
    "AvailabilityZone": "us-west-2c",
    "HealthStatus": "Healthy",
    "LifecycleState": "InService",
    "LaunchConfigurationName": "my-lc",
    "ProtectedFromScaleIn": false
  }
],
"CreatedTime": "2023-10-28T02:39:22.152Z",
"SuspendedProcesses": [],
"VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
"EnabledMetrics": [],
"Tags": [],
"TerminationPolicies": [
  "Default"
],
"NewInstancesProtectedFromScaleIn": false,
"ServiceLinkedRoleARN": "arn",
"TrafficSources": []
}
]
```

Para obtener más información, consulte [Filtrar los AWS CLI resultados](#) en la Guía del usuario de la interfaz de línea de AWS comandos.

- Para API obtener más información, consulte [DescribeAutoScalingGroups](#) la Referencia de AWS CLI comandos.



## describe-auto-scaling-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-auto-scaling-instances`.

### AWS CLI

Ejemplo 1: Descripción de una o varias instancias

En este ejemplo, se describe la instancia especificada.

```
aws autoscaling describe-auto-scaling-instances \  
  --instance-ids i-06905f55584de02da
```

Salida:

```
{  
  "AutoScalingInstances": [  
    {  
      "InstanceId": "i-06905f55584de02da",  
      "InstanceType": "t2.micro",  
      "AutoScalingGroupName": "my-asg",  
      "AvailabilityZone": "us-west-2b",  
      "LifecycleState": "InService",  
      "HealthStatus": "HEALTHY",  
      "ProtectedFromScaleIn": false,  
      "LaunchTemplate": {  
        "LaunchTemplateId": "lt-1234567890abcde12",  
        "LaunchTemplateName": "my-launch-template",  
        "Version": "1"  
      }  
    }  
  ]  
}
```

Ejemplo 2: Descripción de una o varias instancias

En este ejemplo, se utiliza la opción `--max-items` para especificar cuántas instancias se devolverán con esta llamada.

```
aws autoscaling describe-auto-scaling-instances \  
  --max-items 1
```

Si la salida incluye un campo `NextToken`, hay más instancias. Para obtener las instancias adicionales, utilice el valor de este campo con la opción `--starting-token` en una llamada posterior de la siguiente manera.

```
aws autoscaling describe-auto-scaling-instances \  
  --starting-token Z3M3LMPEXAMPLE
```

Consulte el ejemplo 1 para ver una salida de muestra.

Ejemplo 3: Para describir las instancias que utilizan configuraciones de lanzamiento

En este ejemplo, se usa la `--query` opción para describir las instancias que usan configuraciones de lanzamiento.

```
aws autoscaling describe-auto-scaling-instances \  
  --query 'AutoScalingInstances[?LaunchConfigurationName!=`null`]'
```

Salida:

```
[  
  {  
    "InstanceId": "i-088c57934a6449037",  
    "InstanceType": "t2.micro",  
    "AutoScalingGroupName": "my-asg",  
    "AvailabilityZone": "us-west-2c",  
    "LifecycleState": "InService",  
    "HealthStatus": "HEALTHY",  
    "LaunchConfigurationName": "my-lc",  
    "ProtectedFromScaleIn": false  
  }  
]
```

Para obtener más información, consulte [Filtrar los AWS CLI resultados](#) en la Guía del usuario de la interfaz de línea de AWS comandos.

- Para API obtener más información, consulte [DescribeAutoScalingInstances](#) la Referencia de AWS CLI comandos.

## describe-auto-scaling-notification-types

En el siguiente ejemplo de código, se muestra cómo usar `describe-auto-scaling-notification-types`.

### AWS CLI

Para describir los tipos de notificación disponibles

En este ejemplo se describen los tipos de notificación disponibles.

```
aws autoscaling describe-auto-scaling-notification-types
```

Salida:

```
{
  "AutoScalingNotificationTypes": [
    "autoscaling:EC2_INSTANCE_LAUNCH",
    "autoscaling:EC2_INSTANCE_LAUNCH_ERROR",
    "autoscaling:EC2_INSTANCE_TERMINATE",
    "autoscaling:EC2_INSTANCE_TERMINATE_ERROR",
    "autoscaling:TEST_NOTIFICATION"
  ]
}
```

Para obtener más información, consulte [Cómo recibir SNS notificaciones de Amazon cuando su grupo de Auto Scaling se amplía](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DescribeAutoScalingNotificationTypes](#) la Referencia de AWS CLI comandos.

## describe-instance-refreshes

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-refreshes`.

### AWS CLI

Para describir las actualizaciones de instancias

El siguiente `describe-instance-refreshes` ejemplo devuelve una descripción de todas las solicitudes de actualización de instancias para el grupo de Auto Scaling especificado, incluido el mensaje de estado y (si está disponible) el motivo del estado.

```
aws autoscaling describe-instance-refreshes \  
--auto-scaling-group-name my-asg
```

Salida:

```
{  
  "InstanceRefreshes": [  
    {  
      "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b",  
      "AutoScalingGroupName": "my-asg",  
      "Status": "InProgress",  
      "StatusReason": "Waiting for instances to warm up before continuing. For  
example: 0e69cc3f05f825f4f is warming up.",  
      "EndTime": "2023-03-23T16:42:55Z",  
      "PercentageComplete": 0,  
      "InstancesToUpdate": 0,  
      "Preferences": {  
        "MinHealthyPercentage": 100,  
        "InstanceWarmup": 300,  
        "CheckpointPercentages": [  
          50  
        ],  
        "CheckpointDelay": 3600,  
        "SkipMatching": false,  
        "AutoRollback": true,  
        "ScaleInProtectedInstances": "Ignore",  
        "StandbyInstances": "Ignore"  
      }  
    },  
    {  
      "InstanceRefreshId": "dd7728d0-5bc4-4575-96a3-1b2c52bf8bb1",  
      "AutoScalingGroupName": "my-asg",  
      "Status": "Successful",  
      "EndTime": "2022-06-02T16:53:37Z",  
      "PercentageComplete": 100,  
      "InstancesToUpdate": 0,  
      "Preferences": {  
        "MinHealthyPercentage": 90,  
        "InstanceWarmup": 300,  
        "SkipMatching": true,  
        "AutoRollback": true,  
        "ScaleInProtectedInstances": "Ignore",  
        "StandbyInstances": "Ignore"  
      }  
    }  
  ]  
}
```

```

    }
  }
]
}

```

Para obtener más información, [consulte Comprobar el estado de una actualización de instancias](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DescribeInstanceRefreshes](#) la Referencia de AWS CLI comandos.

## describe-launch-configurations

En el siguiente ejemplo de código, se muestra cómo usar `describe-launch-configurations`.

### AWS CLI

Ejemplo 1: Para describir la configuración de lanzamiento especificada

En este ejemplo se describe la configuración de lanzamiento especificada.

```
aws autoscaling describe-launch-configurations \
  --launch-configuration-names my-launch-config
```

Salida:

```
{
  "LaunchConfigurations": [
    {
      "LaunchConfigurationName": "my-launch-config",
      "LaunchConfigurationARN": "arn:aws:autoscaling:us-west-2:123456789012:launchConfiguration:98d3b196-4cf9-4e88-8ca1-8547c24ced8b:launchConfigura
my-launch-config",
      "ImageId": "ami-0528a5175983e7f28",
      "KeyName": "my-key-pair-uswest2",
      "SecurityGroups": [
        "sg-05eaec502fcdadc2e"
      ],
      "ClassicLinkVPCSecurityGroups": [],
      "UserData": "",
      "InstanceType": "t2.micro",
      "KernelId": "",

```

```

    "RamdiskId": "",
    "BlockDeviceMappings": [
      {
        "DeviceName": "/dev/xvda",
        "Ebs": {
          "SnapshotId": "snap-06c1606ba5ca274b1",
          "VolumeSize": 8,
          "VolumeType": "gp2",
          "DeleteOnTermination": true,
          "Encrypted": false
        }
      }
    ],
    "InstanceMonitoring": {
      "Enabled": true
    },
    "CreatedTime": "2020-10-28T02:39:22.321Z",
    "EbsOptimized": false,
    "AssociatePublicIpAddress": true,
    "MetadataOptions": {
      "HttpTokens": "required",
      "HttpPutResponseHopLimit": 1,
      "HttpEndpoint": "disabled"
    }
  }
]
}

```

Ejemplo 2: Para describir un número específico de configuraciones de lanzamiento

Para devolver un número específico de configuraciones de lanzamiento, utilice la `--max-items` opción.

```

aws autoscaling describe-launch-configurations \
  --max-items 1

```

Si la salida incluye un `NextToken` campo, hay más configuraciones de lanzamiento. Para obtener las configuraciones de lanzamiento adicionales, utilice el valor de este campo con la `--starting-token` opción en una llamada posterior, de la siguiente manera.

```

aws autoscaling describe-launch-configurations \
  --starting-token Z3M3LMPEXAMPLE

```

- Para API obtener más información, consulte [DescribeLaunchConfigurations](#) la Referencia de AWS CLI comandos.

## **describe-lifecycle-hook-types**

En el siguiente ejemplo de código, se muestra cómo usar `describe-lifecycle-hook-types`.

AWS CLI

Para describir los tipos de enlaces disponibles durante el ciclo de vida

En este ejemplo, se describen los tipos de enlaces de ciclo de vida disponibles.

```
aws autoscaling describe-lifecycle-hook-types
```

Salida:

```
{
  "LifecycleHookTypes": [
    "autoscaling:EC2_INSTANCE_LAUNCHING",
    "autoscaling:EC2_INSTANCE_TERMINATING"
  ]
}
```

- Para API obtener más información, consulte [DescribeLifecycleHookTypes](#) la Referencia de AWS CLI comandos.

## **describe-lifecycle-hooks**

En el siguiente ejemplo de código, se muestra cómo usar `describe-lifecycle-hooks`.

AWS CLI

Para describir sus ganchos de ciclo de vida

En este ejemplo, se describen los enlaces del ciclo de vida del grupo de Auto Scaling especificado.

```
aws autoscaling describe-lifecycle-hooks \
```

```
--auto-scaling-group-name my-asg
```

Salida:

```
{
  "LifecycleHooks": [
    {
      "GlobalTimeout": 3000,
      "HeartbeatTimeout": 30,
      "AutoScalingGroupName": "my-asg",
      "LifecycleHookName": "my-launch-hook",
      "DefaultResult": "ABANDON",
      "LifecycleTransition": "autoscaling:EC2_INSTANCE_LAUNCHING"
    },
    {
      "GlobalTimeout": 6000,
      "HeartbeatTimeout": 60,
      "AutoScalingGroupName": "my-asg",
      "LifecycleHookName": "my-termination-hook",
      "DefaultResult": "CONTINUE",
      "LifecycleTransition": "autoscaling:EC2_INSTANCE_TERMINATING"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeLifecycleHooks](#) la Referencia de AWS CLI comandos.

## describe-load-balancer-target-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-load-balancer-target-groups`.

AWS CLI

Para describir los grupos objetivo del balanceador de carga para un grupo de Auto Scaling

En este ejemplo, se describen los grupos objetivo del balanceador de carga asociados al grupo de Auto Scaling especificado.

```
aws autoscaling describe-load-balancer-target-groups \
```



```
--auto-scaling-group-name my-asg
```

Salida:

```
{
  "LoadBalancerTargetGroups": [
    {
      "LoadBalancerTargetGroupARN": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
      "State": "Added"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeLoadBalancerTargetGroups](#) la Referencia de AWS CLI comandos.

## describe-load-balancers

En el siguiente ejemplo de código, se muestra cómo usar describe-load-balancers.

AWS CLI

Para describir los balanceadores de carga clásicos para un grupo de Auto Scaling

Este ejemplo describe los balanceadores de carga clásicos para el grupo de Auto Scaling especificado.

```
aws autoscaling describe-load-balancers \
  --auto-scaling-group-name my-asg
```

Salida:

```
{
  "LoadBalancers": [
    {
      "State": "Added",
      "LoadBalancerName": "my-load-balancer"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeLoadBalancers](#) la Referencia de AWS CLI comandos.

## describe-metric-collection-types

En el siguiente ejemplo de código, se muestra cómo usar `describe-metric-collection-types`.

AWS CLI

Para describir los tipos de recopilación de métricas disponibles

En este ejemplo, se describen los tipos de recopilación de métricas disponibles.

```
aws autoscaling describe-metric-collection-types
```

Salida:

```
{
  "Metrics": [
    {
      "Metric": "GroupMinSize"
    },
    {
      "Metric": "GroupMaxSize"
    },
    {
      "Metric": "GroupDesiredCapacity"
    },
    {
      "Metric": "GroupInServiceInstances"
    },
    {
      "Metric": "GroupInServiceCapacity"
    },
    {
      "Metric": "GroupPendingInstances"
    },
    {
      "Metric": "GroupPendingCapacity"
    }
  ]
}
```

```

        "Metric": "GroupTerminatingInstances"
    },
    {
        "Metric": "GroupTerminatingCapacity"
    },
    {
        "Metric": "GroupStandbyInstances"
    },
    {
        "Metric": "GroupStandbyCapacity"
    },
    {
        "Metric": "GroupTotalInstances"
    },
    {
        "Metric": "GroupTotalCapacity"
    }
],
"Granularities": [
    {
        "Granularity": "1Minute"
    }
]
}

```

Para obtener más información, consulte [las métricas de grupo de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DescribeMetricCollectionTypes](#) la Referencia de AWS CLI comandos.

## **describe-notification-configurations**

En el siguiente ejemplo de código, se muestra cómo usar `describe-notification-configurations`.

### AWS CLI

Ejemplo 1: Para describir las configuraciones de notificación de un grupo específico

Este ejemplo describe las configuraciones de notificación para el grupo de Auto Scaling especificado.

```
aws autoscaling describe-notification-configurations \
  --auto-scaling-group-name my-asg
```

Salida:

```
{
  "NotificationConfigurations": [
    {
      "AutoScalingGroupName": "my-asg",
      "NotificationType": "autoscaling:TEST_NOTIFICATION",
      "TopicARN": "arn:aws:sns:us-west-2:123456789012:my-sns-topic-2"
    },
    {
      "AutoScalingGroupName": "my-asg",
      "NotificationType": "autoscaling:TEST_NOTIFICATION",
      "TopicARN": "arn:aws:sns:us-west-2:123456789012:my-sns-topic"
    }
  ]
}
```

Para obtener más información, consulte [Cómo recibir SNS notificaciones de Amazon cuando su grupo de Auto Scaling se amplía](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 1: Para describir un número específico de configuraciones de notificaciones

Para devolver un número específico de configuraciones de notificaciones, utilice el `max-items` parámetro.

```
aws autoscaling describe-notification-configurations \
  --auto-scaling-group-name my-auto-scaling-group \
  --max-items 1
```

Salida:

```
{
  "NotificationConfigurations": [
    {
      "AutoScalingGroupName": "my-asg",
      "NotificationType": "autoscaling:TEST_NOTIFICATION",
      "TopicARN": "arn:aws:sns:us-west-2:123456789012:my-sns-topic-2"
    },
    {
```

```
        "AutoScalingGroupName": "my-asg",
        "NotificationType": "autoscaling:TEST_NOTIFICATION",
        "TopicARN": "arn:aws:sns:us-west-2:123456789012:my-sns-topic"
    }
]
}
```

Si la salida incluye un `NextToken` campo, hay más configuraciones de notificación. Para obtener las configuraciones de notificación adicionales, utilice el valor de este campo con el `starting-token` parámetro en una llamada posterior de la siguiente manera.

```
aws autoscaling describe-notification-configurations \
  --auto-scaling-group-name my-asg \
  --starting-token Z3M3LMPEXAMPLE
```

Para obtener más información, consulte [Cómo recibir SNS notificaciones de Amazon cuando su grupo de Auto Scaling se amplía](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DescribeNotificationConfigurations](#) la Referencia de AWS CLI comandos.

## describe-policies

En el siguiente ejemplo de código, se muestra cómo usar `describe-policies`.

### AWS CLI

Ejemplo 1: Para describir las políticas de escalado de un grupo específico

En este ejemplo, se describen las políticas de escalado del grupo de Auto Scaling especificado.

```
aws autoscaling describe-policies \
  --auto-scaling-group-name my-asg
```

Salida:

```
{
  "ScalingPolicies": [
    {
      "AutoScalingGroupName": "my-asg",
      "PolicyName": "alb1000-target-tracking-scaling-policy",
```

```

    "PolicyARN": "arn:aws:autoscaling:us-
west-2:123456789012:scalingPolicy:3065d9c8-9969-4bec-
bb6a-3fbe5550fde6:autoScalingGroupName/my-asg:policyName/alb1000-target-tracking-
scaling-policy",
    "PolicyType": "TargetTrackingScaling",
    "StepAdjustments": [],
    "Alarms": [
      {
        "AlarmName": "TargetTracking-my-asg-
AlarmHigh-924887a9-12d7-4e01-8686-6f844d13a196",
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-
AlarmHigh-924887a9-12d7-4e01-8686-6f844d13a196"
      },
      {
        "AlarmName": "TargetTracking-my-asg-AlarmLow-f96f899d-b8e7-4d09-
a010-c1aaa35da296",
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-AlarmLow-f96f899d-b8e7-4d09-a010-
c1aaa35da296"
      }
    ],
    "TargetTrackingConfiguration": {
      "PredefinedMetricSpecification": {
        "PredefinedMetricType": "ALBRequestCountPerTarget",
        "ResourceLabel": "app/my-alb/778d41231b141a0f/targetgroup/my-
alb-target-group/943f017f100becff"
      },
      "TargetValue": 1000.0,
      "DisableScaleIn": false
    },
    "Enabled": true
  },
  {
    "AutoScalingGroupName": "my-asg",
    "PolicyName": "cpu40-target-tracking-scaling-policy",
    "PolicyARN": "arn:aws:autoscaling:us-
west-2:123456789012:scalingPolicy:5fd26f71-39d4-4690-82a9-
b8515c45cdde:autoScalingGroupName/my-asg:policyName/cpu40-target-tracking-scaling-
policy",
    "PolicyType": "TargetTrackingScaling",
    "StepAdjustments": [],
    "Alarms": [
      {

```

```

        "AlarmName": "TargetTracking-my-asg-
AlarmHigh-139f9789-37b9-42ad-bea5-b5b147d7f473",
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-AlarmHigh-139f9789-37b9-42ad-bea5-
b5b147d7f473"
    },
    {
        "AlarmName": "TargetTracking-my-asg-AlarmLow-bd681c67-
fc18-4c56-8468-fb8e413009c9",
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-AlarmLow-bd681c67-fc18-4c56-8468-
fb8e413009c9"
    }
],
"TargetTrackingConfiguration": {
    "PredefinedMetricSpecification": {
        "PredefinedMetricType": "ASGAverageCPUUtilization"
    },
    "TargetValue": 40.0,
    "DisableScaleIn": false
},
"Enabled": true
}
]
}

```

Para obtener más información, consulte [Escalado dinámico](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Para describir las políticas de escalado de un nombre específico

Para obtener políticas de escalado específicas, utilice la `--policy-names` opción.

```

aws autoscaling describe-policies \
  --auto-scaling-group-name my-asg \
  --policy-names cpu40-target-tracking-scaling-policy

```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte [Escalado dinámico](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 3: Para describir una serie de políticas de escalado

Para devolver un número específico de políticas, utilice la `--max-items` opción.

```
aws autoscaling describe-policies \  
  --auto-scaling-group-name my-asg \  
  --max-items 1
```

Consulte el ejemplo 1 para ver una salida de muestra.

Si el resultado incluye un `NextToken` campo, utilice el valor de este campo con la `--starting-token` opción en una llamada posterior para obtener las políticas adicionales.

```
aws autoscaling describe-policies --auto-scaling-group-name my-asg --starting-  
token Z3M3LMPEXAMPLE
```

Para obtener más información, consulte [Escalado dinámico](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DescribePolicies](#) la Referencia de AWS CLI comandos.

## describe-scaling-activities

En el siguiente ejemplo de código, se muestra cómo usar `describe-scaling-activities`.

AWS CLI

Ejemplo 1: Descripción de las actividades de escalado para el grupo especificado

En este ejemplo, se describen una o varias actividades de escalado para el grupo de escalado automático especificado.

```
aws autoscaling describe-scaling-activities \  
  --auto-scaling-group-name my-asg
```

Salida:

```
{  
  "Activities": [  
    {  
      "ActivityId": "f9f2d65b-f1f2-43e7-b46d-d86756459699",
```



```

        "Description": "Launching a new EC2 instance: i-0d44425630326060f",
        "AutoScalingGroupName": "my-asg",
        "Cause": "At 2020-10-30T19:35:51Z a user request update of
AutoScalingGroup constraints to min: 0, max: 16, desired: 16 changing the desired
capacity from 0 to 16. At 2020-10-30T19:36:07Z an instance was started in response
to a difference between desired and actual capacity, increasing the capacity from 0
to 16.",
        "StartTime": "2020-10-30T19:36:09.766Z",
        "EndTime": "2020-10-30T19:36:41Z",
        "StatusCode": "Successful",
        "Progress": 100,
        "Details": "{\"Subnet ID\": \"subnet-5ea0c127\", \"Availability Zone\":
\\\"us-west-2b\\\"}"
    }
]
}

```

Para obtener más información, consulte [Verificar una actividad de escalado para un grupo de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Descripción de las actividades de escalado de un grupo eliminado

Para describir las actividades de escalado después de que un grupo de escalado automático se haya eliminado, añada la opción `--include-deleted-groups`.

```

aws autoscaling describe-scaling-activities \
  --auto-scaling-group-name my-asg \
  --include-deleted-groups

```

Salida:

```

{
  "Activities": [
    {
      "ActivityId": "e1f5de0e-f93e-1417-34ac-092a76fba220",
      "Description": "Launching a new EC2 instance. Status Reason: Your Spot
request price of 0.001 is lower than the minimum required Spot request fulfillment
price of 0.0031. Launching EC2 instance failed.",
      "AutoScalingGroupName": "my-asg",
      "Cause": "At 2021-01-13T20:47:24Z a user request update of
AutoScalingGroup constraints to min: 1, max: 5, desired: 3 changing the desired
capacity from 0 to 3. At 2021-01-13T20:47:27Z an instance was started in response

```

```

to a difference between desired and actual capacity, increasing the capacity from 0
to 3.",
    "StartTime": "2021-01-13T20:47:30.094Z",
    "EndTime": "2021-01-13T20:47:30Z",
    "StatusCode": "Failed",
    "StatusMessage": "Your Spot request price of 0.001 is lower than the
minimum required Spot request fulfillment price of 0.0031. Launching EC2 instance
failed.",
    "Progress": 100,
    "Details": "{\"Subnet ID\": \"subnet-5ea0c127\", \"Availability Zone\":
\"us-west-2b\"}",
    "AutoScalingGroupState": "Deleted",
    "AutoScalingGroupARN": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:283179a2-
f3ce-423d-93f6-66bb518232f7:autoScalingGroupName/my-asg"
  }
]
}

```

Para obtener más información, consulte [Solución de problemas de Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 3: Descripción de un número específico de actividades de escalado

Para devolver un número específico de actividades, utilice la opción `--max-items`.

```

aws autoscaling describe-scaling-activities \
  --max-items 1

```

Salida:

```

{
  "Activities": [
    {
      "ActivityId": "f9f2d65b-f1f2-43e7-b46d-d86756459699",
      "Description": "Launching a new EC2 instance: i-0d44425630326060f",
      "AutoScalingGroupName": "my-asg",
      "Cause": "At 2020-10-30T19:35:51Z a user request update of
AutoScalingGroup constraints to min: 0, max: 16, desired: 16 changing the desired
capacity from 0 to 16. At 2020-10-30T19:36:07Z an instance was started in response
to a difference between desired and actual capacity, increasing the capacity from 0
to 16.",
      "StartTime": "2020-10-30T19:36:09.766Z",

```

```

        "EndTime": "2020-10-30T19:36:41Z",
        "StatusCode": "Successful",
        "Progress": 100,
        "Details": "{\"Subnet ID\": \"subnet-5ea0c127\", \"Availability Zone\":
        \\\"us-west-2b\\\"}"
    }
]
}

```

Si la salida incluye un campo `NextToken`, hay más actividades. Para obtener las actividades adicionales, utilice el valor de este campo con la opción `--starting-token` en una llamada posterior de la siguiente manera.

```

aws autoscaling describe-scaling-activities \
  --starting-token Z3M3LMPEXAMPLE

```

Para obtener más información, consulte [Verificar una actividad de escalado para un grupo de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DescribeScalingActivities](#) la Referencia de AWS CLI comandos.

## describe-scaling-process-types

En el siguiente ejemplo de código, se muestra cómo usar `describe-scaling-process-types`.

AWS CLI

Para describir los tipos de procesos disponibles

En este ejemplo se describen los tipos de procesos disponibles.

```

aws autoscaling describe-scaling-process-types

```

Salida:

```

{
  "Processes": [
    {
      "ProcessName": "AZRebalance"
    }
  ]
}

```

```
    },
    {
      "ProcessName": "AddToLoadBalancer"
    },
    {
      "ProcessName": "AlarmNotification"
    },
    {
      "ProcessName": "HealthCheck"
    },
    {
      "ProcessName": "InstanceRefresh"
    },
    {
      "ProcessName": "Launch"
    },
    {
      "ProcessName": "ReplaceUnhealthy"
    },
    {
      "ProcessName": "ScheduledActions"
    },
    {
      "ProcessName": "Terminate"
    }
  ]
}
```

Para obtener más información, consulte [Suspender y reanudar los procesos de escalado](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte la Referencia [DescribeScalingProcessTypes](#) de AWS CLI comandos.

## **describe-scheduled-actions**

En el siguiente ejemplo de código, se muestra cómo usar `describe-scheduled-actions`.

### AWS CLI

Ejemplo 1: Para describir todas las acciones programadas

En este ejemplo se describen todas las acciones programadas.

```
aws autoscaling describe-scheduled-actions
```

Salida:

```
{
  "ScheduledUpdateGroupActions": [
    {
      "AutoScalingGroupName": "my-asg",
      "ScheduledActionName": "my-recurring-action",
      "Recurrence": "30 0 1 1,6,12 *",
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
      "StartTime": "2023-12-01T04:00:00Z",
      "Time": "2023-12-01T04:00:00Z",
      "MinSize": 1,
      "MaxSize": 6,
      "DesiredCapacity": 4,
      "TimeZone": "America/New_York"
    }
  ]
}
```

Para obtener más información, consulte [Escalado programado](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Para describir las acciones programadas para el grupo especificado

Para describir las acciones programadas para un grupo de Auto Scaling específico, utilice la `--auto-scaling-group-name` opción.

```
aws autoscaling describe-scheduled-actions \
  --auto-scaling-group-name my-asg
```

Salida:

```
{
  "ScheduledUpdateGroupActions": [
    {
      "AutoScalingGroupName": "my-asg",
      "ScheduledActionName": "my-recurring-action",
```

```

        "Recurrence": "30 0 1 1,6,12 *",
        "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
        "StartTime": "2023-12-01T04:00:00Z",
        "Time": "2023-12-01T04:00:00Z",
        "MinSize": 1,
        "MaxSize": 6,
        "DesiredCapacity": 4,
        "TimeZone": "America/New_York"
    }
]
}

```

Para obtener más información, consulte [Escalado programado](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 3: Para describir la acción programada especificada

Para describir una acción programada específica, utilice la `--scheduled-action-names` opción.

```

aws autoscaling describe-scheduled-actions \
  --scheduled-action-names my-recurring-action

```

Salida:

```

{
  "ScheduledUpdateGroupActions": [
    {
      "AutoScalingGroupName": "my-asg",
      "ScheduledActionName": "my-recurring-action",
      "Recurrence": "30 0 1 1,6,12 *",
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
      "StartTime": "2023-12-01T04:00:00Z",
      "Time": "2023-12-01T04:00:00Z",
      "MinSize": 1,
      "MaxSize": 6,
      "DesiredCapacity": 4,
      "TimeZone": "America/New_York"
    }
  ]
}

```

```
]
}
```

Para obtener más información, consulte [Escalado programado](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 4: Para describir las acciones programadas con una hora de inicio específica

Para describir las acciones programadas que comienzan a una hora específica, utilice la `--start-time` opción.

```
aws autoscaling describe-scheduled-actions \
  --start-time "2023-12-01T04:00:00Z"
```

Salida:

```
{
  "ScheduledUpdateGroupActions": [
    {
      "AutoScalingGroupName": "my-asg",
      "ScheduledActionName": "my-recurring-action",
      "Recurrence": "30 0 1 1,6,12 *",
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
      "StartTime": "2023-12-01T04:00:00Z",
      "Time": "2023-12-01T04:00:00Z",
      "MinSize": 1,
      "MaxSize": 6,
      "DesiredCapacity": 4,
      "TimeZone": "America/New_York"
    }
  ]
}
```

Para obtener más información, consulte [Escalado programado](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 5: Para describir las acciones programadas que finalizan a una hora específica

Para describir las acciones programadas que finalizan a una hora específica, usa la `--end-time` opción.

```
aws autoscaling describe-scheduled-actions \
  --end-time "2023-12-01T04:00:00Z"
```

Salida:

```
{
  "ScheduledUpdateGroupActions": [
    {
      "AutoScalingGroupName": "my-asg",
      "ScheduledActionName": "my-recurring-action",
      "Recurrence": "30 0 1 1,6,12 *",
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
      "StartTime": "2023-12-01T04:00:00Z",
      "Time": "2023-12-01T04:00:00Z",
      "MinSize": 1,
      "MaxSize": 6,
      "DesiredCapacity": 4,
      "TimeZone": "America/New_York"
    }
  ]
}
```

Para obtener más información, consulte [Escalado programado](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 6: Para describir un número específico de acciones programadas

Para devolver un número específico de acciones programadas, utilice la `--max-items` opción.

```
aws autoscaling describe-scheduled-actions \
  --auto-scaling-group-name my-asg \
  --max-items 1
```

Salida:

```
{
  "ScheduledUpdateGroupActions": [
    {
      "AutoScalingGroupName": "my-asg",
      "ScheduledActionName": "my-recurring-action",
```



```

        "Recurrence": "30 0 1 1,6,12 *",
        "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
        "StartTime": "2023-12-01T04:00:00Z",
        "Time": "2023-12-01T04:00:00Z",
        "MinSize": 1,
        "MaxSize": 6,
        "DesiredCapacity": 4,
        "TimeZone": "America/New_York"
    }
]
}

```

Si la salida incluye un NextToken campo, hay más acciones programadas. Para obtener las acciones programadas adicionales, utilice el valor de este campo con la `--starting-token` opción en una llamada posterior, de la siguiente manera.

```

aws autoscaling describe-scheduled-actions \
  --auto-scaling-group-name my-asg \
  --starting-token Z3M3LMPEXAMPLE

```

Para obtener más información, consulte [Escalado programado](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DescribeScheduledActions](#) la Referencia de AWS CLI comandos.

## describe-tags

En el siguiente ejemplo de código, se muestra cómo usar `describe-tags`.

### AWS CLI

Para describir todas las etiquetas

En este ejemplo se describen todas las etiquetas.

```

aws autoscaling describe-tags

```

Salida:

```
{
  "Tags": [
    {
      "ResourceType": "auto-scaling-group",
      "ResourceId": "my-asg",
      "PropagateAtLaunch": true,
      "Value": "Research",
      "Key": "Dept"
    },
    {
      "ResourceType": "auto-scaling-group",
      "ResourceId": "my-asg",
      "PropagateAtLaunch": true,
      "Value": "WebServer",
      "Key": "Role"
    }
  ]
}
```

Para obtener más información, consulte [Etiquetado de grupos e instancias de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Para describir las etiquetas de un grupo específico

Para describir las etiquetas de un grupo de Auto Scaling específico, utilice la `--filters` opción.

```
aws autoscaling describe-tags --filters Name=auto-scaling-group,Values=my-asg
```

Para obtener más información, consulte [Etiquetado de grupos e instancias de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 3: Para describir el número especificado de etiquetas

Para devolver un número específico de etiquetas, utilice la `--max-items` opción.

```
aws autoscaling describe-tags \  
  --max-items 1
```

Si la salida incluye un `NextToken` campo, hay más etiquetas. Para obtener las etiquetas adicionales, utilice el valor de este campo con la `--starting-token` opción en una llamada posterior, de la siguiente manera.

```
aws autoscaling describe-tags \  
  --filters Name=auto-scaling-group,Values=my-asg \  
  --starting-token Z3M3LMPEXAMPLE
```

Para obtener más información, consulte [Etiquetado de grupos e instancias de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DescribeTags](#) la Referencia de AWS CLI comandos.

## describe-termination-policy-types

En el siguiente ejemplo de código, se muestra cómo usar `describe-termination-policy-types`.

AWS CLI

Para describir los tipos de políticas de rescisión disponibles

En este ejemplo se describen los tipos de políticas de rescisión disponibles.

```
aws autoscaling describe-termination-policy-types
```

Salida:

```
{  
  "TerminationPolicyTypes": [  
    "AllocationStrategy",  
    "ClosestToNextInstanceHour",  
    "Default",  
    "NewestInstance",  
    "OldestInstance",  
    "OldestLaunchConfiguration",  
    "OldestLaunchTemplate"  
  ]  
}
```

Para obtener más información, consulte [Controlar qué instancias de Auto Scaling finalizan durante el escalado en](#) la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DescribeTerminationPolicyTypes](#) la Referencia de AWS CLI comandos.

## describe-warm-pool

En el siguiente ejemplo de código, se muestra cómo usar `describe-warm-pool`.

### AWS CLI

Para describir una piscina caliente

En este ejemplo se describe la piscina caliente del grupo de Auto Scaling especificado.

```
aws autoscaling describe-warm-pool \  
  --auto-scaling-group-name my-asg
```

Salida:

```
{  
  "WarmPoolConfiguration": {  
    "MinSize": 2,  
    "PoolState": "Stopped"  
  },  
  "Instances": [  
    {  
      "InstanceId": "i-070a5bbc7e7f40dc5",  
      "InstanceType": "t2.micro",  
      "AvailabilityZone": "us-west-2c",  
      "LifecycleState": "Warmed:Pending",  
      "HealthStatus": "Healthy",  
      "LaunchTemplate": {  
        "LaunchTemplateId": "lt-00a731f6e9fa48610",  
        "LaunchTemplateName": "my-template-for-auto-scaling",  
        "Version": "6"  
      }  
    },  
    {  
      "InstanceId": "i-0b52f061814d3bd2d",  
      "InstanceType": "t2.micro",  
      "AvailabilityZone": "us-west-2b",  
      "LifecycleState": "Warmed:Pending",  
      "HealthStatus": "Healthy",  
      "LaunchTemplate": {  
        "LaunchTemplateId": "lt-00a731f6e9fa48610",  
        "LaunchTemplateName": "my-template-for-auto-scaling",  
        "Version": "6"  
      }  
    }  
  ]  
}
```

```

    }
  }
]
}

```

Para obtener más información, consulte [Warm Pools for Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DescribeWarmPool](#) la Referencia de AWS CLI comandos.

## detach-instances

En el siguiente ejemplo de código, se muestra cómo usar `detach-instances`.

### AWS CLI

Para separar una instancia de un grupo de Auto Scaling

En este ejemplo, se separa la instancia especificada del grupo de Auto Scaling especificado.

```

aws autoscaling detach-instances \
  --instance-ids i-030017cfa84b20135 \
  --auto-scaling-group-name my-asg \
  --should-decrement-desired-capacity

```

Salida:

```

{
  "Activities": [
    {
      "ActivityId": "5091cb52-547a-47ce-a236-c9ccbc2cb2c9",
      "AutoScalingGroupName": "my-asg",
      "Description": "Detaching EC2 instance: i-030017cfa84b20135",
      "Cause": "At 2020-10-31T17:35:04Z instance i-030017cfa84b20135 was detached in response to a user request, shrinking the capacity from 2 to 1.",
      "StartTime": "2020-04-12T15:02:16.179Z",
      "StatusCode": "InProgress",
      "Progress": 50,
      "Details": "{\"Subnet ID\": \"subnet-6194ea3b\", \"Availability Zone\": \"us-west-2c\"}"
    }
  ]
}

```

```
]
}
```

- Para API obtener más información, consulte [DetachInstances](#) la Referencia de AWS CLI comandos.

## detach-load-balancer-target-groups

En el siguiente ejemplo de código, se muestra cómo usar `detach-load-balancer-target-groups`.

AWS CLI

Para separar un grupo objetivo del balanceador de carga de un grupo de Auto Scaling

En este ejemplo, se separa el grupo objetivo del balanceador de cargas especificado del grupo de Auto Scaling especificado.

```
aws autoscaling detach-load-balancer-target-groups \
  --auto-scaling-group-name my-asg \
  --target-group-arns arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Este comando no produce ningún resultado

Para obtener más información, consulte [Adjuntar un balanceador de carga a su grupo de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte la Referencia [DetachLoadBalancerTargetGroups](#) de AWS CLI comandos.

## detach-load-balancers

En el siguiente ejemplo de código, se muestra cómo usar `detach-load-balancers`.

AWS CLI

Para separar un Classic Load Balancer de un grupo de Auto Scaling

En este ejemplo, se separa el Classic Load Balancer especificado del grupo de Auto Scaling especificado.

```
aws autoscaling detach-load-balancers \  
  --load-balancer-names my-load-balancer \  
  --auto-scaling-group-name my-asg
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Adjuntar un balanceador de carga a su grupo de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte la Referencia [DetachLoadBalancers](#) de AWS CLI comandos.

## **disable-metrics-collection**

En el siguiente ejemplo de código, se muestra cómo usar `disable-metrics-collection`.

### AWS CLI

Desactivación de la recopilación de métricas de grupo de un grupo de escalado automático

En este ejemplo, se deshabilita la recopilación de la métrica `GroupDesiredCapacity` del grupo de escalado automático especificado.

```
aws autoscaling disable-metrics-collection \  
  --auto-scaling-group-name my-asg \  
  --metrics GroupDesiredCapacity
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Supervisión de CloudWatch las métricas de sus grupos e instancias de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [DisableMetricsCollection](#) la Referencia de AWS CLI comandos.

## **enable-metrics-collection**

En el siguiente ejemplo de código, se muestra cómo usar `enable-metrics-collection`.

### AWS CLI

Ejemplo 1: Habilitación de la recopilación de métricas de un grupo de escalado automático

En este ejemplo, se habilita la recopilación de datos para el grupo de escalado automático especificado.

```
aws autoscaling enable-metrics-collection \  
  --auto-scaling-group-name my-asg \  
  --granularity "1Minute"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Supervisión de CloudWatch las métricas de sus grupos e instancias de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Recopilación de datos para la métrica especificada para un grupo de escalado automático

Para recopilar datos para una métrica específica, utilice la opción `--metrics`.

```
aws autoscaling enable-metrics-collection \  
  --auto-scaling-group-name my-asg \  
  --metrics GroupDesiredCapacity --granularity "1Minute"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Supervisión de CloudWatch las métricas de sus grupos e instancias de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [EnableMetricsCollection](#) la Referencia de AWS CLI comandos.

## **enter-standby**

En el siguiente ejemplo de código, se muestra cómo usar `enter-standby`.

### AWS CLI

Para mover las instancias al modo de espera

En este ejemplo, se pone la instancia especificada en modo de espera. Esto resulta útil para actualizar o solucionar problemas de una instancia que está actualmente en servicio.

```
aws autoscaling enter-standby \  
  --instance-ids i-061c63c5eb45f0416 \  
  --
```



```
--auto-scaling-group-name my-asg \  
--should-decrement-desired-capacity
```

Salida:

```
{  
  "Activities": [  
    {  
      "ActivityId": "ffa056b4-6ed3-41ba-ae7c-249dfae6eba1",  
      "AutoScalingGroupName": "my-asg",  
      "Description": "Moving EC2 instance to Standby: i-061c63c5eb45f0416",  
      "Cause": "At 2020-10-31T20:31:00Z instance i-061c63c5eb45f0416 was moved  
to standby in response to a user request, shrinking the capacity from 1 to 0.",  
      "StartTime": "2020-10-31T20:31:00.949Z",  
      "StatusCode": "InProgress",  
      "Progress": 50,  
      "Details": "{\"Subnet ID\": \"subnet-6194ea3b\", \"Availability Zone\":  
\"us-west-2c\"}"  
    }  
  ]  
}
```

Para obtener más información, consulte el [ciclo de vida de las instancias de Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [EnterStandby](#) la Referencia de AWS CLI comandos.

## execute-policy

En el siguiente ejemplo de código, se muestra cómo usar `execute-policy`.

### AWS CLI

Para ejecutar una política de escalado

Este ejemplo ejecuta la política de escalado denominada `my-step-scale-out-policy` para el grupo de Auto Scaling especificado.

```
aws autoscaling execute-policy \  
  --auto-scaling-group-name my-asg \  
  --policy-name my-step-scale-out-policy \  
  --
```

```
--metric-value 95 \  
--breach-threshold 80
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Políticas de escalado simples y escalonadas](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [ExecutePolicy](#) la Referencia de AWS CLI comandos.

## exit-standby

En el siguiente ejemplo de código, se muestra cómo usar `exit-standby`.

### AWS CLI

Para sacar las instancias del modo de espera

En este ejemplo, se saca la instancia especificada del modo de espera.

```
aws autoscaling exit-standby \  
--instance-ids i-061c63c5eb45f0416 \  
--auto-scaling-group-name my-asg
```

Salida:

```
{  
  "Activities": [  
    {  
      "ActivityId": "142928e1-a2dc-453a-9b24-b85ad6735928",  
      "AutoScalingGroupName": "my-asg",  
      "Description": "Moving EC2 instance out of Standby:  
i-061c63c5eb45f0416",  
      "Cause": "At 2020-10-31T20:32:50Z instance i-061c63c5eb45f0416 was moved  
out of standby in response to a user request, increasing the capacity from 0 to  
1.",  
      "StartTime": "2020-10-31T20:32:50.222Z",  
      "StatusCode": "PreInService",  
      "Progress": 30,  
      "Details": "{\"Subnet ID\":\"subnet-6194ea3b\",\"Availability Zone\":  
\"us-west-2c\"}"  
    }  
  ]  
}
```

```
]
}
```

Para obtener más información, consulte [Eliminar temporalmente instancias de su grupo de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [ExitStandby](#) la Referencia de AWS CLI comandos.

## put-lifecycle-hook

En el siguiente ejemplo de código, se muestra cómo usar `put-lifecycle-hook`.

### AWS CLI

Ejemplo 1: Para crear un enlace de ciclo de vida

En este ejemplo, se crea un enlace de ciclo de vida que se invocará en cualquier instancia recién lanzada, con un tiempo de espera de 4800 segundos. Esto resulta útil para mantener las instancias en estado de espera hasta que finalicen los scripts de datos de usuario o para invocar una función AWS Lambda mediante EventBridge.

```
aws autoscaling put-lifecycle-hook \
  --auto-scaling-group-name my-asg \
  --lifecycle-hook-name my-launch-hook \
  --lifecycle-transition autoscaling:EC2_INSTANCE_LAUNCHING \
  --heartbeat-timeout 4800
```

Este comando no genera ninguna salida. Si ya existe un enlace de ciclo de vida con el mismo nombre, el nuevo enlace de ciclo de vida lo sobrescribirá.

Para obtener más información, consulte los [enlaces del ciclo de vida de Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Para enviar un mensaje de SNS correo electrónico de Amazon para notificarle las transiciones de estado de la instancia

En este ejemplo, se crea un enlace de ciclo de vida con el SNS tema y el IAM rol de Amazon que se usará para recibir notificaciones en el lanzamiento de la instancia.

```
aws autoscaling put-lifecycle-hook \
  --auto-scaling-group-name my-asg \
  --lifecycle-hook-name my-launch-hook \
```

```
--lifecycle-transition autoscaling:EC2_INSTANCE_LAUNCHING \  
--notification-target-arn arn:aws:sns:us-west-2:123456789012:my-sns-topic \  
--role-arn arn:aws:iam::123456789012:role/my-auto-scaling-role
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [enlaces del ciclo de vida de Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 3: Para publicar un mensaje en una SQS cola de Amazon

En este ejemplo, se crea un enlace de ciclo de vida que publica un mensaje con metadatos en la SQS cola de Amazon especificada.

```
aws autoscaling put-lifecycle-hook \  
  --auto-scaling-group-name my-asg \  
  --lifecycle-hook-name my-launch-hook \  
  --lifecycle-transition autoscaling:EC2_INSTANCE_LAUNCHING \  
  --notification-target-arn arn:aws:sqs:us-west-2:123456789012:my-sqs-queue \  
  --role-arn arn:aws:iam::123456789012:role/my-notification-role \  
  --notification-metadata "SQS message metadata"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [enlaces del ciclo de vida de Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [PutLifecycleHook](#) la Referencia de AWS CLI comandos.

## put-notification-configuration

En el siguiente ejemplo de código, se muestra cómo usar put-notification-configuration.

### AWS CLI

Para añadir una notificación

En este ejemplo, se agrega la notificación especificada al grupo de Auto Scaling especificado.

```
aws autoscaling put-notification-configuration \  
  --auto-scaling-group-name my-asg \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:my-sns-topic \  
  --notification-configuration my-notification-configuration
```

```
--notification-type autoscaling:TEST_NOTIFICATION
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cómo recibir SNS notificaciones de Amazon cuando su grupo de Auto Scaling se amplía](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [PutNotificationConfiguration](#) la Referencia de AWS CLI comandos.

## put-scaling-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-scaling-policy`.

### AWS CLI

Para agregar una política de escalado de seguimiento de objetivos a un grupo de Auto Scaling

El siguiente `put-scaling-policy` ejemplo aplica una política de escalado de seguimiento de objetivos al grupo de Auto Scaling especificado. El resultado contiene los nombres ARNs y los nombres de las dos CloudWatch alarmas creadas en su nombre. Si ya existe una política de escalado con el mismo nombre, la nueva política de escalado la sobrescribirá.

```
aws autoscaling put-scaling-policy --auto-scaling-group-name my-asg \  
  --policy-name alb1000-target-tracking-scaling-policy \  
  --policy-type TargetTrackingScaling \  
  --target-tracking-configuration file://config.json
```

Contenidos de `config.json`:

```
{  
  "TargetValue": 1000.0,  
  "PredefinedMetricSpecification": {  
    "PredefinedMetricType": "ALBRequestCountPerTarget",  
    "ResourceLabel": "app/my-alb/778d41231b141a0f/targetgroup/my-alb-target-  
group/943f017f100becff"  
  }  
}
```

Salida:

```
{
```

```

    "PolicyARN": "arn:aws:autoscaling:region:account-id:scalingPolicy:228f02c2-
c665-4bfd-aaac-8b04080bea3c:autoScalingGroupName/my-asg:policyName/alb1000-target-
tracking-scaling-policy",
    "Alarms": [
        {
            "AlarmARN": "arn:aws:cloudwatch:region:account-id:alarm:TargetTracking-
my-asg-AlarmHigh-fc0e4183-23ac-497e-9992-691c9980c38e",
            "AlarmName": "TargetTracking-my-asg-AlarmHigh-
fc0e4183-23ac-497e-9992-691c9980c38e"
        },
        {
            "AlarmARN": "arn:aws:cloudwatch:region:account-id:alarm:TargetTracking-
my-asg-AlarmLow-61a39305-ed0c-47af-bd9e-471a352ee1a2",
            "AlarmName": "TargetTracking-my-asg-AlarmLow-61a39305-ed0c-47af-
bd9e-471a352ee1a2"
        }
    ]
}

```

Para ver más ejemplos, consulte [Ejemplos de políticas de escalado para la interfaz de línea de AWS comandos \(AWS CLI\)](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [PutScalingPolicy](#) la Referencia de AWS CLI comandos.

## put-scheduled-update-group-action

En el siguiente ejemplo de código, se muestra cómo usar `put-scheduled-update-group-action`.

AWS CLI

Ejemplo 1: Para agregar una acción programada a un grupo de Auto Scaling

Este ejemplo agrega la acción programada especificada al grupo de Auto Scaling especificado.

```

aws autoscaling put-scheduled-update-group-action \
  --auto-scaling-group-name my-asg \
  --scheduled-action-name my-scheduled-action \
  --start-time "2023-05-12T08:00:00Z" \
  --min-size 2 \
  --max-size 6 \

```

```
--desired-capacity 4
```

Este comando no genera ninguna salida. Si ya existe una acción programada con el mismo nombre, la nueva acción programada la sobrescribirá.

Para ver más ejemplos, consulte el [escalado programado](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Para especificar un programa periódico

En este ejemplo, se crea una acción programada para adaptarla a un programa recurrente cuya ejecución está prevista para las 00:30 horas del primero de enero, junio y diciembre de cada año.

```
aws autoscaling put-scheduled-update-group-action \  
  --auto-scaling-group-name my-asg \  
  --scheduled-action-name my-recurring-action \  
  --recurrence "30 0 1 1,6,12 *" \  
  --min-size 2 \  
  --max-size 6 \  
  --desired-capacity 4
```

Este comando no genera ninguna salida. Si ya existe una acción programada con el mismo nombre, la nueva acción programada la sobrescribirá.

Para ver más ejemplos, consulte el [escalado programado](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [PutScheduledUpdateGroupAction](#) la Referencia de AWS CLI comandos.

## put-warm-pool

En el siguiente ejemplo de código, se muestra cómo usar `put-warm-pool`.

### AWS CLI

Para crear una piscina caliente

El siguiente ejemplo crea una piscina caliente para el grupo de Auto Scaling especificado.

```
aws autoscaling put-warm-pool \  
  --auto-scaling-group-name my-asg \  
  --desired-capacity 4
```

```
--min-size 2
```

Este comando no genera ninguna salida. Si ya existe una piscina caliente, se actualizará.

Para obtener más información, consulte [Warm Pools for Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [PutWarmPool](#) la Referencia de AWS CLI comandos.

## **record-lifecycle-action-heartbeat**

En el siguiente ejemplo de código, se muestra cómo usar `record-lifecycle-action-heartbeat`.

### AWS CLI

Para registrar el latido de una acción del ciclo de vida

En este ejemplo, se registra el latido de una acción del ciclo de vida para mantener la instancia en estado pendiente.

```
aws autoscaling record-lifecycle-action-heartbeat \  
  --lifecycle-hook-name my-launch-hook \  
  --auto-scaling-group-name my-asg \  
  --lifecycle-action-token bcd2f1b8-9a78-44d3-8a7a-4dd07d7cf635
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [enlaces del ciclo de vida de Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [RecordLifecycleActionHeartbeat](#) la Referencia de AWS CLI comandos.

## **resume-processes**

En el siguiente ejemplo de código, se muestra cómo usar `resume-processes`.

### AWS CLI

Para reanudar los procesos suspendidos



En este ejemplo, se reanuda el proceso de escalado suspendido especificado para el grupo de Auto Scaling especificado.

```
aws autoscaling resume-processes \  
  --auto-scaling-group-name my-asg \  
  --scaling-processes AlarmNotification
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Suspender y reanudar los procesos de escalado](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte la Referencia [ResumeProcesses](#) de AWS CLI comandos.

## rollback-instance-refresh

En el siguiente ejemplo de código, se muestra cómo usar `rollback-instance-refresh`.

### AWS CLI

Para revertir una actualización de instancias

El siguiente `rollback-instance-refresh` ejemplo revierte una actualización de instancias en curso para el grupo de Auto Scaling especificado.

```
aws autoscaling rollback-instance-refresh \  
  --auto-scaling-group-name my-asg
```

Salida:

```
{  
  "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b"  
}
```

Para obtener más información, consulte [Deshacer cambios con una reversión](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [RollbackInstanceRefresh](#) la Referencia de AWS CLI comandos.

## set-desired-capacity

En el siguiente ejemplo de código, se muestra cómo usar `set-desired-capacity`.

### AWS CLI

Fijación de la capacidad deseada inicial para el grupo de escalado automático

En este ejemplo, se establece la capacidad deseada para el grupo de escalado automático especificado.

```
aws autoscaling set-desired-capacity \  
  --auto-scaling-group-name my-asg \  
  --desired-capacity 2 \  
  --honor-cooldown
```

Este comando vuelve a la petición si se ejecuta correctamente.

- Para API obtener más información, consulte [SetDesiredCapacity](#) la Referencia de AWS CLI comandos.

## set-instance-health

En el siguiente ejemplo de código, se muestra cómo usar `set-instance-health`.

### AWS CLI

Para establecer el estado de salud de una instancia

En este ejemplo, se establece el estado de salud de la instancia especificada en `Unhealthy`.

```
aws autoscaling set-instance-health \  
  --instance-id i-061c63c5eb45f0416 \  
  --health-status Unhealthy
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SetInstanceHealth](#) la Referencia de AWS CLI comandos.

## set-instance-protection

En el siguiente ejemplo de código, se muestra cómo usar `set-instance-protection`.

## AWS CLI

Ejemplo 1: Para habilitar la configuración de protección de instancias para una instancia

En este ejemplo, se habilita la protección de instancias para la instancia especificada.

```
aws autoscaling set-instance-protection \  
  --instance-ids i-061c63c5eb45f0416 \  
  --auto-scaling-group-name my-asg --protected-from-scale-in
```

Este comando no genera ninguna salida.

Ejemplo 2: Para deshabilitar la configuración de protección de instancias para una instancia

En este ejemplo, se inhabilita la protección de instancias para la instancia especificada.

```
aws autoscaling set-instance-protection \  
  --instance-ids i-061c63c5eb45f0416 \  
  --auto-scaling-group-name my-asg \  
  --no-protected-from-scale-in
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SetInstanceProtection](#) la Referencia de AWS CLI comandos.

## start-instance-refresh

En el siguiente ejemplo de código, se muestra cómo usar `start-instance-refresh`.

## AWS CLI

Ejemplo 1: Para iniciar una actualización de instancias mediante parámetros de línea de comandos

En el siguiente `start-instance-refresh` ejemplo, se inicia una actualización de instancias mediante argumentos de línea de comandos. El `preferences` parámetro opcional especifica un `InstanceWarmup` número de 60 segundos y un `MinHealthyPercentage` 50 porcentaje.

```
aws autoscaling start-instance-refresh \  
  --auto-scaling-group-name my-asg \  
  --preferences
```

```
--preferences '{"InstanceWarmup": 60, "MinHealthyPercentage": 50}'
```

Salida:

```
{
  "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b"
}
```

Para obtener más información, consulte [Iniciar una actualización de instancias](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Para iniciar la actualización de una instancia mediante un JSON archivo

En el siguiente `start-instance-refresh` ejemplo, se inicia una actualización de instancias mediante un JSON archivo. Puede especificar el grupo Auto Scaling y definir la configuración y las preferencias que desee en un JSON archivo, como se muestra en el siguiente ejemplo.

```
aws autoscaling start-instance-refresh \
  --cli-input-json file://config.json
```

Contenidos de `config.json`:

```
{
  "AutoScalingGroupName": "my-asg",
  "DesiredConfiguration": {
    "LaunchTemplate": {
      "LaunchTemplateId": "lt-068f72b729example",
      "Version": "$Default"
    }
  },
  "Preferences": {
    "InstanceWarmup": 60,
    "MinHealthyPercentage": 50,
    "AutoRollback": true,
    "ScaleInProtectedInstances": Ignore,
    "StandbyInstances": Terminate
  }
}
```

Salida:

```
{
  "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b"
}
```

Para obtener más información, consulte [Iniciar una actualización de instancias](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [StartInstanceRefresh](#) la Referencia de AWS CLI comandos.

## suspend-processes

En el siguiente ejemplo de código, se muestra cómo usar `suspend-processes`.

### AWS CLI

Para suspender los procesos de Auto Scaling

Este ejemplo suspende el proceso de escalado especificado para el grupo de Auto Scaling especificado.

```
aws autoscaling suspend-processes \
  --auto-scaling-group-name my-asg \
  --scaling-processes AlarmNotification
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Suspender y reanudar los procesos de escalado](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte la Referencia [SuspendProcesses](#) de AWS CLI comandos.

## terminate-instance-in-auto-scaling-group

En el siguiente ejemplo de código, se muestra cómo usar `terminate-instance-in-auto-scaling-group`.

### AWS CLI

Finalización de una instancia en su grupo de escalado automático

En este ejemplo, se termina la instancia especificada del grupo de escalado automático especificado sin actualizar el tamaño del grupo. Amazon EC2 Auto Scaling lanza una instancia de reemplazo después de que finalice la instancia especificada.

```
aws autoscaling terminate-instance-in-auto-scaling-group \  
--instance-id i-061c63c5eb45f0416 \  
--no-should-decrement-desired-capacity
```

Salida:

```
{  
  "Activities": [  
    {  
      "ActivityId": "8c35d601-793c-400c-fcd0-f64a27530df7",  
      "AutoScalingGroupName": "my-asg",  
      "Description": "Terminating EC2 instance: i-061c63c5eb45f0416",  
      "Cause": "",  
      "StartTime": "2020-10-31T20:34:25.680Z",  
      "StatusCode": "InProgress",  
      "Progress": 0,  
      "Details": "{\"Subnet ID\": \"subnet-6194ea3b\", \"Availability Zone\":  
\"us-west-2c\"}"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [TerminateInstanceInAutoScalingGroup](#) la Referencia de AWS CLI comandos.

## update-auto-scaling-group

En el siguiente ejemplo de código, se muestra cómo usar `update-auto-scaling-group`.

AWS CLI

Ejemplo 1: Actualización de los límites de tamaño de un grupo de escalado automático

En este ejemplo, se actualiza el grupo de escalado automático especificado con un tamaño mínimo de 2 y un tamaño máximo de 10.

```
aws autoscaling update-auto-scaling-group \  

```

```
--auto-scaling-group-name my-asg \  
--min-size 2 \  
--max-size 10
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Establecer límites de capacidad para su grupo de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 2: Adición de comprobaciones de estado de Elastic Load Balancing y especificar qué zonas de disponibilidad y subredes se deben utilizar

En este ejemplo, se actualiza el grupo de escalado automático especificado para añadir comprobaciones de estado de Elastic Load Balancing. Este comando también actualiza el valor de `--vpc-zone-identifier` con una lista de subredes IDs en varias zonas de disponibilidad.

```
aws autoscaling update-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --health-check-type ELB \  
  --health-check-grace-period 600 \  
  --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Elastic Load Balancing y Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

Ejemplo 3: Actualización del grupo con ubicación y la política de terminación

En este ejemplo, se actualizan el grupo con ubicación y la política de terminación que se van a utilizar.

```
aws autoscaling update-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --placement-group my-placement-group \  
  --termination-policies "OldestInstance"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [grupos de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

#### Ejemplo 4: Uso de la versión más reciente de la plantilla de lanzamiento

En este ejemplo, se actualiza el grupo de escalado automático para utilizar la versión más reciente de la plantilla de lanzamiento.

```
aws autoscaling update-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --launch-template LaunchTemplateId=lt-1234567890abcde12,Version='$Latest'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Plantillas de lanzamiento](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

#### Ejemplo 5: Uso de una versión específica de la plantilla de lanzamiento

En este ejemplo, se actualiza el grupo de escalado automático especificado para utilizar una versión específica de una plantilla de lanzamiento en lugar de la versión más reciente o predeterminada.

```
aws autoscaling update-auto-scaling-group \  
  --auto-scaling-group-name my-asg \  
  --launch-template LaunchTemplateName=my-template-for-auto-scaling,Version='2'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Plantillas de lanzamiento](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

#### Ejemplo 6: Definición de una política de instancias mixtas y habilitar el reequilibrio de la capacidad

En este ejemplo, se actualiza el grupo de escalado automático especificado para que utilice una política de instancias mixtas y se habilita el reequilibrio de la capacidad. Esta estructura le permite especificar grupos con capacidades bajo demanda y de spot y utilizar distintas plantillas de lanzamiento para diferentes arquitecturas.

```
aws autoscaling update-auto-scaling-group \  
  --cli-input-json file://~/config.json
```

Contenidos de config.json:

```
{
```



```
"AutoScalingGroupName": "my-asg",
"CapacityRebalance": true,
"MixedInstancesPolicy": {
  "LaunchTemplate": {
    "LaunchTemplateSpecification": {
      "LaunchTemplateName": "my-launch-template-for-x86",
      "Version": "$Latest"
    },
    "Overrides": [
      {
        "InstanceType": "c6g.large",
        "LaunchTemplateSpecification": {
          "LaunchTemplateName": "my-launch-template-for-arm",
          "Version": "$Latest"
        }
      },
      {
        "InstanceType": "c5.large"
      },
      {
        "InstanceType": "c5a.large"
      }
    ]
  },
  "InstancesDistribution": {
    "OnDemandPercentageAboveBaseCapacity": 50,
    "SpotAllocationStrategy": "capacity-optimized"
  }
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Grupos de Auto Scaling con varios tipos de instancias y opciones de compra](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

- Para API obtener más información, consulte [UpdateAutoScalingGroup](#) la Referencia de AWS CLI comandos.

## Ejemplos de planes de Auto Scaling utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de los planes AWS Command Line Interface de Auto Scaling.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **create-scaling-plan**

En el siguiente ejemplo de código, se muestra cómo usar `create-scaling-plan`.

#### AWS CLI

Para crear un plan de escalado

En el siguiente `create-scaling-plan` ejemplo, se crea un plan de escalado denominado `my-scaling-plan` mediante un JSON archivo ya creado (denominado `config.json`). La estructura del plan de escalado incluye una instrucción de escalado para un grupo de Auto Scaling denominado `my-asg`. Especifica la propiedad `TagFilters` como el origen de la aplicación y permite el escalado predictivo y el escalado dinámico.

```
aws autoscaling-plans create-scaling-plan \  
  --scaling-plan-name my-scaling-plan \  
  --cli-input-json file://~/config.json
```

Contenido del archivo `config.json`:

```
{  
  "ApplicationSource": {  
    "TagFilters": [  
      {  
        "Key": "purpose",  
        "Values": [  
          "my-application"  
        ]  
      }  
    ]  
  }  
}
```

```

    ]
  },
  "ScalingInstructions": [
    {
      "ServiceNamespace": "autoscaling",
      "ResourceId": "autoScalingGroup/my-asg",
      "ScalableDimension": "autoscaling:autoScalingGroup:DesiredCapacity",
      "ScheduledActionBufferTime": 300,
      "PredictiveScalingMaxCapacityBehavior":
"SetForecastCapacityToMaxCapacity",
      "PredictiveScalingMode": "ForecastAndScale",
      "PredefinedLoadMetricSpecification": {
        "PredefinedLoadMetricType": "ASGTotalCPUUtilization"
      },
      "ScalingPolicyUpdateBehavior": "ReplaceExternalPolicies",
      "MinCapacity": 1,
      "MaxCapacity": 4,
      "TargetTrackingConfigurations": [
        {
          "PredefinedScalingMetricSpecification": {
            "PredefinedScalingMetricType": "ASGAverageCPUUtilization"
          },
          "TargetValue": 50
        }
      ]
    }
  ]
}

```

Salida:

```

{
  "ScalingPlanVersion": 1
}

```

Para obtener más información, consulte la [Guía del usuario de AWS Auto Scaling](#).

- Para API obtener más información, consulte [CreateScalingPlan](#) la Referencia de AWS CLI comandos.

## delete-scaling-plan

En el siguiente ejemplo de código, se muestra cómo usar delete-scaling-plan.

## AWS CLI

Para eliminar un plan de escalado

En el siguiente `delete-scaling-plan` ejemplo, se elimina el plan de escalado especificado.

```
aws autoscaling-plans delete-scaling-plan \  
  --scaling-plan-name my-scaling-plan \  
  --scaling-plan-version 1
```

Este comando no genera ninguna salida.

Para obtener más información, consulte la [Guía del usuario de AWS Auto Scaling](#).

- Para API obtener más información, consulte [DeleteScalingPlan](#) la Referencia de AWS CLI comandos.

## `describe-scaling-plan-resources`

En el siguiente ejemplo de código, se muestra cómo usar `describe-scaling-plan-resources`.

## AWS CLI

Para describir los recursos escalables de un plan de escalado

El siguiente `describe-scaling-plan-resources` ejemplo muestra detalles sobre el único recurso escalable (un grupo de Auto Scaling) que está asociado al plan de escalado especificado.

```
aws autoscaling-plans describe-scaling-plan-resources \  
  --scaling-plan-name my-scaling-plan \  
  --scaling-plan-version 1
```

Salida:

```
{  
  "ScalingPlanResources": [  
    {  
      "ScalableDimension": "autoscaling:autoScalingGroup:DesiredCapacity",  
      "ScalingPlanVersion": 1,  
      "ResourceId": "autoScalingGroup/my-asg",  
      "ScalingStatusCode": "Active",
```

```

    "ScalingStatusMessage": "Target tracking scaling policies have been
    applied to the resource.",
    "ScalingPolicies": [
      {
        "PolicyName": "AutoScaling-my-asg-b1ab65ae-4be3-4634-bd64-
c7471662b251",
        "PolicyType": "TargetTrackingScaling",
        "TargetTrackingConfiguration": {
          "PredefinedScalingMetricSpecification": {
            "PredefinedScalingMetricType":
"ALBRequestCountPerTarget",
            "ResourceLabel": "app/my-alb/f37c06a68c1748aa/
targetgroup/my-target-group/6d4ea56ca2d6a18d"
          },
          "TargetValue": 40.0
        }
      }
    ],
    "ServiceNamespace": "autoscaling",
    "ScalingPlanName": "my-scaling-plan"
  }
]
}

```

Para obtener más información, consulte [¿Qué es AWS Auto Scaling?](#) en la Guía del usuario de AWS Auto Scaling.

- Para API obtener más información, consulte [DescribeScalingPlanResources](#) la Referencia de AWS CLI comandos.

## describe-scaling-plans

En el siguiente ejemplo de código, se muestra cómo usar `describe-scaling-plans`.

### AWS CLI

Para describir un plan de escalado

En el siguiente `describe-scaling-plans` ejemplo, se muestran los detalles del plan de escalado especificado.

```
aws autoscaling-plans describe-scaling-plans \
```

```
--scaling-plan-names scaling-plan-with-asg-and-ddb
```

Salida:

```
{
  "ScalingPlans": [
    {
      "LastMutatingRequestTime": 1565388443.963,
      "ScalingPlanVersion": 1,
      "CreationTime": 1565388443.963,
      "ScalingInstructions": [
        {
          "ScalingPolicyUpdateBehavior": "ReplaceExternalPolicies",
          "ScalableDimension":
"autoscaling:autoScalingGroup:DesiredCapacity",
          "TargetTrackingConfigurations": [
            {
              "PredefinedScalingMetricSpecification": {
                "PredefinedScalingMetricType":
"ASGAverageCPUUtilization"
              },
              "TargetValue": 50.0,
              "EstimatedInstanceWarmup": 300,
              "DisableScaleIn": false
            }
          ],
          "ResourceId": "autoScalingGroup/my-asg",
          "DisableDynamicScaling": false,
          "MinCapacity": 1,
          "ServiceNamespace": "autoscaling",
          "MaxCapacity": 10
        },
        {
          "ScalingPolicyUpdateBehavior": "ReplaceExternalPolicies",
          "ScalableDimension": "dynamodb:table:ReadCapacityUnits",
          "TargetTrackingConfigurations": [
            {
              "PredefinedScalingMetricSpecification": {
                "PredefinedScalingMetricType":
"DynamoDBReadCapacityUtilization"
              },
              "TargetValue": 50.0,
              "ScaleInCooldown": 60,

```

```

        "DisableScaleIn": false,
        "ScaleOutCooldown": 60
    }
],
"ResourceId": "table/my-table",
"DisableDynamicScaling": false,
"MinCapacity": 5,
"ServiceNamespace": "dynamodb",
"MaxCapacity": 10000
},
{
    "ScalingPolicyUpdateBehavior": "ReplaceExternalPolicies",
    "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
    "TargetTrackingConfigurations": [
        {
            "PredefinedScalingMetricSpecification": {
                "PredefinedScalingMetricType":
"DynamoDBWriteCapacityUtilization"
            },
            "TargetValue": 50.0,
            "ScaleInCooldown": 60,
            "DisableScaleIn": false,
            "ScaleOutCooldown": 60
        }
    ],
    "ResourceId": "table/my-table",
    "DisableDynamicScaling": false,
    "MinCapacity": 5,
    "ServiceNamespace": "dynamodb",
    "MaxCapacity": 10000
}
],
"ApplicationSource": {
    "TagFilters": [
        {
            "Values": [
                "my-application-id"
            ],
            "Key": "application"
        }
    ]
}
},
"StatusStartTime": 1565388455.836,
"ScalingPlanName": "scaling-plan-with-asg-and-ddb",

```

```

        "StatusMessage": "Scaling plan has been created and applied to all
resources.",
        "StatusCode": "Active"
    }
]
}

```

Para obtener más información, consulte [¿Qué es AWS Auto Scaling?](#) en la Guía del usuario de AWS Auto Scaling.

- Para API obtener más información, consulte [DescribeScalingPlans](#) la Referencia de AWS CLI comandos.

### get-scaling-plan-resource-forecast-data

En el siguiente ejemplo de código, se muestra cómo usar `get-scaling-plan-resource-forecast-data`.

#### AWS CLI

Para recuperar los datos de previsión de carga

En este ejemplo, se recuperan los datos de previsión de carga de un recurso escalable (un grupo de Auto Scaling) que está asociado al plan de escalado especificado.

```

aws autoscaling-plans get-scaling-plan-resource-forecast-data \
  --scaling-plan-name my-scaling-plan \
  --scaling-plan-version 1 \
  --service-namespace "autoscaling" \
  --resource-id autoScalingGroup/my-asg \
  --scalable-dimension "autoscaling:autoScalingGroup:DesiredCapacity" \
  --forecast-data-type "LoadForecast" \
  --start-time "2019-08-30T00:00:00Z" \
  --end-time "2019-09-06T00:00:00Z"

```

Salida:

```

{
  "Datapoints": [...]
}

```



Para obtener más información, consulte [Qué es AWS Auto Scaling](#) en la Guía del usuario de AWS Auto Scaling.

- Para API obtener más información, consulte [GetScalingPlanResourceForecastData](#) la Referencia de AWS CLI comandos.

## update-scaling-plan

En el siguiente ejemplo de código, se muestra cómo usar `update-scaling-plan`.

### AWS CLI

Para actualizar un plan de escalado

El siguiente `update-scaling-plan` ejemplo modifica la métrica de escalado de un grupo de Auto Scaling en el plan de escalado especificado.

```
aws autoscaling-plans update-scaling-plan \
  --scaling-plan-name my-scaling-plan \
  --scaling-plan-version 1 \
  --scaling-instructions
  '{"ScalableDimension":"autoscaling:autoScalingGroup:DesiredCapacity","ResourceId":"autoScalingGroup/my-asg","ServiceNamespace":"autoscaling","TargetTrackingConfigurations":
  [{"PredefinedScalingMetricSpecification":
  {"PredefinedScalingMetricType":"ALBRequestCountPerTarget","ResourceLabel":"app/my-alb/f37c06a68c1748aa/targetgroup/my-target-group/6d4ea56ca2d6a18d"},"TargetValue":40.0}],"MinCapacity": 1,"MaxCapacity": 10}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [¿Qué es AWS Auto Scaling?](#) en la Guía del usuario de AWS Auto Scaling.

- Para API obtener más información, consulte [UpdateScalingPlan](#) la Referencia de AWS CLI comandos.

## AWS Backup ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Backup.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **create-backup-plan**

En el siguiente ejemplo de código, se muestra cómo usar `create-backup-plan`.

AWS CLI

Para crear un plan de respaldo

En el siguiente `create-backup-plan` ejemplo, se crea el plan de respaldo especificado con una retención de 35 días.

```
aws backup create-backup-plan \
--backup-plan "{\"BackupPlanName\": \"Example-Backup-Plan\", \"Rules\": [{\"RuleName\":
\"DailyBackups\", \"ScheduleExpression\": \"cron(0 5 ? * * *)\", \"StartWindowMinutes
\":480, \"TargetBackupVaultName\": \"Default\", \"Lifecycle\": {\"DeleteAfterDays
\":35}}]}"
```

Salida:

```
{
  "BackupPlanId": "1fa3895c-a7f5-484a-a371-2dd6a1a9f729",
  "BackupPlanArn": "arn:aws:backup:us-west-2:123456789012:backup-plan:1fa3895c-
a7f5-484a-a371-2dd6a1a9f729",
  "CreationDate": 1568928754.747,
  "VersionId": "ZjQ2ZTI5YWQtZDg5Yi00MzYzLWJmZTAzMDE1Mzh1MDhjYjEz"
}
```

Para obtener más información, consulte [Creación de un plan de backup](#) en la Guía para desarrolladores AWS de Backup.

- Para API obtener más información, consulte [CreateBackupPlan](#) la Referencia de AWS CLI comandos.

## create-backup-vault

En el siguiente ejemplo de código, se muestra cómo usar `create-backup-vault`.

### AWS CLI

Para crear un almacén de copias de seguridad

En el siguiente `create-backup-vault` ejemplo, se crea un almacén de copias de seguridad con el nombre especificado.

```
aws backup create-backup-vault
  --backup-vault-name sample-vault
```

Este comando no genera ninguna salida. Salida:

```
{
  "BackupVaultName": "sample-vault",
  "BackupVaultArn": "arn:aws:backup:us-west-2:123456789012:backup-vault:sample-
vault",
  "CreationDate": 1568928338.385
}
```

Para obtener más información, consulte [Creación de un almacén de copias de seguridad](#) en la Guía para desarrolladores AWS de Backup.

- Para API obtener más información, consulte [CreateBackupVault](#) la Referencia de AWS CLI comandos.

## get-backup-plan-from-template

En el siguiente ejemplo de código, se muestra cómo usar `get-backup-plan-from-template`.

### AWS CLI

Para obtener un plan de respaldo existente a partir de una plantilla

En el siguiente `get-backup-plan-from-template` ejemplo, se obtiene un plan de copias de seguridad existente a partir de una plantilla que especifica una copia de seguridad diaria con una retención de 35 días.

```
aws backup get-backup-plan-from-template \  
  --backup-plan-template-id "87c0c1ef-254d-4180-8fef-2e76a2c38aaa"
```

Salida:

```
{  
  "BackupPlanDocument": {  
    "Rules": [  
      {  
        "RuleName": "DailyBackups",  
        "ScheduleExpression": "cron(0 5 ? * * *)",  
        "StartWindowMinutes": 480,  
        "Lifecycle": {  
          "DeleteAfterDays": 35  
        }  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [Creación de un plan de backup](#) en la Guía para desarrolladores AWS de Backup.

- Para API obtener más información, consulte [GetBackupPlanFromTemplate](#) la Referencia de AWS CLI comandos.

## get-backup-plan

En el siguiente ejemplo de código, se muestra cómo usar `get-backup-plan`.

### AWS CLI

Para obtener los detalles de un plan de respaldo

En el siguiente `get-backup-plan` ejemplo, se muestran los detalles del plan de respaldo especificado.

```
aws backup get-backup-plan \  
  --plan-id "87c0c1ef-254d-4180-8fef-2e76a2c38aaa"
```

```
--backup-plan-id "fcbf5d8f-bd77-4f3a-9c97-f24fb3d373a5"
```

Salida:

```
{
  "BackupPlan": {
    "BackupPlanName": "Example-Backup-Plan",
    "Rules": [
      {
        "RuleName": "DailyBackups",
        "TargetBackupVaultName": "Default",
        "ScheduleExpression": "cron(0 5 ? * * *)",
        "StartWindowMinutes": 480,
        "CompletionWindowMinutes": 10080,
        "Lifecycle": {
          "DeleteAfterDays": 35
        },
        "RuleId": "70e0ccdc-e9df-4e83-82ad-c1e5a9471cc3"
      }
    ]
  },
  "BackupPlanId": "fcbf5d8f-bd77-4f3a-9c97-f24fb3d373a5",
  "BackupPlanArn": "arn:aws:backup:us-west-2:123456789012:backup-plan:fcbf5d8f-
bd77-4f3a-9c97-f24fb3d373a5",
  "VersionId": "NjQ2ZTZkODktMGVhNy00MmQ0LWE4YjktZTkWNTQ3OTkyYTcw",
  "CreationDate": 1568926091.57
}
```

Para obtener más información, consulte [Creación de un plan de backup](#) en la Guía para desarrolladores AWS de Backup.

- Para API obtener más información, consulte [GetBackupPlan](#) la Referencia de AWS CLI comandos.

## list-backup-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-backup-jobs`.

AWS CLI

Ejemplo 1: Para enumerar todos los trabajos de copia de seguridad

En el siguiente `list-backup-jobs` ejemplo, se muestran los metadatos sobre los trabajos de copia de seguridad de su AWS cuenta.

```
aws backup list-backup-jobs
```

Salida:

```
{
  "BackupJobs": [
    {
      "BackupJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "BackupVaultName": "Default",
      "BackupVaultArn": "arn:aws:backup:us-west-2:123456789012:backup-
vault:Default",
      "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/
i-12345678901234567",
      "CreationDate": 1600721892.929,
      "State": "CREATED",
      "PercentDone": "0.0",
      "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/
AWSBackupDefaultServiceRole",
      "StartBy": 1600725492.929,
      "ResourceType": "EC2"
    },
    {
      "BackupJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "BackupVaultName": "Default",
      "BackupVaultArn": "arn:aws:backup:us-west-2:123456789012:backup-
vault:Default",
      "RecoveryPointArn": "arn:aws:backup:us-west-2:123456789012:recovery-
point:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
      "ResourceArn": "arn:aws:elasticfilesystem:us-west-2:123456789012:file-
system/fs-12345678",
      "CreationDate": 1600721724.77,
      "CompletionDate": 1600721744.488,
      "State": "COMPLETED",
      "PercentDone": "100.0",
      "BackupSizeInBytes": 71,
      "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/
AWSBackupDefaultServiceRole",
      "StartBy": 1600725324.77,
      "ResourceType": "EFS"
    }
  ]
}
```

```
]
}
```

Para obtener más información, consulte [Creación de una copia de seguridad](#) en la Guía para desarrolladores AWS de copias de seguridad.

Ejemplo 2: Para enumerar los trabajos de copia de seguridad completados

En el siguiente `list-backup-jobs` ejemplo, se muestran los metadatos sobre los trabajos de copia de seguridad completados en su AWS cuenta.

```
aws backup list-backup-jobs \
  --by-state COMPLETED
```

Salida:

```
{
  "BackupJobs": [
    {
      "BackupJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "BackupVaultName": "Default",
      "BackupVaultArn": "arn:aws:backup:us-west-2:123456789012:backup-
vault:Default",
      "RecoveryPointArn": "arn:aws:backup:us-west-2:123456789012:recovery-
point:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
      "ResourceArn": "arn:aws:elasticfilesystem:us-west-2:123456789012:file-
system/fs-12345678",
      "CreationDate": 1600721724.77,
      "CompletionDate": 1600721744.488,
      "State": "COMPLETED",
      "PercentDone": "100.0",
      "BackupSizeInBytes": 71,
      "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/
AWSBackupDefaultServiceRole",
      "StartBy": 1600725324.77,
      "ResourceType": "EFS"
    }
  ]
}
```

Para obtener más información, consulte [Creación de una copia de seguridad](#) en la Guía para desarrolladores AWS de copias de seguridad.

- Para API obtener más información, consulte [ListBackupJobs](#) la Referencia de AWS CLI comandos.

## AWS Batch ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Batch.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **cancel-job**

En el siguiente ejemplo de código, se muestra cómo usar `cancel-job`.

AWS CLI

Para cancelar un trabajo

En este ejemplo se cancela un trabajo con el identificador de trabajo especificado.

Comando:

```
aws batch cancel-job --job-id bcf0b186-a532-4122-842e-2ccab8d54efb --  
reason "Canceling job."
```

- Para API obtener más información, consulte [CancelJob](#) la Referencia de AWS CLI comandos.

#### **create-compute-environment**

En el siguiente ejemplo de código, se muestra cómo usar `create-compute-environment`.



## AWS CLI

Para crear un entorno informático gestionado con instancias bajo demanda

En este ejemplo, se crea un entorno informático gestionado con tipos de instancias C4 específicos que se lanzan bajo demanda. El entorno informático se denomina OnDemand C4.

Comando:

```
aws batch create-compute-environment --cli-input-json file://<path_to_json_file>/C4OnDemand.json
```

JSONformato de archivo:

```
{
  "computeEnvironmentName": "C4OnDemand",
  "type": "MANAGED",
  "state": "ENABLED",
  "computeResources": {
    "type": "EC2",
    "minvCpus": 0,
    "maxvCpus": 128,
    "desiredvCpus": 48,
    "instanceTypes": [
      "c4.large",
      "c4.xlarge",
      "c4.2xlarge",
      "c4.4xlarge",
      "c4.8xlarge"
    ],
  },
  "subnets": [
    "subnet-220c0e0a",
    "subnet-1a95556d",
    "subnet-978f6dce"
  ],
  "securityGroupIds": [
    "sg-cf5093b2"
  ],
  "ec2KeyPair": "id_rsa",
  "instanceRole": "ecsInstanceRole",
  "tags": {
    "Name": "Batch Instance - C4OnDemand"
  }
}
```

```
  },  
  "serviceRole": "arn:aws:iam::012345678910:role/AWSBatchServiceRole"  
}
```

Salida:

```
{  
  "computeEnvironmentName": "C4OnDemand",  
  "computeEnvironmentArn": "arn:aws:batch:us-east-1:012345678910:compute-  
environment/C4OnDemand"  
}
```

Para crear un entorno informático gestionado con instancias puntuales

En este ejemplo, se crea un entorno informático gestionado con el tipo de instancia M4 que se lanza cuando el precio de oferta puntual es igual o inferior al 20% del precio bajo demanda del tipo de instancia. El entorno informático se denomina M4Spot.

Comando:

```
aws batch create-compute-environment --cli-input-json file://<path_to_json_file>/  
M4Spot.json
```

JSONformato de archivo:

```
{  
  "computeEnvironmentName": "M4Spot",  
  "type": "MANAGED",  
  "state": "ENABLED",  
  "computeResources": {  
    "type": "SPOT",  
    "spotIamFleetRole": "arn:aws:iam::012345678910:role/aws-ec2-spot-fleet-role",  
    "minvCpus": 0,  
    "maxvCpus": 128,  
    "desiredvCpus": 4,  
    "instanceTypes": [  
      "m4"  
    ],  
    "bidPercentage": 20,  
    "subnets": [  
      "subnet-220c0e0a",  
    ]  
  }  
}
```

```
    "subnet-1a95556d",
    "subnet-978f6dce"
  ],
  "securityGroupIds": [
    "sg-cf5093b2"
  ],
  "ec2KeyPair": "id_rsa",
  "instanceRole": "ecsInstanceRole",
  "tags": {
    "Name": "Batch Instance - M4Spot"
  }
},
"serviceRole": "arn:aws:iam::012345678910:role/AWSBatchServiceRole"
}
```

Salida:

```
{
  "computeEnvironmentName": "M4Spot",
  "computeEnvironmentArn": "arn:aws:batch:us-east-1:012345678910:compute-
environment/M4Spot"
}
```

- Para API obtener más información, consulte [CreateComputeEnvironment](#) la Referencia de AWS CLI comandos.

## create-job-queue

En el siguiente ejemplo de código, se muestra cómo usar `create-job-queue`.

### AWS CLI

Para crear una cola de trabajos de baja prioridad con un único entorno informático

En este ejemplo, se crea una cola de trabajos denominada `LowPriority` que utiliza el entorno informático `M4Spot`.

Comando:

```
aws batch create-job-queue --cli-input-json file://<path_to_json_file>/
LowPriority.json
```

## JSONformato de archivo:

```
{
  "jobQueueName": "LowPriority",
  "state": "ENABLED",
  "priority": 10,
  "computeEnvironmentOrder": [
    {
      "order": 1,
      "computeEnvironment": "M4Spot"
    }
  ]
}
```

## Salida:

```
{
  "jobQueueArn": "arn:aws:batch:us-east-1:012345678910:job-queue/LowPriority",
  "jobQueueName": "LowPriority"
}
```

Para crear una cola de trabajos de alta prioridad con dos entornos de procesamiento

En este ejemplo, se crea una cola de trabajos denominada HighPriority que utiliza el entorno OnDemand informático C4 con un orden de 1 y el entorno informático M4Spot con un orden de 2. El programador intentará colocar primero los trabajos en el entorno informático C4. OnDemand

## Comando:

```
aws batch create-job-queue --cli-input-json file://<path_to_json_file>/HighPriority.json
```

## JSONformato de archivo:

```
{
  "jobQueueName": "HighPriority",
  "state": "ENABLED",
  "priority": 1,
  "computeEnvironmentOrder": [
    {
```

```
    "order": 1,
    "computeEnvironment": "C4OnDemand"
  },
  {
    "order": 2,
    "computeEnvironment": "M4Spot"
  }
]
```

Salida:

```
{
  "jobQueueArn": "arn:aws:batch:us-east-1:012345678910:job-queue/HighPriority",
  "jobQueueName": "HighPriority"
}
```

- Para API obtener más información, consulte [CreateJobQueue](#) la Referencia de AWS CLI comandos.

## delete-compute-environment

En el siguiente ejemplo de código, se muestra cómo usar `delete-compute-environment`.

AWS CLI

Para eliminar un entorno de cómputo

En este ejemplo, se elimina el entorno de OnDemand cómputo P2.

Comando:

```
aws batch delete-compute-environment --compute-environment P2OnDemand
```

- Para API obtener más información, consulte la Referencia [DeleteComputeEnvironment](#) de AWS CLI comandos.

## delete-job-queue

En el siguiente ejemplo de código, se muestra cómo usar `delete-job-queue`.

## AWS CLI

Para eliminar una cola de trabajos

En este ejemplo se elimina la cola de GPGPU trabajos.

Comando:

```
aws batch delete-job-queue --job-queue GPGPU
```

- Para API obtener más información, consulte la Referencia [DeleteJobQueue](#) de AWS CLI comandos.

## deregister-job-definition

En el siguiente ejemplo de código, se muestra cómo usar deregister-job-definition.

### AWS CLI

Para anular el registro de una definición de trabajo

En este ejemplo se anula el registro de una definición de trabajo llamada sleep10.

Comando:

```
aws batch deregister-job-definition --job-definition sleep10
```

- Para API obtener más información, consulte la Referencia de [DeregisterJobDefinition](#) comandos AWS CLI .

## describe-compute-environments

En el siguiente ejemplo de código, se muestra cómo usar describe-compute-environments.

### AWS CLI

Descripción de entornos informáticos

Este ejemplo describe el entorno de OnDemand cómputo P2.

## Comando:

```
aws batch describe-compute-environments --compute-environments P20nDemand
```

## Salida:

```
{
  "computeEnvironments": [
    {
      "status": "VALID",
      "serviceRole": "arn:aws:iam::012345678910:role/AWSBatchServiceRole",
      "computeEnvironmentArn": "arn:aws:batch:us-east-1:012345678910:compute-
environment/P20nDemand",
      "computeResources": {
        "subnets": [
          "subnet-220c0e0a",
          "subnet-1a95556d",
          "subnet-978f6dce"
        ],
        "tags": {
          "Name": "Batch Instance - P20nDemand"
        },
        "desiredvCpus": 48,
        "minvCpus": 0,
        "instanceTypes": [
          "p2"
        ],
        "securityGroupIds": [
          "sg-cf5093b2"
        ],
        "instanceRole": "ecsInstanceRole",
        "maxvCpus": 128,
        "type": "EC2",
        "ec2KeyPair": "id_rsa"
      },
      "statusReason": "ComputeEnvironment Healthy",
      "ecsClusterArn": "arn:aws:ecs:us-east-1:012345678910:cluster/
P20nDemand_Batch_2c06f29d-d1fe-3a49-879d-42394c86effc",
      "state": "ENABLED",
      "computeEnvironmentName": "P20nDemand",
      "type": "MANAGED"
    }
  ]
}
```

```
}
```

- Para API obtener más información, consulte [DescribeComputeEnvironments](#) la Referencia de AWS CLI comandos.

## describe-job-definitions

En el siguiente ejemplo de código, se muestra cómo usar `describe-job-definitions`.

### AWS CLI

Para describir las definiciones de tareas activas

En este ejemplo se describen todas las definiciones de puestos activos.

Comando:

```
aws batch describe-job-definitions --status ACTIVE
```

Salida:

```
{
  "jobDefinitions": [
    {
      "status": "ACTIVE",
      "jobDefinitionArn": "arn:aws:batch:us-east-1:012345678910:job-
definition/sleep60:1",
      "containerProperties": {
        "mountPoints": [],
        "parameters": {},
        "image": "busybox",
        "environment": {},
        "vcpus": 1,
        "command": [
          "sleep",
          "60"
        ],
        "volumes": [],
        "memory": 128,
        "ulimits": []
      },
    },
  ],
}
```



```
        "type": "container",
        "jobDefinitionName": "sleep60",
        "revision": 1
      }
    ]
  }
}
```

- Para API obtener más información, consulte [DescribeJobDefinitions](#) la Referencia de AWS CLI comandos.

## describe-job-queues

En el siguiente ejemplo de código, se muestra cómo usar `describe-job-queues`.

### AWS CLI

Para describir una cola de trabajos

En este ejemplo se describe la cola de HighPriority trabajos.

Comando:

```
aws batch describe-job-queues --job-queues HighPriority
```

Salida:

```
{
  "jobQueues": [
    {
      "status": "VALID",
      "jobQueueArn": "arn:aws:batch:us-east-1:012345678910:job-queue/HighPriority",
      "computeEnvironmentOrder": [
        {
          "computeEnvironment": "arn:aws:batch:us-east-1:012345678910:compute-environment/C4OnDemand",
          "order": 1
        }
      ],
      "statusReason": "JobQueue Healthy",
      "priority": 1,
    }
  ]
}
```

```
        "state": "ENABLED",
        "jobQueueName": "HighPriority"
    }
]
}
```

- Para API obtener más información, consulte [DescribeJobQueues](#) la Referencia de AWS CLI comandos.

## describe-jobs

En el siguiente ejemplo de código, se muestra cómo usar `describe-jobs`.

### AWS CLI

Para describir un trabajo

El siguiente `describe-jobs` ejemplo describe un trabajo con el identificador de trabajo especificado.

```
aws batch describe-jobs \
  --jobs bcf0b186-a532-4122-842e-2ccab8d54efb
```

Salida:

```
{
  "jobs": [
    {
      "status": "SUBMITTED",
      "container": {
        "mountPoints": [],
        "image": "busybox",
        "environment": [],
        "vcpus": 1,
        "command": [
          "sleep",
          "60"
        ],
        "volumes": [],
        "memory": 128,
        "ulimits": []
      }
    }
  ]
}
```

```
    },
    "parameters": {},
    "jobDefinition": "arn:aws:batch:us-east-1:012345678910:job-definition/
sleep60:1",
    "jobQueue": "arn:aws:batch:us-east-1:012345678910:job-queue/
HighPriority",
    "jobId": "bcf0b186-a532-4122-842e-2ccab8d54efb",
    "dependsOn": [],
    "jobName": "example",
    "createdAt": 1480483387803
  }
]
}
```

- Para API obtener más información, consulte [DescribeJobs](#) la Referencia de AWS CLI comandos.

## list-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-jobs`.

### AWS CLI

Para ver una lista de los trabajos en ejecución

En este ejemplo, se enumeran los trabajos en ejecución de la cola de HighPriority trabajos.

Comando:

```
aws batch list-jobs --job-queue HighPriority
```

Salida:

```
{
  "jobSummaryList": [
    {
      "jobName": "example",
      "jobId": "e66ff5fd-a1ff-4640-b1a2-0b0a142f49bb"
    }
  ]
}
```

Para enumerar los trabajos enviados

En este ejemplo, se enumeran los HighPriority trabajos de la cola de trabajos que se encuentran en el estado de SUBMITTED trabajo.

Comando:

```
aws batch list-jobs --job-queue HighPriority --job-status SUBMITTED
```

Salida:

```
{
  "jobSummaryList": [
    {
      "jobName": "example",
      "jobId": "68f0c163-fbd4-44e6-9fd1-25b14a434786"
    }
  ]
}
```

- Para API obtener más información, consulte [ListJobs](#) la Referencia de AWS CLI comandos.

## register-job-definition

En el siguiente ejemplo de código, se muestra cómo usar register-job-definition.

AWS CLI

Para registrar una definición de trabajo

En este ejemplo, se registra una definición de trabajo para un trabajo de contenedor simple.

Comando:

```
aws batch register-job-definition --job-definition-name sleep30 --type container --
container-properties '{ "image": "busybox", "vcpus": 1, "memory": 128, "command":
[ "sleep", "30"]}'
```

Salida:

```
{
```

```
"jobDefinitionArn": "arn:aws:batch:us-east-1:012345678910:job-definition/sleep30:1",
  "jobDefinitionName": "sleep30",
  "revision": 1
}
```

- Para API obtener más información, consulte [RegisterJobDefinition](#) la Referencia de AWS CLI comandos.

## submit-job

En el siguiente ejemplo de código, se muestra cómo usar `submit-job`.

### AWS CLI

Para enviar un trabajo

En este ejemplo, se envía un simple trabajo contenedor denominado `example` a la cola de `HighPriority` trabajos.

Comando:

```
aws batch submit-job --job-name example --job-queue HighPriority --job-definition sleep60
```

Salida:

```
{
  "jobName": "example",
  "jobId": "876da822-4198-45f2-a252-6cea32512ea8"
}
```

- Para API obtener más información, consulte la Referencia [SubmitJob](#) de AWS CLI comandos.

## terminate-job

En el siguiente ejemplo de código, se muestra cómo usar `terminate-job`.

### AWS CLI

Para terminar un trabajo

En este ejemplo se termina un trabajo con el identificador de trabajo especificado.

Comando:

```
aws batch terminate-job --job-id 61e743ed-35e4-48da-b2de-5c8333821c84 --  
reason "Terminating job."
```

- Para API obtener más información, consulte [TerminateJob](#) la Referencia de AWS CLI comandos.

## update-compute-environment

En el siguiente ejemplo de código, se muestra cómo usar `update-compute-environment`.

AWS CLI

Para actualizar un entorno informático

En este ejemplo, se deshabilita el entorno de OnDemand cómputo P2 para que se pueda eliminar.

Comando:

```
aws batch update-compute-environment --compute-environment P2OnDemand --  
state DISABLED
```

Salida:

```
{  
  "computeEnvironmentName": "P2OnDemand",  
  "computeEnvironmentArn": "arn:aws:batch:us-east-1:012345678910:compute-  
environment/P2OnDemand"  
}
```

- Para API obtener más información, consulte la Referencia [UpdateComputeEnvironment](#) de AWS CLI comandos.

## update-job-queue

En el siguiente ejemplo de código, se muestra cómo usar `update-job-queue`.

## AWS CLI

Para actualizar una cola de trabajos

En este ejemplo, se deshabilita una cola de trabajos para poder eliminarla.

Comando:

```
aws batch update-job-queue --job-queue GPGPU --state DISABLED
```

Salida:

```
{
  "jobQueueArn": "arn:aws:batch:us-east-1:012345678910:job-queue/GPGPU",
  "jobQueueName": "GPGPU"
}
```

- Para API obtener más información, consulte la Referencia [UpdateJobQueue](#) de AWS CLI comandos.

## AWS Budgets ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Budgets.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-budget**

En el siguiente ejemplo de código, se muestra cómo usar `create-budget`.

## AWS CLI

Para crear un presupuesto de costos y uso

El siguiente `create-budget` comando crea un presupuesto de costo y uso.

```
aws budgets create-budget \  
  --account-id 111122223333 \  
  --budget file://budget.json \  
  --notifications-with-subscribers file://notifications-with-subscribers.json
```

Contenidos de `budget.json`:

```
{  
  "BudgetLimit": {  
    "Amount": "100",  
    "Unit": "USD"  
  },  
  "BudgetName": "Example Tag Budget",  
  "BudgetType": "COST",  
  "CostFilters": {  
    "TagKeyValue": [  
      "user:Key$value1",  
      "user:Key$value2"  
    ]  
  },  
  "CostTypes": {  
    "IncludeCredit": true,  
    "IncludeDiscount": true,  
    "IncludeOtherSubscription": true,  
    "IncludeRecurring": true,  
    "IncludeRefund": true,  
    "IncludeSubscription": true,  
    "IncludeSupport": true,  
    "IncludeTax": true,  
    "IncludeUpfront": true,  
    "UseBlended": false  
  },  
  "TimePeriod": {  
    "Start": 1477958399,  
    "End": 3706473600  
  },  
  "TimeUnit": "MONTHLY"
```



```
}
```

Contenidos de `notifications-with-subscribers.json`:

```
[
  {
    "Notification": {
      "ComparisonOperator": "GREATER_THAN",
      "NotificationType": "ACTUAL",
      "Threshold": 80,
      "ThresholdType": "PERCENTAGE"
    },
    "Subscribers": [
      {
        "Address": "example@example.com",
        "SubscriptionType": "EMAIL"
      }
    ]
  }
]
```

- Para API obtener más información, consulte [CreateBudget](#) la Referencia de AWS CLI comandos.

## **create-notification**

En el siguiente ejemplo de código, se muestra cómo usar `create-notification`.

### AWS CLI

Para crear una notificación para el presupuesto de costes y uso especificado

En este ejemplo, se crea una notificación para el presupuesto de coste y uso especificado.

Comando:

```
aws budgets create-notification --account-id 111122223333 --budget-name "Example Budget" --notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENTAGE --subscriber SubscriptionType=EMAIL,Address=example@example.com
```

- Para API obtener más información, consulte [CreateNotification](#) la Referencia de AWS CLI comandos.

## create-subscriber

En el siguiente ejemplo de código, se muestra cómo usar `create-subscriber`.

### AWS CLI

Para crear un suscriptor para una notificación asociada a un presupuesto de costo y uso

En este ejemplo, se crea un suscriptor para la notificación especificada.

Comando:

```
aws budgets create-subscriber --account-id 111122223333 --budget-name "Example Budget" --notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENTAGE --subscriber SubscriptionType=EMAIL,Address=example@example.com
```

- Para API obtener más información, consulte [CreateSubscriber](#) la Referencia de AWS CLI comandos.

## delete-budget

En el siguiente ejemplo de código, se muestra cómo usar `delete-budget`.

### AWS CLI

Para eliminar un presupuesto de costo y uso

En este ejemplo, se elimina el presupuesto de coste y uso especificado.

Comando:

```
aws budgets delete-budget --account-id 111122223333 --budget-name "Example Budget"
```

- Para API obtener más información, consulte [DeleteBudget](#) la Referencia de AWS CLI comandos.

## delete-notification

En el siguiente ejemplo de código, se muestra cómo usar `delete-notification`.

### AWS CLI

Para eliminar una notificación de un presupuesto

En este ejemplo, se elimina la notificación especificada del presupuesto especificado.

Comando:

```
aws budgets delete-notification --account-id 111122223333 --budget-name "Example Budget" --notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENT
```

- Para API obtener más información, consulte [DeleteNotification](#) la Referencia de AWS CLI comandos.

## delete-subscriber

En el siguiente ejemplo de código, se muestra cómo usar `delete-subscriber`.

### AWS CLI

Para eliminar un suscriptor de una notificación

En este ejemplo, se elimina el suscriptor especificado de la notificación especificada.

Comando:

```
aws budgets delete-subscriber --account-id 111122223333 --budget-name "Example Budget" --notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENT --subscriber SubscriptionType=EMAIL,Address=example@example.com
```

- Para API obtener más información, consulte [DeleteSubscriber](#) la Referencia de AWS CLI comandos.

## describe-budget

En el siguiente ejemplo de código, se muestra cómo usar `describe-budget`.

## AWS CLI

Para recuperar un presupuesto asociado a una cuenta

En este ejemplo, se recupera el presupuesto de coste y uso especificado.

Comando:

```
aws budgets describe-budget --account-id 111122223333 --budget-name "Example Budget"
```

Salida:

```
{
  "Budget": {
    "CalculatedSpend": {
      "ForecastedSpend": {
        "Amount": "2641.548000000000022919266484677791595458984375",
        "Unit": "USD"
      },
      "ActualSpend": {
        "Amount": "604.45600000000000172803993336856365203857421875",
        "Unit": "USD"
      }
    },
    "BudgetType": "COST",
    "BudgetLimit": {
      "Amount": "100",
      "Unit": "USD"
    },
    "BudgetName": "Example Budget",
    "CostTypes": {
      "IncludeOtherSubscription": true,
      "IncludeUpfront": true,
      "IncludeRefund": true,
      "UseBlended": false,
      "IncludeDiscount": true,
      "UseAmortized": false,
      "IncludeTax": true,
      "IncludeCredit": true,
      "IncludeSupport": true,
      "IncludeRecurring": true,
      "IncludeSubscription": true
    }
  },
}
```

```

    "TimeUnit": "MONTHLY",
    "TimePeriod": {
      "Start": 1477958399.0,
      "End": 3706473600.0
    },
    "CostFilters": {
      "AZ": [
        "us-east-1"
      ]
    }
  }
}

```

- Para API obtener más información, consulte [DescribeBudget](#) la Referencia de AWS CLI comandos.

## describe-budgets

En el siguiente ejemplo de código, se muestra cómo usar `describe-budgets`.

### AWS CLI

Para recuperar los presupuestos asociados a una cuenta

En este ejemplo, se recuperan los presupuestos de coste y uso de una cuenta.

Comando:

```
aws budgets describe-budgets --account-id 111122223333 --max-results 20
```

Salida:

```

{
  "Budgets": [
    {
      "CalculatedSpend": {
        "ForecastedSpend": {
          "Amount": "2641.548000000000022919266484677791595458984375",
          "Unit": "USD"
        },
        "ActualSpend": {
          "Amount": "604.45600000000000172803993336856365203857421875",

```

```

        "Unit": "USD"
    }
},
"BudgetType": "COST",
"BudgetLimit": {
    "Amount": "100",
    "Unit": "USD"
},
"BudgetName": "Example Budget",
"CostTypes": {
    "IncludeOtherSubscription": true,
    "IncludeUpfront": true,
    "IncludeRefund": true,
    "UseBlended": false,
    "IncludeDiscount": true,
    "UseAmortized": false,
    "IncludeTax": true,
    "IncludeCredit": true,
    "IncludeSupport": true,
    "IncludeRecurring": true,
    "IncludeSubscription": true
},
"TimeUnit": "MONTHLY",
"TimePeriod": {
    "Start": 1477958399.0,
    "End": 3706473600.0
},
"CostFilters": {
    "AZ": [
        "us-east-1"
    ]
}
}
]
}

```

- Para API obtener más información, consulte [DescribeBudgets](#) la Referencia de AWS CLI comandos.

### **describe-notifications-for-budget**

En el siguiente ejemplo de código, se muestra cómo usar `describe-notifications-for-budget`.

## AWS CLI

Para recuperar las notificaciones de un presupuesto

En este ejemplo, se recuperan las notificaciones de un presupuesto de costo y uso.

Comando:

```
aws budgets describe-notifications-for-budget --account-id 111122223333 --budget-name "Example Budget" --max-results 5
```

Salida:

```
{
  "Notifications": [
    {
      "Threshold": 80.0,
      "ComparisonOperator": "GREATER_THAN",
      "NotificationType": "ACTUAL"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeNotificationsForBudget](#) la Referencia de AWS CLI comandos.

## describe-subscribers-for-notification

En el siguiente ejemplo de código, se muestra cómo usar `describe-subscribers-for-notification`.

## AWS CLI

Para recuperar los suscriptores de una notificación de presupuesto

En este ejemplo, se recuperan los suscriptores de una notificación de presupuesto de costo y uso.

Comando:

```
aws budgets describe-subscribers-for-notification --account-id 111122223333 --budget-name "Example Budget" --
```

```
notification NotificationType=ACTUAL, ComparisonOperator=GREATER_THAN, Threshold=80, ThresholdType=PERCENTAGE
--max-results 5
```

Salida:

```
{
  "Subscribers": [
    {
      "SubscriptionType": "EMAIL",
      "Address": "example2@example.com"
    },
    {
      "SubscriptionType": "EMAIL",
      "Address": "example@example.com"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeSubscribersForNotification](#) la Referencia de AWS CLI comandos.

## update-budget

En el siguiente ejemplo de código, se muestra cómo usar `update-budget`.

### AWS CLI

Para reemplazar un presupuesto por un presupuesto de costo y uso

En este ejemplo, se reemplaza un presupuesto de costo y uso por un presupuesto nuevo.

Comando:

```
aws budgets update-budget --account-id 111122223333 --new-budget file://new-budget.json
```

`new-budget.json`:

```
{
  "BudgetLimit": {
```



```
    "Amount": "100",
    "Unit": "USD"
  },
  "BudgetName": "Example Budget",
  "BudgetType": "COST",
  "CostFilters": {
    "AZ" : [ "us-east-1" ]
  },
  "CostTypes": {
    "IncludeCredit": false,
    "IncludeDiscount": true,
    "IncludeOtherSubscription": true,
    "IncludeRecurring": true,
    "IncludeRefund": true,
    "IncludeSubscription": true,
    "IncludeSupport": true,
    "IncludeTax": true,
    "IncludeUpfront": true,
    "UseBlended": false,
    "UseAmortized": true
  },
  "TimePeriod": {
    "Start": 1477958399,
    "End": 3706473600
  },
  "TimeUnit": "MONTHLY"
}
```

- Para obtener API más información, consulte la Referencia de comandos [UpdateBudget](#).AWS CLI

## update-notification

En el siguiente ejemplo de código, se muestra cómo usar update-notification.

### AWS CLI

Para reemplazar una notificación por un presupuesto de costo y uso

En este ejemplo, se sustituye una notificación del 80% de un presupuesto de coste y uso por una notificación del 90%.

Comando:

```
aws budgets update-notification --account-id 111122223333 --budget-name "Example Budget" --old-notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENT --new-notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=90,ThresholdType=PERCENT
```

- Para API obtener más información, consulte [UpdateNotification](#) la Referencia de AWS CLI comandos.

## update-subscriber

En el siguiente ejemplo de código, se muestra cómo usar `update-subscriber`.

### AWS CLI

Para reemplazar un suscriptor por un presupuesto de costo y uso

En este ejemplo, se reemplaza al suscriptor por un presupuesto de costo y uso.

Comando:

```
aws budgets update-subscriber --account-id 111122223333 --budget-name "Example Budget" --notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENT --old-subscriber SubscriptionType=EMAIL,Address=example@example.com --new-subscriber SubscriptionType=EMAIL,Address=example2@example.com
```

- Para API obtener más información, consulte [UpdateSubscriber](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon Chime con AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Amazon Chime.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **associate-phone-number-with-user**

En el siguiente ejemplo de código, se muestra cómo usar `associate-phone-number-with-user`.

#### AWS CLI

Para asociar un número de teléfono a un usuario

El siguiente `associate-phone-number-with-user` ejemplo asocia el número de teléfono especificado a un usuario.

```
aws chime associate-phone-number-with-user \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --user-id 1ab2345c-67de-8901-f23g-45h678901j2k \  
  --e164-phone-number "+12065550100"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar los números de teléfono de los usuarios](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [AssociatePhoneNumberWithUser](#) la Referencia de AWS CLI comandos.

### **associate-phone-numbers-with-voice-connector-group**

En el siguiente ejemplo de código, se muestra cómo usar `associate-phone-numbers-with-voice-connector-group`.

#### AWS CLI

Para asociar números de teléfono a un grupo de conectores de voz de Amazon Chime

En el siguiente `associate-phone-numbers-with-voice-connector-group` ejemplo, se asocian los números de teléfono especificados a un grupo de conectores de voz de Amazon Chime.

```
aws chime associate-phone-numbers-with-voice-connector-group \  
  --voice-connector-group-id 123a456b-c7d8-90e1-fg23-4h567jk18901 \  
  --e164-phone-numbers "+12065550100" "+12065550101" \  
  --force-associate
```

Salida:

```
{  
  "PhoneNumberErrors": []  
}
```

Para obtener más información, consulte [Uso de grupos de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [AssociatePhoneNumbersWithVoiceConnectorGroup](#) de AWS CLI comandos.

## `associate-phone-numbers-with-voice-connector`

En el siguiente ejemplo de código, se muestra cómo usar `associate-phone-numbers-with-voice-connector`.

AWS CLI

Para asociar números de teléfono a un conector de voz de Amazon Chime

En el siguiente `associate-phone-numbers-with-voice-connector` ejemplo, se asocian los números de teléfono especificados a un conector de voz de Amazon Chime.

```
aws chime associate-phone-numbers-with-voice-connector \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --e164-phone-numbers "+12065550100" "+12065550101" \  
  --force-associate
```

Salida:

```
{
```

```
"PhoneNumberErrors": []  
}
```

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [AssociatePhoneNumbersWithVoiceConnector](#) de AWS CLI comandos.

## **associate-signin-delegate-groups-with-account**

En el siguiente ejemplo de código, se muestra cómo usar `associate-signin-delegate-groups-with-account`.

### AWS CLI

Para asociar grupos de delegados que inicien sesión

El siguiente `associate-signin-delegate-groups-with-account` ejemplo asocia el grupo de delegados de inicio de sesión especificado con la cuenta de Amazon Chime especificada.

```
aws chime associate-signin-delegate-groups-with-account \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --signin-delegate-groups GroupName=my_users
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar el acceso y los permisos de los usuarios](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [AssociateSigninDelegateGroupsWithAccount](#) la Referencia de AWS CLI comandos.

## **batch-create-room-membership**

En el siguiente ejemplo de código, se muestra cómo usar `batch-create-room-membership`.

### AWS CLI

Para crear membresías a varias salas

El siguiente `batch-create-room-membership` ejemplo agrega varios usuarios a una sala de chat como miembros de la sala de chat. También asigna funciones de administrador y miembro a los usuarios.

```
aws chime batch-create-room-membership \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \  
  --membership-item-list "MemberId=1ab2345c-67de-8901-  
f23g-45h678901j2k,Role=Administrator" "MemberId=2ab2345c-67de-8901-  
f23g-45h678901j2k,Role=Member"
```

Salida:

```
{  
  "ResponseMetadata": {  
    "RequestId": "169ba401-d886-475f-8b3f-e01eac6fadfb",  
    "HTTPStatusCode": 201,  
    "HTTPHeaders": {  
      "x-amzn-requestid": "169ba401-d886-475f-8b3f-e01eac6fadfb",  
      "content-type": "application/json",  
      "content-length": "13",  
      "date": "Mon, 02 Dec 2019 22:46:58 GMT",  
      "connection": "keep-alive"  
    },  
    "RetryAttempts": 0  
  },  
  "Errors": []  
}
```

Para obtener más información, consulte [Creación de una sala de chat](#) en la Guía del usuario de Amazon Chime.

- Para API obtener más información, consulte [BatchCreateRoomMembership](#) la Referencia de AWS CLI comandos.

## **batch-delete-phone-number**

En el siguiente ejemplo de código, se muestra cómo usar `batch-delete-phone-number`.

### AWS CLI

Para eliminar varios números de teléfono

En el siguiente `batch-delete-phone-number` ejemplo, se eliminan todos los números de teléfono especificados.

```
aws chime batch-delete-phone-number \  
  --phone-number-ids "%2B12065550100" "%2B12065550101"
```

Este comando no genera ninguna salida. Salida:

```
{  
  "PhoneNumberErrors": []  
}
```

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [BatchDeletePhoneNumber](#) la Referencia de AWS CLI comandos.

## `batch-suspend-user`

En el siguiente ejemplo de código, se muestra cómo usar `batch-suspend-user`.

### AWS CLI

Para suspender a varios usuarios

El siguiente `batch-suspend-user` ejemplo suspende a los usuarios de la lista de la cuenta de Amazon Chime especificada.

```
aws chime batch-suspend-user \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --user-id-list "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE" "a1b2c3d4-5678-90ab-  
cdef-33333EXAMPLE" "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE"
```

Salida:

```
{  
  "UserErrors": []  
}
```

- Para API obtener más información, consulte la Referencia [BatchSuspendUser](#) de AWS CLI comandos.

## batch-unsuspend-user

En el siguiente ejemplo de código, se muestra cómo usar `batch-unsuspend-user`.

### AWS CLI

Para anular la suspensión de varios usuarios

En el siguiente `batch-unsuspend-user` ejemplo, se elimina cualquier suspensión anterior de los usuarios incluidos en la lista de la cuenta de Amazon Chime especificada.

```
aws chime batch-unsuspend-user \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --user-id-list "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE" "a1b2c3d4-5678-90ab-  
cdef-33333EXAMPLE" "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE"
```

Salida:

```
{  
  "UserErrors": []  
}
```

- Para API obtener más información, consulte [BatchUnsuspendUser](#) la Referencia de AWS CLI comandos.

## batch-update-phone-number

En el siguiente ejemplo de código, se muestra cómo usar `batch-update-phone-number`.

### AWS CLI

Para actualizar varios tipos de productos de números de teléfono al mismo tiempo

En el siguiente `batch-update-phone-number` ejemplo, se actualizan los tipos de producto de todos los números de teléfono especificados.

```
aws chime batch-update-phone-number \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --phone-number-list "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE" "a1b2c3d4-5678-90ab-  
cdef-33333EXAMPLE" "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE"
```



```
--update-phone-number-request-items PhoneNumberId=  
%2B12065550100,ProductType=BusinessCalling PhoneNumberId=  
%2B12065550101,ProductType=BusinessCalling
```

Salida:

```
{  
  "PhoneNumberErrors": []  
}
```

Para actualizar varios nombres de llamadas a números de teléfono al mismo tiempo

En el siguiente `batch-update-phone-number` ejemplo, se actualizan los nombres de llamada de todos los números de teléfono especificados.

```
aws chime batch-update-phone-number \  
  --update-phone-number-request-items PhoneNumberId=  
%2B14013143874,CallingName=phonenumber1 PhoneNumberId=  
%2B14013144061,CallingName=phonenumber2
```

Salida:

```
{  
  "PhoneNumberErrors": []  
}
```

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [BatchUpdatePhoneNumber](#) la Referencia de AWS CLI comandos.

## batch-update-user

En el siguiente ejemplo de código, se muestra cómo usar `batch-update-user`.

### AWS CLI

Para actualizar varios usuarios en un solo comando

En el siguiente batch-update-user ejemplo, se actualiza la cuenta de Amazon Chime especificada LicenseType para cada uno de los usuarios de la lista.

```
aws chime batch-update-user \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --update-user-request-items "UserId=a1b2c3d4-5678-90ab-  
cdef-22222EXAMPLE,LicenseType=Basic" "UserId=a1b2c3d4-5678-90ab-  
cdef-33333EXAMPLE,LicenseType=Basic"
```

Salida:

```
{  
  "UserErrors": []  
}
```

- Para API obtener más información, consulte [BatchUpdateUser](#) la Referencia de AWS CLI comandos.

## create-account

En el siguiente ejemplo de código, se muestra cómo usar create-account.

### AWS CLI

Para crear una cuenta

En el siguiente create-account ejemplo, se crea una cuenta de Amazon Chime en la cuenta del AWS administrador.

```
aws chime create-account \  
  --name MyChimeAccount
```

Salida:

```
{  
  "Account": {  
    "AwsAccountId": "111122223333",  
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "Name": "MyChimeAccount",
```

```

    "AccountType": "Team",
    "CreatedTimestamp": "2019-01-04T17:11:22.003Z",
    "DefaultLicense": "Pro",
    "SupportedLicenses": [
      "Basic",
      "Pro"
    ],
    "SigninDelegateGroups": [
      {
        "GroupName": "myGroup"
      },
    ]
  }
}

```

Para obtener más información, consulte [Introducción](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [CreateAccount](#) la Referencia de AWS CLI comandos.

## create-bot

En el siguiente ejemplo de código, se muestra cómo usar `create-bot`.

### AWS CLI

Para crear un bot de Amazon Chime

En el siguiente `create-bot` ejemplo, se crea un bot para la cuenta empresarial de Amazon Chime especificada.

```

aws chime create-bot \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --display-name "myBot" \
  --domain "example.com"

```

Salida:

```

{
  "Bot": {

```

```

    "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",
    "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",
    "DisplayName": "myBot (Bot)",
    "BotType": "ChatBot",
    "Disabled": false,
    "CreatedTimestamp": "2019-09-09T18:05:56.749Z",
    "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",
    "BotEmail": "myBot-chimebot@example.com",
    "SecurityToken": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY"
  }
}

```

Para obtener más información, consulte [Integrar un bot de chat con Amazon Chime](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte la Referencia [CreateBot](#) de AWS CLI comandos.

## create-phone-number-order

En el siguiente ejemplo de código, se muestra cómo usar `create-phone-number-order`.

### AWS CLI

Para crear un pedido de números de teléfono

En el siguiente `create-phone-number-order` ejemplo, se crea un orden de números de teléfono para los números de teléfono especificados.

```

aws chime create-phone-number-order \
  --product-type VoiceConnector \
  --e164-phone-numbers " +12065550100" "+12065550101" "+12065550102"

```

Salida:

```

{
  "PhoneNumberOrder": {
    "PhoneNumberOrderId": "abc12345-de67-89f0-123g-h45i678j9012",
    "ProductType": "VoiceConnector",
    "Status": "Processing",
    "OrderedPhoneNumbers": [
      {
        "E164PhoneNumber": "+12065550100",

```

```

        "Status": "Processing"
      },
      {
        "E164PhoneNumber": "+12065550101",
        "Status": "Processing"
      },
      {
        "E164PhoneNumber": "+12065550102",
        "Status": "Processing"
      }
    ],
    "CreatedTimestamp": "2019-08-09T21:35:21.427Z",
    "UpdatedTimestamp": "2019-08-09T21:35:22.408Z"
  }
}

```

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [CreatePhoneNumberOrder](#) la Referencia de AWS CLI comandos.

## create-proxy-session

En el siguiente ejemplo de código, se muestra cómo usar `create-proxy-session`.

### AWS CLI

Para crear una sesión de proxy

En el siguiente `create-proxy-session` ejemplo, se crea una sesión de proxy con voz y SMS funciones.

```

aws chime create-proxy-session \
  --voice-connector-id abcdef1ghij2klmno3pqr4 \
  --participant-phone-numbers "+14015550101" "+12065550100" \
  --capabilities "Voice" "SMS"

```

Salida:

```
{
```

```

"ProxySession": {
  "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",
  "ProxySessionId": "123a4bc5-67d8-901e-2f3g-h4ghjk567891",
  "Status": "Open",
  "ExpiryMinutes": 60,
  "Capabilities": [
    "SMS",
    "Voice"
  ],
  "CreatedTimestamp": "2020-04-15T16:10:10.288Z",
  "UpdatedTimestamp": "2020-04-15T16:10:10.288Z",
  "Participants": [
    {
      "PhoneNumber": "+12065550100",
      "ProxyPhoneNumber": "+19135550199"
    },
    {
      "PhoneNumber": "+14015550101",
      "ProxyPhoneNumber": "+19135550199"
    }
  ]
}
}

```

Para obtener más información, consulte [Proxy Phone Sessions](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte [CreateProxySession](#) la Referencia de AWS CLI comandos.

## **create-room-membership**

En el siguiente ejemplo de código, se muestra cómo usar `create-room-membership`.

### AWS CLI

Para crear una membresía de sala

El siguiente `create-room-membership` ejemplo agrega el usuario especificado a la sala de chat como miembro de la sala de chat.

```
aws chime create-room-membership \
```

```
--account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
--room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \  
--member-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Salida:

```
{  
  "RoomMembership": {  
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",  
    "Member": {  
      "MemberId": "1ab2345c-67de-8901-f23g-45h678901j2k",  
      "MemberType": "User",  
      "Email": "janed@example.com",  
      "FullName": "Jane Doe",  
      "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45"  
    },  
    "Role": "Member",  
    "InvitedBy": "arn:aws:iam::111122223333:user/alejandro",  
    "UpdatedTimestamp": "2019-12-02T22:36:41.969Z"  
  }  
}
```

Para obtener más información, consulte [Creación de una sala de chat](#) en la Guía del usuario de Amazon Chime.

- Para API obtener más información, consulte [CreateRoomMembership](#) la Referencia de AWS CLI comandos.

## create-room

En el siguiente ejemplo de código, se muestra cómo usar `create-room`.

### AWS CLI

Para crear una sala de chat

En el siguiente `create-room` ejemplo, se crea una sala de chat para la cuenta de Amazon Chime especificada.

```
aws chime create-room \  
--account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  

```

```
--name chatRoom
```

Salida:

```
{
  "Room": {
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
    "Name": "chatRoom",
    "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",
    "CreatedBy": "arn:aws:iam::111122223333:user/alejandro",
    "CreatedTimestamp": "2019-12-02T22:29:31.549Z",
    "UpdatedTimestamp": "2019-12-02T22:29:31.549Z"
  }
}
```

Para obtener más información, consulte [Creación de una sala de chat](#) en la Guía del usuario de Amazon Chime.

- Para API obtener más información, consulte [CreateRoom](#) la Referencia de AWS CLI comandos.

## create-user

En el siguiente ejemplo de código, se muestra cómo usar `create-user`.

### AWS CLI

Para crear un perfil de usuario para un dispositivo compartido

En el siguiente `create-user` ejemplo, se crea un perfil de dispositivo compartido para la dirección de correo electrónico especificada.

```
aws chime create-user \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --email roomdevice@example.com \  
  --user-type SharedDevice
```

Salida:

```
{
  "User": {
    "UserId": "1ab2345c-67de-8901-f23g-45h678901j2k",
  }
}
```



```

    "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",
    "PrimaryEmail": "roomdevice@example.com",
    "DisplayName": "Room Device",
    "LicenseType": "Pro",
    "UserType": "SharedDevice",
    "UserRegistrationStatus": "Registered",
    "RegisteredOn": "2020-01-15T22:38:09.806Z",
    "AlexaForBusinessMetadata": {
      "IsAlexaForBusinessEnabled": false
    }
  }
}

```

Para obtener más información, consulte [Preparación de la configuración](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [CreateUser](#) la Referencia de AWS CLI comandos.

## create-voice-connector-group

En el siguiente ejemplo de código, se muestra cómo usar `create-voice-connector-group`.

### AWS CLI

Para crear un grupo de conectores de voz de Amazon Chime

En el siguiente `create-voice-connector-group` ejemplo, se crea un grupo de conectores de voz de Amazon Chime que incluye el conector de voz de Amazon Chime especificado.

```

aws chime create-voice-connector-group \
  --name myGroup \
  --voice-connector-items VoiceConnectorId=abcdef1ghij2klmno3pqr4,Priority=2

```

Salida:

```

{
  "VoiceConnectorGroup": {
    "VoiceConnectorGroupId": "123a456b-c7d8-90e1-fg23-4h567jkl18901",
    "Name": "myGroup",
    "VoiceConnectorItems": [],
    "CreatedTimestamp": "2019-09-18T16:38:34.734Z",
    "UpdatedTimestamp": "2019-09-18T16:38:34.734Z"
  }
}

```

```
}  
}
```

Para obtener más información, consulte [Uso de grupos de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [CreateVoiceConnectorGroup](#) de AWS CLI comandos.

## **create-voice-connector**

En el siguiente ejemplo de código, se muestra cómo usar `create-voice-connector`.

### AWS CLI

Para crear un conector de voz de Amazon Chime

En el siguiente `create-voice-connector` ejemplo, se crea un conector de voz de Amazon Chime en la AWS región especificada, con el cifrado activado.

```
aws chime create-voice-connector \  
  --name newVoiceConnector \  
  --aws-region us-west-2 \  
  --require-encryption
```

Salida:

```
{  
  "VoiceConnector": {  
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",  
    "AwsRegion": "us-west-2",  
    "Name": "newVoiceConnector",  
    "OutboundHostName": "abcdef1ghij2klmno3pqr4.voiceconnector.chime.aws",  
    "RequireEncryption": true,  
    "CreatedTimestamp": "2019-09-18T20:34:01.352Z",  
    "UpdatedTimestamp": "2019-09-18T20:34:01.352Z"  
  }  
}
```

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [CreateVoiceConnector](#) de AWS CLI comandos.

## **delete-account**

En el siguiente ejemplo de código, se muestra cómo usar `delete-account`.

### AWS CLI

Para eliminar una cuenta

En el siguiente `delete-account` ejemplo, se elimina la cuenta especificada.

```
aws chime delete-account --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar su cuenta](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [DeleteAccount](#) la Referencia de AWS CLI comandos.

## **delete-phone-number**

En el siguiente ejemplo de código, se muestra cómo usar `delete-phone-number`.

### AWS CLI

Para eliminar un número de teléfono

En el siguiente `delete-phone-number` ejemplo, se mueve el número de teléfono especificado a la cola de eliminación.

```
aws chime delete-phone-number \  
  --phone-number-id " +12065550100"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [DeletePhoneNumber](#) la Referencia de AWS CLI comandos.

## **delete-proxy-session**

En el siguiente ejemplo de código, se muestra cómo usar `delete-proxy-session`.

AWS CLI

Para eliminar una sesión de proxy

En el siguiente `delete-proxy-session` ejemplo, se elimina la sesión de proxy especificada.

```
aws chime delete-proxy-session \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --proxy-session-id 123a4bc5-67d8-901e-2f3g-h4ghjk56789l
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Proxy Phone Sessions](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte [DeleteProxySession](#) la Referencia de AWS CLI comandos.

## **delete-room-membership**

En el siguiente ejemplo de código, se muestra cómo usar `delete-room-membership`.

AWS CLI

Para eliminar a un usuario como miembro de una sala de chat

En el siguiente `delete-room-membership` ejemplo, se elimina al miembro especificado de la sala de chat especificada.

```
aws chime delete-room-membership \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \  
  --member-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de una sala de chat](#) en la Guía del usuario de Amazon Chime.

- Para API obtener más información, consulte [DeleteRoomMemberships](#) la Referencia de AWS CLI comandos.

## **delete-room**

En el siguiente ejemplo de código, se muestra cómo usar `delete-room`.

### AWS CLI

Para eliminar una sala de chat

En el siguiente `delete-room` ejemplo, se elimina la sala de chat especificada y se quitan las pertenencias a la sala de chat.

```
aws chime delete-room \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de una sala de chat](#) en la Guía del usuario de Amazon Chime.

- Para API obtener más información, consulte [DeleteRoom](#) la Referencia de AWS CLI comandos.

## **delete-voice-connector-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-voice-connector-group`.

### AWS CLI

title

En el siguiente `delete-voice-connector-group` ejemplo, se elimina el grupo de conectores de voz de Amazon Chime especificado.

```
aws chime delete-voice-connector-group \  
  title
```

```
--voice-connector-group-id 123a456b-c7d8-90e1-fg23-4h567jk18901
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de grupos de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [DeleteVoiceConnectorGroup](#) de AWS CLI comandos.

## **delete-voice-connector-origination**

En el siguiente ejemplo de código, se muestra cómo usar `delete-voice-connector-origination`.

### AWS CLI

Para eliminar la configuración de origen

En el siguiente `delete-voice-connector-origination` ejemplo, se eliminan el host, el puerto, el protocolo, la prioridad y el peso de origen del conector de voz Amazon Chime especificado.

```
aws chime delete-voice-connector-origination \  
--voice-connector-id abcdef1ghij2klmno3pqr4
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [DeleteVoiceConnectorOrigination](#) de AWS CLI comandos.

## **delete-voice-connector-proxy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-voice-connector-proxy`.

### AWS CLI

Para eliminar una configuración de proxy

En el siguiente `delete-voice-connector-proxy` ejemplo, se elimina la configuración de proxy del conector de voz de Amazon Chime.

```
aws chime delete-voice-connector-proxy \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Proxy Phone Sessions](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte [DeleteVoiceConnectorProxy](#) la Referencia de AWS CLI comandos.

## **delete-voice-connector-streaming-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `delete-voice-connector-streaming-configuration`.

AWS CLI

Para eliminar una configuración de streaming

En el siguiente `delete-voice-connector-streaming-configuration` ejemplo, se elimina la configuración de streaming del conector de voz Amazon Chime especificado.

```
aws chime delete-voice-connector-streaming-configuration \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Transmisión de datos del conector de voz de Amazon Chime a Kinesis](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia de [DeleteVoiceConnectorStreamingConfiguration](#) comandos AWS CLI .

## **delete-voice-connector-termination-credentials**

En el siguiente ejemplo de código, se muestra cómo usar `delete-voice-connector-termination-credentials`.

## AWS CLI

Para eliminar las credenciales de terminación

En el siguiente `delete-voice-connector-termination-credentials` ejemplo, se eliminan las credenciales de terminación del nombre de usuario especificado y del conector de voz Amazon Chime.

```
aws chime delete-voice-connector-termination-credentials \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --usernames "jdoe"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [DeleteVoiceConnectorTerminationCredentials](#) de AWS CLI comandos.

## `delete-voice-connector-termination`

En el siguiente ejemplo de código, se muestra cómo usar `delete-voice-connector-termination`.

## AWS CLI

Para eliminar la configuración de terminación

En el siguiente `delete-voice-connector-termination` ejemplo, se elimina la configuración de terminación del conector de voz Amazon Chime especificado.

```
aws chime delete-voice-connector-termination \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [DeleteVoiceConnectorTermination](#) de AWS CLI comandos.



## **delete-voice-connector**

En el siguiente ejemplo de código, se muestra cómo usar `delete-voice-connector`.

### AWS CLI

Para eliminar un conector de voz de Amazon Chime

En el siguiente `delete-voice-connector` ejemplo se hace lo siguiente

```
aws chime delete-voice-connector \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [DeleteVoiceConnector](#) de AWS CLI comandos.

## **disassociate-phone-number-from-user**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-phone-number-from-user`.

### AWS CLI

Para desasociar un número de teléfono de un usuario

El siguiente `disassociate-phone-number-from-user` ejemplo desasocia un número de teléfono del usuario especificado.

```
aws chime disassociate-phone-number-from-user \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --user-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar los números de teléfono de los usuarios](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [DisassociatePhoneNumberFromUser](#) la Referencia de AWS CLI comandos.

## **disassociate-phone-numbers-from-voice-connector-group**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-phone-numbers-from-voice-connector-group`.

### AWS CLI

Para desasociar números de teléfono de un grupo de conectores de voz de Amazon Chime

El siguiente `disassociate-phone-numbers-from-voice-connector-group` ejemplo desasocia los números de teléfono especificados de un grupo de conectores de voz de Amazon Chime.

```
aws chime disassociate-phone-numbers-from-voice-connector-group \
  --voice-connector-group-id 123a456b-c7d8-90e1-fg23-4h567jkl8901 \
  --e164-phone-numbers "+12065550100" "+12065550101"
```

Salida:

```
{
  "PhoneNumberErrors": []
}
```

Para obtener más información, consulte [Uso de grupos de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [DisassociatePhoneNumbersFromVoiceConnectorGroup](#) de AWS CLI comandos.

## **disassociate-phone-numbers-from-voice-connector**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-phone-numbers-from-voice-connector`.

### AWS CLI

Para desasociar números de teléfono de un conector de voz de Amazon Chime

El siguiente `disassociate-phone-numbers-from-voice-connector` ejemplo desasocia los números de teléfono especificados de un conector de voz de Amazon Chime.

```
aws chime disassociate-phone-numbers-from-voice-connector \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --e164-phone-numbers " +12065550100" "+12065550101"
```

Salida:

```
{  
  "PhoneNumberErrors": []  
}
```

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [DisassociatePhoneNumbersFromVoiceConnector](#) de AWS CLI comandos.

## **disassociate-signin-delegate-groups-from-account**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-signin-delegate-groups-from-account`.

AWS CLI

Para desasociar los grupos de delegados que inician sesión

El siguiente `disassociate-signin-delegate-groups-from-account` ejemplo desasocia el grupo de delegados de inicio de sesión especificado de la cuenta de Amazon Chime especificada.

```
aws chime disassociate-signin-delegate-groups-from-account \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --group-names "my_users"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar el acceso y los permisos de los usuarios](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [DisassociateSigninDelegateGroupsFromAccount](#) la Referencia de AWS CLI comandos.

## get-account-settings

En el siguiente ejemplo de código, se muestra cómo usar `get-account-settings`.

### AWS CLI

Para recuperar la configuración de una cuenta

En el siguiente `get-account-settings` ejemplo, se recupera la configuración de la cuenta especificada.

```
aws chime get-account-settings --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Salida:

```
{
  "AccountSettings": {
    "DisableRemoteControl": false,
    "EnableDialOut": false
  }
}
```

Para obtener más información, consulte [Gestión de sus cuentas de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [GetAccountSettings](#) de AWS CLI comandos.

## get-account

En el siguiente ejemplo de código, se muestra cómo usar `get-account`.

### AWS CLI

Para recuperar los detalles de una cuenta

En el siguiente `get-account` ejemplo, se recuperan los detalles de la cuenta de Amazon Chime especificada.

```
aws chime get-account \  
--account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Salida:

```
{  
  "Account": {  
    "AwsAccountId": "111122223333",  
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "Name": "EnterpriseDirectory",  
    "AccountType": "EnterpriseDirectory",  
    "CreatedTimestamp": "2018-12-20T18:38:02.181Z",  
    "DefaultLicense": "Pro",  
    "SupportedLicenses": [  
      "Basic",  
      "Pro"  
    ],  
    "SigninDelegateGroups": [  
      {  
        "GroupName": "myGroup"  
      },  
    ]  
  }  
}
```

Para obtener más información, consulte [Gestión de sus cuentas de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [GetAccount](#) de AWS CLI comandos.

## get-bot

En el siguiente ejemplo de código, se muestra cómo usar get-bot.

### AWS CLI

Para recuperar detalles sobre un bot

En el siguiente get-bot ejemplo, se muestran los detalles del bot especificado.

```
aws chime get-bot \  
--account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
--bot-id 12345678-90ab-cdef-11111EXAMPLE
```

```
--bot-id 123abcd4-5ef6-789g-0h12-34j56789012k
```

Salida:

```
{
  "Bot": {
    "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",
    "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",
    "DisplayName": "myBot (Bot)",
    "BotType": "ChatBot",
    "Disabled": false,
    "CreatedTimestamp": "2019-09-09T18:05:56.749Z",
    "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",
    "BotEmail": "myBot-chimebot@example.com",
    "SecurityToken": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY"
  }
}
```

Para obtener más información, consulte [Actualizar los bots de chat](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte [GetBot](#) la Referencia de AWS CLI comandos.

## get-global-settings

En el siguiente ejemplo de código, se muestra cómo usar `get-global-settings`.

AWS CLI

Para obtener la configuración global

En el siguiente `get-global-settings` ejemplo, se recuperan los nombres de los buckets de S3 que se utilizan para almacenar los registros de detalles de llamadas de los conectores Amazon Chime Business Calling y Amazon Chime Voice asociados a la cuenta del administrador.  
AWS

```
aws chime get-global-settings
```

Salida:

```
{
  "BusinessCalling": {
```

```
    "CdrBucket": "s3bucket"
  },
  "VoiceConnector": {
    "CdrBucket": "s3bucket"
  }
}
```

Para obtener más información, consulte [Gestión de la configuración global](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [GetGlobalSettings](#) la Referencia de AWS CLI comandos.

## get-phone-number-order

En el siguiente ejemplo de código, se muestra cómo usar `get-phone-number-order`.

### AWS CLI

Para obtener los detalles de un pedido con un número de teléfono

En el siguiente `get-phone-number-order` ejemplo, se muestran los detalles del pedido de números de teléfono especificado.

```
aws chime get-phone-number-order \
  --phone-number-order-id abc12345-de67-89f0-123g-h45i678j9012
```

Salida:

```
{
  "PhoneNumberOrder": {
    "PhoneNumberOrderId": "abc12345-de67-89f0-123g-h45i678j9012",
    "ProductType": "VoiceConnector",
    "Status": "Partial",
    "OrderedPhoneNumbers": [
      {
        "E164PhoneNumber": "+12065550100",
        "Status": "Acquired"
      },
      {
        "E164PhoneNumber": "+12065550101",
        "Status": "Acquired"
      }
    ]
  }
}
```

```
    },
    {
      "E164PhoneNumber": "+12065550102",
      "Status": "Failed"
    }
  ],
  "CreatedTimestamp": "2019-08-09T21:35:21.427Z",
  "UpdatedTimestamp": "2019-08-09T21:35:31.926Z"
}
```

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [GetPhoneNumberOrder](#) la Referencia de AWS CLI comandos.

## get-phone-number-settings

En el siguiente ejemplo de código, se muestra cómo usar `get-phone-number-settings`.

### AWS CLI

Para recuperar el nombre de una llamada saliente

En el siguiente `get-phone-number-settings` ejemplo, se recupera el nombre de llamada saliente predeterminado de la cuenta del usuario que realiza la llamada. AWS

```
aws chime get-phone-number-settings
```

Este comando no genera ninguna salida. Salida:

```
{
  "CallingName": "myName",
  "CallingNameUpdatedTimestamp": "2019-10-28T18:56:42.911Z"
}
```

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [GetPhoneNumberSettings](#) la Referencia de AWS CLI comandos.



## get-phone-number

En el siguiente ejemplo de código, se muestra cómo usar `get-phone-number`.

### AWS CLI

Para obtener los detalles del número de teléfono

En el siguiente `get-phone-number` ejemplo, se muestran los detalles del número de teléfono especificado.

```
aws chime get-phone-number \
  --phone-number-id +12065550100
```

Salida:

```
{
  "PhoneNumber": {
    "PhoneNumberId": "%2B12065550100",
    "E164PhoneNumber": "+12065550100",
    "Type": "Local",
    "ProductType": "VoiceConnector",
    "Status": "Unassigned",
    "Capabilities": {
      "InboundCall": true,
      "OutboundCall": true,
      "InboundSMS": true,
      "OutboundSMS": true,
      "InboundMMS": true,
      "OutboundMMS": true
    },
    "Associations": [
      {
        "Value": "abcdef1ghij2klmno3pqr4",
        "Name": "VoiceConnectorId",
        "AssociatedTimestamp": "2019-10-28T18:40:37.453Z"
      }
    ],
    "CallingNameStatus": "UpdateInProgress",
    "CreatedTimestamp": "2019-08-09T21:35:21.445Z",
    "UpdatedTimestamp": "2019-08-09T21:35:31.745Z"
  }
}
```

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [GetPhoneNumber](#) la Referencia de AWS CLI comandos.

## get-proxy-session

En el siguiente ejemplo de código, se muestra cómo usar `get-proxy-session`.

### AWS CLI

Para obtener los detalles de la sesión de proxy

En el siguiente `get-proxy-session` ejemplo, se muestran los detalles de la sesión de proxy especificada.

```
aws chime get-proxy-session \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --proxy-session-id 123a4bc5-67d8-901e-2f3g-h4ghjk567891
```

Salida:

```
{  
  "ProxySession": {  
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",  
    "ProxySessionId": "123a4bc5-67d8-901e-2f3g-h4ghjk567891",  
    "Status": "Open",  
    "ExpiryMinutes": 60,  
    "Capabilities": [  
      "SMS",  
      "Voice"  
    ],  
    "CreatedTimestamp": "2020-04-15T16:10:10.288Z",  
    "UpdatedTimestamp": "2020-04-15T16:10:10.288Z",  
    "Participants": [  
      {  
        "PhoneNumber": "+12065550100",  
        "ProxyPhoneNumber": "+19135550199"  
      },  
      {  
        "PhoneNumber": "+14015550101",
```

```

        "ProxyPhoneNumber": "+19135550199"
      }
    ]
  }
}

```

Para obtener más información, consulte [Proxy Phone Sessions](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte [GetProxySession](#) la Referencia de AWS CLI comandos.

## get-room

En el siguiente ejemplo de código, se muestra cómo usar `get-room`.

### AWS CLI

Para obtener los detalles sobre una sala de chat

En el siguiente `get-room` ejemplo, se muestran detalles sobre la sala de chat especificada.

```

aws chime get-room \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j

```

Salida:

```

{
  "Room": {
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
    "Name": "chatRoom",
    "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",
    "CreatedBy": "arn:aws:iam::111122223333:user/alejandro",
    "CreatedTimestamp": "2019-12-02T22:29:31.549Z",
    "UpdatedTimestamp": "2019-12-02T22:29:31.549Z"
  }
}

```

Para obtener más información, consulte [Creación de una sala de chat](#) en la Guía del usuario de Amazon Chime.

- Para API obtener más información, consulte [GetRoom](#) la Referencia de AWS CLI comandos.

## get-user-settings

En el siguiente ejemplo de código, se muestra cómo usar `get-user-settings`.

### AWS CLI

Para recuperar la configuración del usuario

En el siguiente `get-user-settings` ejemplo, se muestra la configuración de usuario especificada.

```
aws chime get-user-settings \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --user-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Salida:

```
{
  "UserSettings": {
    "Telephony": {
      "InboundCalling": true,
      "OutboundCalling": true,
      "SMS": true
    }
  }
}
```

Para obtener más información, consulte [Administrar los números de teléfono de los usuarios](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [GetUserSettings](#) la Referencia de AWS CLI comandos.

## get-user

En el siguiente ejemplo de código, se muestra cómo usar `get-user`.

### AWS CLI

Para obtener detalles sobre un usuario

En el siguiente `get-user` ejemplo, se recuperan los detalles del usuario especificado.

```
aws chime get-user \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --user-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Salida:

```
{  
  "User": {  
    "UserId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "PrimaryEmail": "marthar@example.com",  
    "DisplayName": "Martha Rivera",  
    "LicenseType": "Pro",  
    "UserRegistrationStatus": "Registered",  
    "RegisteredOn": "2018-12-20T18:45:25.231Z",  
    "InvitedOn": "2018-12-20T18:45:25.231Z",  
    "AlexaForBusinessMetadata": {  
      "IsAlexaForBusinessEnabled": False,  
      "AlexaForBusinessRoomArn": "null"  
    },  
    "PersonalPIN": "XXXXXXXXXX"  
  }  
}
```

Para obtener más información, consulte [Administración de usuarios](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [GetUser](#) la Referencia de AWS CLI comandos.

## get-voice-connector-group

En el siguiente ejemplo de código, se muestra cómo usar `get-voice-connector-group`.

### AWS CLI

Para obtener información sobre un grupo de conectores de voz de Amazon Chime

En el siguiente `get-voice-connector-group` ejemplo, se muestran los detalles del grupo de conectores de voz de Amazon Chime especificado.

```
aws chime get-voice-connector-group \  
  --voice-connector-group-id 123a456b-c7d8-90e1-fg23-4h567jkl8901
```

Salida:

```
{  
  "VoiceConnectorGroup": {  
    "VoiceConnectorGroupId": "123a456b-c7d8-90e1-fg23-4h567jkl8901",  
    "Name": "myGroup",  
    "VoiceConnectorItems": [],  
    "CreatedTimestamp": "2019-09-18T16:38:34.734Z",  
    "UpdatedTimestamp": "2019-09-18T16:38:34.734Z"  
  }  
}
```

Para obtener más información, consulte [Uso de grupos de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [GetVoiceConnectorGroup](#) de AWS CLI comandos.

## get-voice-connector-logging-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-voice-connector-logging-configuration`.

AWS CLI

Para obtener los detalles de la configuración del registro

En el siguiente `get-voice-connector-logging-configuration` ejemplo, se recuperan los detalles de la configuración de registro del conector de voz Amazon Chime especificado.

```
aws chime get-voice-connector-logging-configuration \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Salida:

```
{
```

```
"LoggingConfiguration": {
  "EnableSIPLogs": true
}
}
```

Para obtener más información, consulte [Transmisión de contenido multimedia de Amazon Chime Voice Connector a Kinesis](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia de [GetVoiceConnectorLoggingConfiguration](#) comandos AWS CLI .

## get-voice-connector-origination

En el siguiente ejemplo de código, se muestra cómo usar `get-voice-connector-origination`.

### AWS CLI

Para recuperar la configuración de origen

El siguiente `get-voice-connector-origination` ejemplo recupera el host de origen, el puerto, el protocolo, la prioridad y el peso del conector de voz Amazon Chime especificado.

```
aws chime get-voice-connector-origination \
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Salida:

```
{
  "Origination": {
    "Routes": [
      {
        "Host": "10.24.34.0",
        "Port": 1234,
        "Protocol": "TCP",
        "Priority": 1,
        "Weight": 5
      }
    ],
    "Disabled": false
  }
}
```

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [GetVoiceConnectorOrigination](#) de AWS CLI comandos.

## get-voice-connector-proxy

En el siguiente ejemplo de código, se muestra cómo usar `get-voice-connector-proxy`.

### AWS CLI

Para obtener los detalles de la configuración del proxy

En el siguiente `get-voice-connector-proxy` ejemplo, se obtienen los detalles de configuración del proxy de su conector de voz Amazon Chime.

```
aws chime get-voice-connector-proxy \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Salida:

```
{  
  "Proxy": {  
    "DefaultSessionExpiryMinutes": 60,  
    "Disabled": false,  
    "PhoneNumberCountries": [  
      "US"  
    ]  
  }  
}
```

Para obtener más información, consulte [Proxy Phone Sessions](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte [GetVoiceConnectorProxy](#) la Referencia de AWS CLI comandos.

## get-voice-connector-streaming-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-voice-connector-streaming-configuration`.



## AWS CLI

Para obtener los detalles de la configuración de streaming

En el siguiente `get-voice-connector-streaming-configuration` ejemplo, se obtienen los detalles de configuración de streaming del conector de voz Amazon Chime especificado.

```
aws chime get-voice-connector-streaming-configuration \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Salida:

```
{  
  "StreamingConfiguration": {  
    "DataRetentionInHours": 24,  
    "Disabled": false  
  }  
}
```

Para obtener más información, consulte [Transmisión de datos del conector de voz de Amazon Chime a Kinesis](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia de [GetVoiceConnectorStreamingConfiguration](#) comandos AWS CLI .

## `get-voice-connector-termination-health`

En el siguiente ejemplo de código, se muestra cómo usar `get-voice-connector-termination-health`.

## AWS CLI

Para recuperar los detalles del estado de la rescisión

En el siguiente `get-voice-connector-termination-health` ejemplo, se recuperan los detalles del estado de terminación del conector de voz Amazon Chime especificado.

```
aws chime get-voice-connector-termination-health \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Salida:

```
{
  "TerminationHealth": {
    "Timestamp": "Fri Aug 23 16:45:55 UTC 2019",
    "Source": "10.24.34.0"
  }
}
```

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [GetVoiceConnectorTerminationHealth](#) de AWS CLI comandos.

## get-voice-connector-termination

En el siguiente ejemplo de código, se muestra cómo usar `get-voice-connector-termination`.

### AWS CLI

Para recuperar la configuración de terminación

En el siguiente `get-voice-connector-termination` ejemplo, se recupera la configuración de terminación del conector de voz Amazon Chime especificado.

```
aws chime get-voice-connector-termination \
  --voice-connector-id abcdefghijklmno3pqr4
```

Este comando no genera ninguna salida. Salida:

```
{
  "Termination": {
    "CpsLimit": 1,
    "DefaultPhoneNumber": "+12065550100",
    "CallingRegions": [
      "US"
    ],
    "CidrAllowedList": [
      "10.24.34.0/23"
    ],
    "Disabled": false
  }
}
```

```
}
```

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [GetVoiceConnectorTermination](#) de AWS CLI comandos.

## get-voice-connector

En el siguiente ejemplo de código, se muestra cómo usar `get-voice-connector`.

### AWS CLI

Para obtener detalles de un conector de voz de Amazon Chime

En el siguiente `get-voice-connector` ejemplo, se muestran los detalles del conector de voz Amazon Chime especificado.

```
aws chime get-voice-connector \  
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Salida:

```
{  
  "VoiceConnector": {  
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",  
    "AwsRegion": "us-west-2",  
    "Name": "newVoiceConnector",  
    "OutboundHostName": "abcdef1ghij2klmno3pqr4.voiceconnector.chime.aws",  
    "RequireEncryption": true,  
    "CreatedTimestamp": "2019-09-18T20:34:01.352Z",  
    "UpdatedTimestamp": "2019-09-18T20:34:01.352Z"  
  }  
}
```

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [GetVoiceConnector](#) de AWS CLI comandos.

## invite-users

En el siguiente ejemplo de código, se muestra cómo usar `invite-users`.

### AWS CLI

Para invitar a los usuarios a unirse a Amazon Chime

El siguiente `invite-users` ejemplo envía un correo electrónico para invitar a un usuario a la cuenta de Amazon Chime especificada.

```
aws chime invite-users \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --user-email-list "alejandr@example.com" "janed@example.com"
```

Salida:

```
{  
  "Invites": [  
    {  
      "InviteId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
      "Status": "Pending",  
      "EmailAddress": "alejandr@example.com",  
      "EmailStatus": "Sent"  
    },  
    {  
      "InviteId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",  
      "Status": "Pending",  
      "EmailAddress": "janed@example.com",  
      "EmailStatus": "Sent"  
    }  
  ]  
}
```

Para obtener más información, consulte [Invitación y suspensión de usuarios](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [InviteUsers](#) de AWS CLI comandos.

## list-accounts

En el siguiente ejemplo de código, se muestra cómo usar `list-accounts`.

## AWS CLI

Para obtener una lista de cuentas

En el siguiente `list-accounts` ejemplo, se recupera una lista de las cuentas de Amazon Chime de la cuenta del administrador AWS .

```
aws chime list-accounts
```

Salida:

```
{
  "Accounts": [
    {
      "AwsAccountId": "111122223333",
      "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "Name": "First Chime Account",
      "AccountType": "EnterpriseDirectory",
      "CreatedTimestamp": "2018-12-20T18:38:02.181Z",
      "DefaultLicense": "Pro",
      "SupportedLicenses": [
        "Basic",
        "Pro"
      ],
      "SigninDelegateGroups": [
        {
          "GroupName": "myGroup"
        }
      ]
    },
    {
      "AwsAccountId": "111122223333",
      "AccountId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
      "Name": "Second Chime Account",
      "AccountType": "Team",
      "CreatedTimestamp": "2018-09-04T21:44:22.292Z",
      "DefaultLicense": "Pro",
      "SupportedLicenses": [
        "Basic",
        "Pro"
      ],
      "SigninDelegateGroups": [
        {
```

```

    "GroupName": "myGroup"
  },
]
}

```

Para obtener más información, consulte [Gestión de sus cuentas de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [ListAccounts](#) de AWS CLI comandos.

## list-bots

En el siguiente ejemplo de código, se muestra cómo usar `list-bots`.

### AWS CLI

Para recuperar una lista de bots

En el siguiente `list-bots` ejemplo, se enumeran los bots asociados a la cuenta de Amazon Chime Enterprise especificada.

```

aws chime list-bots \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45

```

Salida:

```

{
  "Bot": {
    "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",
    "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",
    "DisplayName": "myBot (Bot)",
    "BotType": "ChatBot",
    "Disabled": false,
    "CreatedTimestamp": "2019-09-09T18:05:56.749Z",
    "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",
    "BotEmail": "myBot-chimebot@example.com",
    "SecurityToken": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY"
  }
}

```

Para obtener más información, consulte [Uso de bots de chat con Amazon Chime](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte la Referencia [ListBots](#) de AWS CLI comandos.

## **list-phone-number-orders**

En el siguiente ejemplo de código, se muestra cómo usar `list-phone-number-orders`.

### AWS CLI

Para enumerar los pedidos de números de teléfono

El siguiente `list-phone-number-orders` ejemplo muestra los pedidos con números de teléfono asociados a la cuenta del administrador de Amazon Chime.

```
aws chime list-phone-number-orders
```

Salida:

```
{
  "PhoneNumberOrders": [
    {
      "PhoneNumberOrderId": "abc12345-de67-89f0-123g-h45i678j9012",
      "ProductType": "VoiceConnector",
      "Status": "Partial",
      "OrderedPhoneNumbers": [
        {
          "E164PhoneNumber": "+12065550100",
          "Status": "Acquired"
        },
        {
          "E164PhoneNumber": "+12065550101",
          "Status": "Acquired"
        },
        {
          "E164PhoneNumber": "+12065550102",
          "Status": "Failed"
        }
      ],
      "CreatedTimestamp": "2019-08-09T21:35:21.427Z",
      "UpdatedTimestamp": "2019-08-09T21:35:31.926Z"
    }
  ]
}
```

```

    {
      "PhoneNumberOrderId": "cba54321-ed76-09f5-321g-h54i876j2109",
      "ProductType": "BusinessCalling",
      "Status": "Partial",
      "OrderedPhoneNumbers": [
        {
          "E164PhoneNumber": "+12065550103",
          "Status": "Acquired"
        },
        {
          "E164PhoneNumber": "+12065550104",
          "Status": "Acquired"
        },
        {
          "E164PhoneNumber": "+12065550105",
          "Status": "Failed"
        }
      ],
      "CreatedTimestamp": "2019-08-09T21:35:21.427Z",
      "UpdatedTimestamp": "2019-08-09T21:35:31.926Z"
    }
  ]
}

```

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [ListPhoneNumberOrders](#) la Referencia de AWS CLI comandos.

## list-phone-numbers

En el siguiente ejemplo de código, se muestra cómo usar `list-phone-numbers`.

### AWS CLI

Para enumerar los números de teléfono de una cuenta de Amazon Chime

El siguiente `list-phone-numbers` ejemplo muestra los números de teléfono asociados a la cuenta Amazon Chime del administrador.

```
aws chime list-phone-numbers
```



Este comando no genera ninguna salida. Salida:

```
{
  "PhoneNumbers": [
    {
      "PhoneNumberId": "%2B12065550100",
      "E164PhoneNumber": "+12065550100",
      "Type": "Local",
      "ProductType": "VoiceConnector",
      "Status": "Assigned",
      "Capabilities": {
        "InboundCall": true,
        "OutboundCall": true,
        "InboundSMS": true,
        "OutboundSMS": true,
        "InboundMMS": true,
        "OutboundMMS": true
      },
      "Associations": [
        {
          "Value": "abcdef1ghij2klmno3pqr4",
          "Name": "VoiceConnectorId",
          "AssociatedTimestamp": "2019-10-28T18:40:37.453Z"
        }
      ],
      "CallingNameStatus": "UpdateInProgress",
      "CreatedTimestamp": "2019-08-12T22:10:20.521Z",
      "UpdatedTimestamp": "2019-10-28T18:42:07.964Z"
    },
    {
      "PhoneNumberId": "%2B12065550101",
      "E164PhoneNumber": "+12065550101",
      "Type": "Local",
      "ProductType": "VoiceConnector",
      "Status": "Assigned",
      "Capabilities": {
        "InboundCall": true,
        "OutboundCall": true,
        "InboundSMS": true,
        "OutboundSMS": true,
        "InboundMMS": true,
        "OutboundMMS": true
      },
      "Associations": [
```

```

        {
            "Value": "abcdef1ghij2klmno3pqr4",
            "Name": "VoiceConnectorId",
            "AssociatedTimestamp": "2019-10-28T18:40:37.511Z"
        }
    ],
    "CallingNameStatus": "UpdateInProgress",
    "CreatedTimestamp": "2019-08-12T22:10:20.521Z",
    "UpdatedTimestamp": "2019-10-28T18:42:07.960Z"
}
]
}

```

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [ListPhoneNumbers](#) la Referencia de AWS CLI comandos.

## list-proxy-sessions

En el siguiente ejemplo de código, se muestra cómo usar `list-proxy-sessions`.

### AWS CLI

Para ver una lista de las sesiones de proxy

En el siguiente `list-proxy-sessions` ejemplo, se enumeran las sesiones de proxy de su conector de voz Amazon Chime.

```

aws chime list-proxy-sessions \
  --voice-connector-id abcdef1ghij2klmno3pqr4

```

Salida:

```

{
  "ProxySession": {
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",
    "ProxySessionId": "123a4bc5-67d8-901e-2f3g-h4ghjk567891",
    "Status": "Open",
    "ExpiryMinutes": 60,
    "Capabilities": [

```

```

        "SMS",
        "Voice"
    ],
    "CreatedTimestamp": "2020-04-15T16:10:10.288Z",
    "UpdatedTimestamp": "2020-04-15T16:10:10.288Z",
    "Participants": [
        {
            "PhoneNumber": "+12065550100",
            "ProxyPhoneNumber": "+19135550199"
        },
        {
            "PhoneNumber": "+14015550101",
            "ProxyPhoneNumber": "+19135550199"
        }
    ]
}
}

```

Para obtener más información, consulte [Proxy Phone Sessions](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte [ListProxySessions](#) la Referencia de AWS CLI comandos.

## list-room-memberships

En el siguiente ejemplo de código, se muestra cómo usar `list-room-memberships`.

### AWS CLI

Para ver una lista de las membresías de salas

En el siguiente `list-room-memberships` ejemplo, se muestra una lista de los detalles de pertenencia a la sala de chat especificada.

```

aws chime list-room-memberships \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j

```

Salida:

```
{
```

```
"RoomMemberships": [  
  {  
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",  
    "Member": {  
      "MemberId": "2ab2345c-67de-8901-f23g-45h678901j2k",  
      "MemberType": "User",  
      "Email": "zhangw@example.com",  
      "FullName": "Zhang Wei",  
      "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45"  
    },  
    "Role": "Member",  
    "InvitedBy": "arn:aws:iam::111122223333:user/alejandro",  
    "UpdatedTimestamp": "2019-12-02T22:46:58.532Z"  
  },  
  {  
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",  
    "Member": {  
      "MemberId": "1ab2345c-67de-8901-f23g-45h678901j2k",  
      "MemberType": "User",  
      "Email": "janed@example.com",  
      "FullName": "Jane Doe",  
      "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45"  
    },  
    "Role": "Administrator",  
    "InvitedBy": "arn:aws:iam::111122223333:user/alejandro",  
    "UpdatedTimestamp": "2019-12-02T22:46:58.532Z"  
  }  
]  
}
```

Para obtener más información, consulte [Creación de una sala de chat](#) en la Guía del usuario de Amazon Chime.

- Para API obtener más información, consulte [ListRoomMemberships](#) la Referencia de AWS CLI comandos.

## list-rooms

En el siguiente ejemplo de código, se muestra cómo usar `list-rooms`.

### AWS CLI

Para ver una lista de salas de chat

En el siguiente `list-rooms` ejemplo, se muestra una lista de salas de chat de la cuenta especificada. La lista se filtra solo para incluir las salas de chat a las que pertenece el miembro especificado.

```
aws chime list-rooms \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --member-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Salida:

```
{  
  "Room": {  
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",  
    "Name": "teamRoom",  
    "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",  
    "CreatedBy": "arn:aws:iam::111122223333:user/alejandro",  
    "CreatedTimestamp": "2019-12-02T22:29:31.549Z",  
    "UpdatedTimestamp": "2019-12-02T22:33:19.310Z"  
  }  
}
```

Para obtener más información, consulte [Creación de una sala de chat](#) en la Guía del usuario de Amazon Chime.

- Para API obtener más información, consulte [ListRooms](#) la Referencia de AWS CLI comandos.

## list-users

En el siguiente ejemplo de código, se muestra cómo usar `list-users`.

### AWS CLI

Para enumerar los usuarios de una cuenta

En el siguiente `list-users` ejemplo, se enumeran los usuarios de la cuenta de Amazon Chime especificada.

```
aws chime list-users --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Salida:

```
{
```

```
"Users": [  
  {  
    "UserId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "PrimaryEmail": "mariag@example.com",  
    "DisplayName": "Maria Garcia",  
    "LicenseType": "Pro",  
    "UserType": "PrivateUser",  
    "UserRegistrationStatus": "Registered",  
    "RegisteredOn": "2018-12-20T18:45:25.231Z"  
    "AlexaForBusinessMetadata": {  
      "IsAlexaForBusinessEnabled": false  
    }  
  },  
  {  
    "UserId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",  
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "PrimaryEmail": "richardr@example.com",  
    "DisplayName": "Richard Roe",  
    "LicenseType": "Pro",  
    "UserType": "PrivateUser",  
    "UserRegistrationStatus": "Registered",  
    "RegisteredOn": "2018-12-20T18:45:45.415Z"  
    "AlexaForBusinessMetadata": {  
      "IsAlexaForBusinessEnabled": false  
    }  
  },  
  {  
    "UserId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",  
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "PrimaryEmail": "saanvis@example.com",  
    "DisplayName": "Saanvi Sarkar",  
    "LicenseType": "Basic",  
    "UserType": "PrivateUser",  
    "UserRegistrationStatus": "Registered",  
    "RegisteredOn": "2018-12-20T18:46:57.747Z"  
    "AlexaForBusinessMetadata": {  
      "IsAlexaForBusinessEnabled": false  
    }  
  },  
  {  
    "UserId": "a1b2c3d4-5678-90ab-cdef-55555EXAMPLE",  
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "PrimaryEmail": "wxiulan@example.com",
```

```

        "DisplayName": "Wang Xiulan",
        "LicenseType": "Basic",
        "UserType": "PrivateUser",
        "UserRegistrationStatus": "Registered",
        "RegisteredOn": "2018-12-20T18:47:15.390Z"
        "AlexaForBusinessMetadata": {
            "IsAlexaForBusinessEnabled": false
        }
    }
]
}

```

Para obtener más información, consulte [Administración de usuarios](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [ListUsers](#) la Referencia de AWS CLI comandos.

## list-voice-connector-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-voice-connector-groups`.

### AWS CLI

Para enumerar los grupos de conectores de voz de Amazon Chime para una cuenta de Amazon Chime

En el siguiente `list-voice-connector-groups` ejemplo, se enumeran los grupos de conectores de voz de Amazon Chime asociados a la cuenta de Amazon Chime del administrador.

```
aws chime list-voice-connector-groups
```

Salida:

```

{
  "VoiceConnectorGroups": [
    {
      "VoiceConnectorGroupId": "123a456b-c7d8-90e1-fg23-4h567jk18901",
      "Name": "myGroup",
      "VoiceConnectorItems": [],
      "CreatedTimestamp": "2019-09-18T16:38:34.734Z",
      "UpdatedTimestamp": "2019-09-18T16:38:34.734Z"
    }
  ]
}

```

```
]
}
```

Para obtener más información, consulte [Uso de grupos de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [ListVoiceConnectorGroups](#) de AWS CLI comandos.

## **list-voice-connector-termination-credentials**

En el siguiente ejemplo de código, se muestra cómo usar `list-voice-connector-termination-credentials`.

### AWS CLI

Para recuperar una lista de credenciales de terminación

En el siguiente `list-voice-connector-termination-credentials` ejemplo, se recupera una lista de las credenciales de terminación del conector de voz Amazon Chime especificado.

```
aws chime list-voice-connector-termination-credentials \
  --voice-connector-id abcdef1ghij2klmno3pqr4
```

Este comando no genera ninguna salida. Salida:

```
{
  "Usernames": [
    "jdoe"
  ]
}
```

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [ListVoiceConnectorTerminationCredentials](#) de AWS CLI comandos.

## **list-voice-connectors**

En el siguiente ejemplo de código, se muestra cómo usar `list-voice-connectors`.



## AWS CLI

Para enumerar los conectores de voz de Amazon Chime para una cuenta

En el siguiente `list-voice-connectors` ejemplo, se enumeran los conectores de voz de Amazon Chime asociados a la cuenta de la persona que llama.

```
aws chime list-voice-connectors
```

Salida:

```
{
  "VoiceConnectors": [
    {
      "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",
      "AwsRegion": "us-east-1",
      "Name": "MyVoiceConnector",
      "OutboundHostName": "abcdef1ghij2klmno3pqr4.voiceconnector.chime.aws",
      "RequireEncryption": true,
      "CreatedTimestamp": "2019-06-04T18:46:56.508Z",
      "UpdatedTimestamp": "2019-09-18T16:33:00.806Z"
    },
    {
      "VoiceConnectorId": "cbadef1ghij2klmno3pqr5",
      "AwsRegion": "us-west-2",
      "Name": "newVoiceConnector",
      "OutboundHostName": "cbadef1ghij2klmno3pqr5.voiceconnector.chime.aws",
      "RequireEncryption": true,
      "CreatedTimestamp": "2019-09-18T20:34:01.352Z",
      "UpdatedTimestamp": "2019-09-18T20:34:01.352Z"
    }
  ]
}
```

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [ListVoiceConnectors](#) de AWS CLI comandos.

## logout-user

En el siguiente ejemplo de código, se muestra cómo usar `logout-user`.

### AWS CLI

Para cerrar la sesión de un usuario

En el siguiente `logout-user` ejemplo, se cierra la sesión del usuario especificado.

```
aws chime logout-user \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --user-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [LogoutUser](#) la Referencia de AWS CLI comandos.

## put-voice-connector-logging-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-voice-connector-logging-configuration`.

### AWS CLI

Para añadir una configuración de registro para un conector de voz de Amazon Chime

El siguiente `put-voice-connector-logging-configuration` ejemplo activa la configuración de SIP registro para el conector de voz Amazon Chime especificado.

```
aws chime put-voice-connector-logging-configuration \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --logging-configuration EnableSIPLogs=true
```

Salida:

```
{  
  "LoggingConfiguration": {  
    "EnableSIPLogs": true  
  }  
}
```

Para obtener más información, consulte [Transmisión de contenido multimedia de Amazon Chime Voice Connector a Kinesis](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia de [PutVoiceConnectorLoggingConfiguration](#) comandos AWS CLI .

## put-voice-connector-origination

En el siguiente ejemplo de código, se muestra cómo usar `put-voice-connector-origination`.

### AWS CLI

Para configurar los ajustes de origen

El siguiente `put-voice-connector-origination` ejemplo configura el host de origen, el puerto, el protocolo, la prioridad y el peso del conector de voz Amazon Chime especificado.

```
aws chime put-voice-connector-origination \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --origination  
  Routes=[{Host="10.24.34.0",Port=1234,Protocol="TCP",Priority=1,Weight=5}],Disabled=false
```

Salida:

```
{  
  "Origination": {  
    "Routes": [  
      {  
        "Host": "10.24.34.0",  
        "Port": 1234,  
        "Protocol": "TCP",  
        "Priority": 1,  
        "Weight": 5  
      }  
    ],  
    "Disabled": false  
  }  
}
```

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [PutVoiceConnectorOrigination](#) de AWS CLI comandos.

## put-voice-connector-proxy

En el siguiente ejemplo de código, se muestra cómo usar `put-voice-connector-proxy`.

### AWS CLI

Para poner una configuración de proxy

El siguiente `put-voice-connector-proxy` ejemplo establece una configuración de proxy para su conector de voz Amazon Chime.

```
aws chime put-voice-connector-proxy \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --default-session-expiry-minutes 60 \  
  --phone-number-pool-countries "US"
```

Salida:

```
{  
  "Proxy": {  
    "DefaultSessionExpiryMinutes": 60,  
    "Disabled": false,  
    "PhoneNumberCountries": [  
      "US"  
    ]  
  }  
}
```

Para obtener más información, consulte [Proxy Phone Sessions](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte [PutVoiceConnectorProxy](#) la Referencia de AWS CLI comandos.

## put-voice-connector-streaming-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-voice-connector-streaming-configuration`.

## AWS CLI

Para crear una configuración de streaming

En el siguiente `put-voice-connector-streaming-configuration` ejemplo, se crea una configuración de streaming para el conector de voz Amazon Chime especificado. Permite la transmisión de contenido multimedia desde el conector de voz de Amazon Chime a Amazon Kinesis y establece el período de retención de datos en 24 horas.

```
aws chime put-voice-connector-streaming-configuration \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --streaming-configuration DataRetentionInHours=24,Disabled=false
```

Salida:

```
{  
  "StreamingConfiguration": {  
    "DataRetentionInHours": 24,  
    "Disabled": false  
  }  
}
```

Para obtener más información, consulte [Transmisión de datos del conector de voz de Amazon Chime a Kinesis](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia de [PutVoiceConnectorStreamingConfiguration](#) comandos AWS CLI .

## `put-voice-connector-termination-credentials`

En el siguiente ejemplo de código, se muestra cómo usar `put-voice-connector-termination-credentials`.

### AWS CLI

Para configurar las credenciales de terminación

El siguiente `put-voice-connector-termination-credentials` ejemplo establece las credenciales de terminación para el conector de voz Amazon Chime especificado.

```
aws chime put-voice-connector-termination-credentials \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --termination-credentials arn:aws:kms:us-east-1:123456789012:key/12345678-1234-5678-9012-123456789012
```

```
--voice-connector-id abcdef1ghij2klmno3pqr4 \  
--credentials Username="jdoe",Password="XXXXXXXX"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [PutVoiceConnectorTerminationCredentials](#) de AWS CLI comandos.

## put-voice-connector-termination

En el siguiente ejemplo de código, se muestra cómo usar `put-voice-connector-termination`.

### AWS CLI

Para configurar los ajustes de terminación

El siguiente `put-voice-connector-termination` ejemplo establece las regiones de llamada y la configuración de terminación del host IP permitido para el conector de voz Amazon Chime especificado.

```
aws chime put-voice-connector-termination \  
--voice-connector-id abcdef1ghij2klmno3pqr4 \  
--termination CallingRegions="US",CidrAllowedList="10.24.34.0/23",Disabled=false
```

Salida:

```
{  
  "Termination": {  
    "CpsLimit": 0,  
    "CallingRegions": [  
      "US"  
    ],  
    "CidrAllowedList": [  
      "10.24.34.0/23"  
    ],  
    "Disabled": false  
  }  
}
```

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [PutVoiceConnectorTermination](#) de AWS CLI comandos.

## regenerate-security-token

En el siguiente ejemplo de código, se muestra cómo usar `regenerate-security-token`.

### AWS CLI

Para regenerar un token de seguridad

En el siguiente `regenerate-security-token` ejemplo, se regenera el token de seguridad del bot especificado.

```
aws chime regenerate-security-token \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --bot-id 123abcd4-5ef6-789g-0h12-34j56789012k
```

Salida:

```
{  
  "Bot": {  
    "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",  
    "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",  
    "DisplayName": "myBot (Bot)",  
    "BotType": "ChatBot",  
    "Disabled": false,  
    "CreatedTimestamp": "2019-09-09T18:05:56.749Z",  
    "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",  
    "BotEmail": "myBot-chimebot@example.com",  
    "SecurityToken": "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY"  
  }  
}
```

Para obtener más información, consulte [Autenticación de solicitudes de bots de chat](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte la Referencia [RegenerateSecurityToken](#) de AWS CLI comandos.

## reset-personal-pin

En el siguiente ejemplo de código, se muestra cómo usar `reset-personal-pin`.

### AWS CLI

Para restablecer la reunión personal de un usuario PIN

El siguiente `reset-personal-pin` ejemplo restablece la reunión PIN personal del usuario especificado.

```
aws chime reset-personal-pin \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --user-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Salida:

```
{  
  "User": {  
    "UserId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "PrimaryEmail": "mateo@example.com",  
    "DisplayName": "Mateo Jackson",  
    "LicenseType": "Pro",  
    "UserType": "PrivateUser",  
    "UserRegistrationStatus": "Registered",  
    "RegisteredOn": "2018-12-20T18:45:25.231Z",  
    "AlexaForBusinessMetadata": {  
      "IsAlexaForBusinessEnabled": False,  
      "AlexaForBusinessRoomArn": "null"  
    },  
    "PersonalPIN": "XXXXXXXXXX"  
  }  
}
```

Para obtener más información, consulte [Cambiar una reunión personal PINs](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [ResetPersonalPin](#) la Referencia de AWS CLI comandos.



## restore-phone-number

En el siguiente ejemplo de código, se muestra cómo usar `restore-phone-number`.

### AWS CLI

Para restaurar un número de teléfono

El siguiente `restore-phone-number` ejemplo restaura el número de teléfono especificado de la cola de eliminación.

```
aws chime restore-phone-number \  
  --phone-number-id "+12065550100"
```

Salida:

```
{  
  "PhoneNumber": {  
    "PhoneNumberId": "%2B12065550100",  
    "E164PhoneNumber": "+12065550100",  
    "Type": "Local",  
    "ProductType": "BusinessCalling",  
    "Status": "Unassigned",  
    "Capabilities": {  
      "InboundCall": true,  
      "OutboundCall": true,  
      "InboundSMS": true,  
      "OutboundSMS": true,  
      "InboundMMS": true,  
      "OutboundMMS": true  
    },  
    "Associations": [],  
    "CreatedTimestamp": "2019-08-09T21:35:21.445Z",  
    "UpdatedTimestamp": "2019-08-12T22:06:36.355Z"  
  }  
}
```

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [RestorePhoneNumber](#) la Referencia de AWS CLI comandos.

## search-available-phone-numbers

En el siguiente ejemplo de código, se muestra cómo usar `search-available-phone-numbers`.

### AWS CLI

Para buscar los números de teléfono disponibles

En el siguiente `search-available-phone-numbers` ejemplo, se buscan los números de teléfono disponibles por código de área.

```
aws chime search-available-phone-numbers \  
  --area-code "206"
```

Salida:

```
{  
  "E164PhoneNumbers": [  
    "+12065550100",  
    "+12065550101",  
    "+12065550102",  
    "+12065550103",  
    "+12065550104",  
    "+12065550105",  
    "+12065550106",  
    "+12065550107",  
    "+12065550108",  
    "+12065550109",  
  ]  
}
```

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [SearchAvailablePhoneNumbers](#) la Referencia de AWS CLI comandos.

## update-account-settings

En el siguiente ejemplo de código, se muestra cómo usar `update-account-settings`.

## AWS CLI

Para actualizar la configuración de su cuenta

El siguiente `update-account-settings` ejemplo deshabilita el control remoto de las pantallas compartidas para la cuenta de Amazon Chime especificada.

```
aws chime update-account-settings \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --account-settings DisableRemoteControl=true
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte la Referencia [UpdateAccountSettings](#) de AWS CLI comandos.

## update-account

En el siguiente ejemplo de código, se muestra cómo usar `update-account`.

## AWS CLI

Para actualizar una cuenta

En el siguiente `update-account` ejemplo, se actualiza el nombre de la cuenta especificada.

```
aws chime update-account \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --name MyAccountName
```

Salida:

```
{  
  "Account": {  
    "AwsAccountId": "111122223333",  
    "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
    "Name": "MyAccountName",  
    "AccountType": "Team",  
    "CreatedTimestamp": "2018-09-04T21:44:22.292Z",  
    "DefaultLicense": "Pro",  
    "SupportedLicenses": [  
      "Basic",
```

```
        "Pro"
      ],
      "SignInDelegateGroups": [
        {
          "GroupName": "myGroup"
        },
      ]
    }
  }
}
```

Para obtener más información, consulte [Cambiar el nombre de su cuenta](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [UpdateAccount](#) de AWS CLI comandos.

## update-bot

En el siguiente ejemplo de código, se muestra cómo usar `update-bot`.

### AWS CLI

Para actualizar un bot

En el siguiente `update-bot` ejemplo, se actualiza el estado del bot especificado para impedir su ejecución.

```
aws chime update-bot \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --bot-id 123abcd4-5ef6-789g-0h12-34j56789012k \
  --disabled
```

Salida:

```
{
  "Bot": {
    "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",
    "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",
    "DisplayName": "myBot (Bot)",
    "BotType": "ChatBot",
    "Disabled": true,
    "CreatedTimestamp": "2019-09-09T18:05:56.749Z",
```

```
"UpdatedTimestamp": "2019-09-09T18:05:56.749Z",
"BotEmail": "myBot-chimebot@example.com",
"SecurityToken": "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY"
}
}
```

Para obtener más información, consulte [Actualizar los bots de chat](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte [UpdateBot](#) la Referencia de AWS CLI comandos.

## update-global-settings

En el siguiente ejemplo de código, se muestra cómo usar `update-global-settings`.

### AWS CLI

Para actualizar la configuración global

En el siguiente `update-global-settings` ejemplo, se actualiza el bucket de S3 que se utiliza para almacenar los registros de detalles de llamadas de Amazon Chime Business Calling y Amazon Chime Voice Connectors asociados a la cuenta del administrador. AWS

```
aws chime update-global-settings \
  --business-calling CdrBucket="s3bucket" \
  --voice-connector CdrBucket="s3bucket"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión de la configuración global](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [UpdateGlobalSettings](#) la Referencia de AWS CLI comandos.

## update-phone-number-settings

En el siguiente ejemplo de código, se muestra cómo usar `update-phone-number-settings`.

### AWS CLI

Para actualizar el nombre de una llamada saliente

En el siguiente `update-phone-number-settings` ejemplo, se actualiza el nombre de llamada saliente predeterminado de la cuenta del AWS administrador.

```
aws chime update-phone-number-settings \  
  --calling-name "myName"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [UpdatePhoneNumberSettings](#) la Referencia de AWS CLI comandos.

## update-phone-number

En el siguiente ejemplo de código, se muestra cómo usar `update-phone-number`.

### AWS CLI

Ejemplo 1: Para actualizar el tipo de producto de un número de teléfono

En el siguiente `update-phone-number` ejemplo, se actualiza el tipo de producto del número de teléfono especificado.

```
aws chime update-phone-number \  
  --phone-number-id "+12065550100" \  
  --product-type "BusinessCalling"
```

Salida:

```
{  
  "PhoneNumber": {  
    "PhoneNumberId": "%2B12065550100",  
    "E164PhoneNumber": "+12065550100",  
    "Type": "Local",  
    "ProductType": "BusinessCalling",  
    "Status": "Unassigned",  
    "Capabilities": {  
      "InboundCall": true,  

```

```

        "OutboundCall": true,
        "InboundSMS": true,
        "OutboundSMS": true,
        "InboundMMS": true,
        "OutboundMMS": true
    },
    "Associations": [],
    "CallingName": "phonenumber1",
    "CreatedTimestamp": "2019-08-09T21:35:21.445Z",
    "UpdatedTimestamp": "2019-08-12T21:44:07.591Z"
}
}

```

Ejemplo 2: Para actualizar el nombre de la llamada saliente de un número de teléfono

En el siguiente `update-phone-number` ejemplo, se actualiza el nombre de la llamada saliente del número de teléfono especificado.

```

as chime update-phone-number : phone-number-id «+12065550100" --nombre de llamada
«phonenumber2"

```

Salida:

```

{
  "PhoneNumber": {
    "PhoneNumberId": "%2B12065550100",
    "E164PhoneNumber": "+12065550100",
    "Type": "Local",
    "ProductType": "BusinessCalling",
    "Status": "Unassigned",
    "Capabilities": {
      "InboundCall": true,
      "OutboundCall": true,
      "InboundSMS": true,
      "OutboundSMS": true,
      "InboundMMS": true,
      "OutboundMMS": true
    },
    "Associations": [],
    "CallingName": "phonenumber2",
    "CreatedTimestamp": "2019-08-09T21:35:21.445Z",
    "UpdatedTimestamp": "2019-08-12T21:44:07.591Z"
  }
}

```

```
}
```

Para obtener más información, consulte [Uso de números de teléfono](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [UpdatePhoneNumber](#) la Referencia de AWS CLI comandos.

## update-proxy-session

En el siguiente ejemplo de código, se muestra cómo usar `update-proxy-session`.

### AWS CLI

Para actualizar una sesión de proxy

En el siguiente `update-proxy-session` ejemplo, se actualizan las capacidades de la sesión de proxy.

```
aws chime update-proxy-session \  
  --voice-connector-id abcdef1ghij2klmno3pqr4 \  
  --proxy-session-id 123a4bc5-67d8-901e-2f3g-h4ghjk567891 \  
  --capabilities "Voice"
```

Salida:

```
{  
  "ProxySession": {  
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",  
    "ProxySessionId": "123a4bc5-67d8-901e-2f3g-h4ghjk567891",  
    "Status": "Open",  
    "ExpiryMinutes": 60,  
    "Capabilities": [  
      "Voice"  
    ],  
    "CreatedTimestamp": "2020-04-15T16:10:10.288Z",  
    "UpdatedTimestamp": "2020-04-15T16:10:10.288Z",  
    "Participants": [  
      {  
        "PhoneNumber": "+12065550100",  
        "ProxyPhoneNumber": "+19135550199"  
      },  
    ],  
  },  
}
```



```

    {
      "PhoneNumber": "+14015550101",
      "ProxyPhoneNumber": "+19135550199"
    }
  ]
}

```

Para obtener más información, consulte [Proxy Phone Sessions](#) en la Guía para desarrolladores de Amazon Chime.

- Para API obtener más información, consulte [UpdateProxySession](#) la Referencia de AWS CLI comandos.

## update-room-membership

En el siguiente ejemplo de código, se muestra cómo usar `update-room-membership`.

### AWS CLI

Para actualizar la membresía de una sala

En el siguiente `update-room-membership` ejemplo, se modifica el rol del miembro de la sala de chat especificado a `Administrator`.

```

aws chime update-room-membership \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \
  --member-id 1ab2345c-67de-8901-f23g-45h678901j2k \
  --role Administrator

```

Salida:

```

{
  "RoomMembership": {
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
    "Member": {
      "MemberId": "1ab2345c-67de-8901-f23g-45h678901j2k",
      "MemberType": "User",
      "Email": "sofiamartinez@example.com",
      "FullName": "Sofia Martinez",
    }
  }
}

```

```

        "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45"
    },
    "Role": "Administrator",
    "InvitedBy": "arn:aws:iam::111122223333:user/admin",
    "UpdatedTimestamp": "2019-12-02T22:40:22.931Z"
}
}

```

Para obtener más información, consulte [Creación de una sala de chat](#) en la Guía del usuario de Amazon Chime.

- Para API obtener más información, consulte [UpdateRoomMembership](#) la Referencia de AWS CLI comandos.

## update-room

En el siguiente ejemplo de código, se muestra cómo usar update-room.

### AWS CLI

Para actualizar una sala de chat

En el siguiente update-room ejemplo, se modifica el nombre de la sala de chat especificada.

```

aws chime update-room \
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
  --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \
  --name teamRoom

```

Salida:

```

{
  "Room": {
    "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
    "Name": "teamRoom",
    "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",
    "CreatedBy": "arn:aws:iam::111122223333:user/alejandro",
    "CreatedTimestamp": "2019-12-02T22:29:31.549Z",
    "UpdatedTimestamp": "2019-12-02T22:33:19.310Z"
  }
}

```

Para obtener más información, consulte [Creación de una sala de chat](#) en la Guía del usuario de Amazon Chime.

- Para API obtener más información, consulte [UpdateRoom](#) la Referencia de AWS CLI comandos.

## update-user-settings

En el siguiente ejemplo de código, se muestra cómo usar `update-user-settings`.

### AWS CLI

Para actualizar la configuración del usuario

El siguiente `update-user-settings` ejemplo permite al usuario especificado realizar llamadas entrantes y salientes y enviar y recibir SMS mensajes.

```
aws chime update-user-settings \  
  --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \  
  --user-id 1ab2345c-67de-8901-f23g-45h678901j2k \  
  --user-settings "Telephony={InboundCalling=true,OutboundCalling=true,SMS=true}"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar los números de teléfono de los usuarios](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte [UpdateUserSettings](#) la Referencia de AWS CLI comandos.

## update-user

En el siguiente ejemplo de código, se muestra cómo usar `update-user`.

### AWS CLI

Para actualizar los detalles del usuario

En este ejemplo, se actualizan los detalles especificados para el usuario especificado.

Comando:

```
aws chime update-user \  
  --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --user-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE \  
  --license-type "Basic"
```

Salida:

```
{  
  "User": {  
    "UserId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE"  
  }  
}
```

- Para API obtener más información, consulte [UpdateUser](#) la Referencia de AWS CLI comandos.

## update-voice-connector-group

En el siguiente ejemplo de código, se muestra cómo usar `update-voice-connector-group`.

AWS CLI

Para actualizar los detalles de un grupo de conectores de voz de Amazon Chime

En el siguiente `update-voice-connector-group` ejemplo, se actualizan los detalles del grupo de conectores de voz de Amazon Chime especificado.

```
aws chime update-voice-connector-group \  
  --voice-connector-group-id 123a456b-c7d8-90e1-fg23-4h567jkl8901 \  
  --name "newGroupName" \  
  --voice-connector-items VoiceConnectorId=abcdef1ghij2klmno3pqr4,Priority=1
```

Salida:

```
{  
  "VoiceConnectorGroup": {  
    "VoiceConnectorGroupId": "123a456b-c7d8-90e1-fg23-4h567jkl8901",  
    "Name": "newGroupName",  
    "VoiceConnectorItems": [  
      {  
        "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",
```

```

        "Priority": 1
      }
    ],
    "CreatedTimestamp": "2019-09-18T16:38:34.734Z",
    "UpdatedTimestamp": "2019-10-28T19:00:57.081Z"
  }
}

```

Para obtener más información, consulte [Uso de grupos de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [UpdateVoiceConnectorGroup](#) de AWS CLI comandos.

## update-voice-connector

En el siguiente ejemplo de código, se muestra cómo usar `update-voice-connector`.

### AWS CLI

Para actualizar los detalles de un conector de voz de Amazon Chime

En el siguiente `update-voice-connector` ejemplo, se actualiza el nombre del conector de voz Amazon Chime especificado.

```

aws chime update-voice-connector \
  --voice-connector-id abcdef1ghij2klmno3pqr4 \
  --name newName \
  --require-encryption

```

Salida:

```

{
  "VoiceConnector": {
    "VoiceConnectorId": "abcdef1ghij2klmno3pqr4",
    "AwsRegion": "us-west-2",
    "Name": "newName",
    "OutboundHostName": "abcdef1ghij2klmno3pqr4.voiceconnector.chime.aws",
    "RequireEncryption": true,
    "CreatedTimestamp": "2019-09-18T20:34:01.352Z",
    "UpdatedTimestamp": "2019-09-18T20:40:52.895Z"
  }
}

```

```
}
```

Para obtener más información, consulte [Uso de conectores de voz de Amazon Chime](#) en la Guía de administración de Amazon Chime.

- Para API obtener más información, consulte la Referencia [UpdateVoiceConnector](#) de AWS CLI comandos.

## API Ejemplos de Cloud Control con AWS CLI

Los siguientes ejemplos de código te muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Cloud Control API.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puedes encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-resource**

En el siguiente ejemplo de código, se muestra cómo usar `create-resource`.

AWS CLI

Para crear un recurso

En el siguiente `create-resource` ejemplo, se crea un recurso `AWS::Kinesis::Stream`, denominado `ResourceExample`, con un período de retención de 168 horas y un recuento de fragmentos de tres.

```
aws cloudcontrol create-resource \  
  --type-name AWS::Kinesis::Stream \  
  --name ResourceExample
```

```
--desired-state "{\"Name\": \"ResourceExample\", \"RetentionPeriodHours\":168,
\"ShardCount\":3}"
```

Salida:

```
{
  "ProgressEvent": {
    "EventTime": 1632506656.706,
    "TypeName": "AWS::Kinesis::Stream",
    "OperationStatus": "IN_PROGRESS",
    "Operation": "CREATE",
    "Identifier": "ResourceExample",
    "RequestToken": "20999d87-e304-4725-ad84-832dcbfd7fc5"
  }
}
```

Para obtener más información, consulta [Cómo crear un recurso](#) en la Guía del usuario de Cloud Control. API

- Para API obtener más información, consulta [CreateResource](#) la Referencia de AWS CLI comandos.

## delete-resource

En el siguiente ejemplo de código, se muestra cómo usar delete-resource.

AWS CLI

Para eliminar un recurso

En el siguiente delete-resource ejemplo, se elimina un recurso AWS::Kinesis::Stream con el identificador ResourceExample de su cuenta. AWS

```
aws cloudcontrol delete-resource \
  --type-name AWS::Kinesis::Stream \
  --identifier ResourceExample
```

Salida:

```
{
  "ProgressEvent": {
```

```

    "TypeName": "AWS::Kinesis::Stream",
    "Identifier": "ResourceExample",
    "RequestToken": "e48f26ff-d0f9-4ab8-a878-120db1edf111",
    "Operation": "DELETE",
    "OperationStatus": "IN_PROGRESS",
    "EventTime": 1632950300.14
  }
}

```

Para obtener más información, consulta [Eliminar un recurso](#) en la Guía del usuario de Cloud Control. API

- Para API obtener más información, consulta [DeleteResource](#) la Referencia de AWS CLI comandos.

## get-resource-request-status

En el siguiente ejemplo de código, se muestra cómo usar `get-resource-request-status`.

### AWS CLI

Para obtener la información de estado de una solicitud de recursos

El siguiente `get-resource-request-status` ejemplo devuelve información de estado sobre la solicitud de recursos especificada.

```

aws cloudcontrol get-resource-request-status \
  --request-token "e1a6b86e-46bd-41ac-bfba-001234567890"

```

Salida:

```

{
  "ProgressEvent": {
    "TypeName": "AWS::Kinesis::Stream",
    "Identifier": "Demo",
    "RequestToken": "e1a6b86e-46bd-41ac-bfba-001234567890",
    "Operation": "CREATE",
    "OperationStatus": "FAILED",
    "EventTime": 1632950268.481,
    "StatusMessage": "Resource of type 'AWS::Kinesis::Stream' with identifier
'Demo' already exists.",
    "ErrorCode": "AlreadyExists"
  }
}

```



```
}
}
```

Para obtener más información, consulta [Administrar las solicitudes de operaciones de recursos](#) en la Guía del API usuario de Cloud Control.

- Para API obtener más información, consulta [GetResourceRequestStatus](#) la Referencia de AWS CLI comandos.

## get-resource

En el siguiente ejemplo de código, se muestra cómo usar `get-resource`.

### AWS CLI

Para obtener el estado actual de un recurso

El siguiente `get-resource` ejemplo devuelve el estado actual del recurso AWS: :Kinesis: :Stream denominado. ResourceExample

```
aws cloudcontrol get-resource \
  --type-name AWS::Kinesis::Stream \
  --identifier ResourceExample
```

Salida:

```
{
  "TypeName": "AWS::Kinesis::Stream",
  "ResourceDescription": {
    "Identifier": "ResourceExample",
    "Properties": "{\"Arn\":\"arn:aws:kinesis:us-west-2:099908667365:stream/ResourceExample\", \"RetentionPeriodHours\":168, \"Name\":\"ResourceExample\", \"ShardCount\":3}"
  }
}
```

Para obtener más información, consulta Cómo [leer el estado actual de un recurso](#) en la Guía del usuario de Cloud Control API.

- Para API obtener más información, consulta [GetResource](#) la Referencia de AWS CLI comandos.

## list-resource-requests

En el siguiente ejemplo de código, se muestra cómo usar `list-resource-requests`.

### AWS CLI

Para enumerar las solicitudes de operación de recursos activas

En el siguiente `list-resource-requests` ejemplo, se enumeran las solicitudes de recursos CREATE y UPDATE las operaciones que han fallado en su AWS cuenta.

```
aws cloudcontrol list-resource-requests \
  --resource-request-status-filter Operations=CREATE,OperationStatuses=FAILED
```

Salida:

```
{
  "ResourceRequestStatusSummaries": [
    {
      "TypeName": "AWS::Kinesis::Stream",
      "Identifier": "Demo",
      "RequestToken": "e1a6b86e-46bd-41ac-bfba-633abcdfdbd7",
      "Operation": "CREATE",
      "OperationStatus": "FAILED",
      "EventTime": 1632950268.481,
      "StatusMessage": "Resource of type 'AWS::Kinesis::Stream' with
identifier 'Demo' already exists.",
      "ErrorCode": "AlreadyExists"
    }
  ]
}
```

Para obtener más información, consulta [Administrar las solicitudes de operaciones de recursos](#) en la Guía del API usuario de Cloud Control.

- Para API obtener más información, consulta [ListResourceRequests](#) la Referencia de AWS CLI comandos.

## list-resources

En el siguiente ejemplo de código, se muestra cómo usar `list-resources`.

## AWS CLI

Para enumerar los recursos de un tipo determinado

El siguiente `list-resources` ejemplo muestra los recursos `AWS::Kinesis::Stream` aprovisionados en su cuenta. AWS

```
aws cloudcontrol list-resources \  
  --type-name AWS::Kinesis::Stream
```

Salida:

```
{  
  "TypeName": "AWS::Kinesis::Stream",  
  "ResourceDescriptions": [  
    {  
      "Identifier": "MyKinesisStream",  
      "Properties": "{\"Name\":\"MyKinesisStream\"}"  
    },  
    {  
      "Identifier": "AnotherStream",  
      "Properties": "{\"Name\":\"AnotherStream\"}"  
    }  
  ]  
}
```

Para obtener más información, consulta Cómo [descubrir recursos](#) en la Guía del usuario de Cloud Control. API

- Para API obtener más información, consulta [ListResources](#) la Referencia de AWS CLI comandos.

## update-resource

En el siguiente ejemplo de código, se muestra cómo usar `update-resource`.

## AWS CLI

Para actualizar las propiedades de un recurso existente

En el siguiente `update-resource` ejemplo, se actualiza la política de retención de un `LogGroup` recurso `AWS::Logs::` denominado `ExampleLogGroup` 90 días.

```
aws cloudcontrol update-resource \  
  --type-name AWS::Logs::LogGroup \  
  --identifier ExampleLogGroup \  
  --patch-document "[{"op":"replace","path":"/RetentionInDays","value":90}]"
```

Salida:

```
{  
  "ProgressEvent": {  
    "EventTime": "2021-08-09T18:17:15.219Z",  
    "TypeName": "AWS::Logs::LogGroup",  
    "OperationStatus": "IN_PROGRESS",  
    "Operation": "UPDATE",  
    "Identifier": "ExampleLogGroup",  
    "RequestToken": "5f40c577-3534-4b20-9599-0b0123456789"  
  }  
}
```

Para obtener más información, consulta Cómo [actualizar un recurso](#) en la Guía del API usuario de Cloud Control.

- Para API obtener más información, consulta [UpdateResource](#) la Referencia de AWS CLI comandos.

## AWS Cloud Map ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Cloud Map.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **create-private-dns-namespace**

En el siguiente ejemplo de código, se muestra cómo usar `create-private-dns-namespace`.

#### AWS CLI

Para crear un espacio de DNS nombres privado

En el siguiente `create-private-dns-namespace` ejemplo, se crea un espacio de nombres privadoDNS.

```
aws servicediscovery create-private-dns-namespace \  
  --name example.com \  
  --vpc vpc-1c56417b
```

Salida:

```
{  
  "OperationId": "gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9302yzd"  
}
```

Para confirmar que la operación se realizó correctamente, puede ejecutar `get-operation`. Para obtener más información, consulte [get-operation](#).

Para obtener más información, consulta [Cómo crear espacios de nombres](#) en la Guía para desarrolladores de AWS Cloud Map.

- Para API obtener más información, consulta la Referencia [CreatePrivateDnsNamespacede](#) AWS CLI comandos.

### **create-service**

En el siguiente ejemplo de código, se muestra cómo usar `create-service`.

#### AWS CLI

Para crear un servicio

En el siguiente `create-service` ejemplo, se crea un servicio.

```
aws servicediscovery create-service \  
  --name example.com \  
  --vpc vpc-1c56417b
```

```
--name myservice \  
--namespace-id ns-ylexjili4cdxy3xm \  
--dns-config "NamespaceId=ns-  
ylexjili4cdxy3xm,RoutingPolicy=MULTIVALUE,DnsRecords=[{Type=A,TTL=60}]"
```

Salida:

```
{  
  "Service": {  
    "Id": "srv-p5zdwlg5uvvzjita",  
    "Arn": "arn:aws:servicediscovery:us-west-2:803642222207:service/srv-  
p5zdwlg5uvvzjita",  
    "Name": "myservice",  
    "NamespaceId": "ns-ylexjili4cdxy3xm",  
    "DnsConfig": {  
      "NamespaceId": "ns-ylexjili4cdxy3xm",  
      "RoutingPolicy": "MULTIVALUE",  
      "DnsRecords": [  
        {  
          "Type": "A",  
          "TTL": 60  
        }  
      ]  
    },  
    "CreateDate": 1587081768.334,  
    "CreatorRequestId": "567c1193-6b00-4308-bd57-ad38a8822d25"  
  }  
}
```

Para obtener más información, consulte [Creación de servicios](#) en la Guía para desarrolladores de AWS Cloud Map.

- Para API obtener más información, consulte [CreateService](#) la Referencia de AWS CLI comandos.

## delete-namespace

En el siguiente ejemplo de código, se muestra cómo usar delete-namespace.

### AWS CLI

Para eliminar un espacio de nombres

En el siguiente delete-namespace ejemplo, se elimina un espacio de nombres.

```
aws servicediscovery delete-namespace \  
  --id ns-ylexjili4cdxy3xm
```

Salida:

```
{  
  "OperationId": "gv4g5meo7ndmeh4fqskygvk23d2fijwa-k98y6drk"  
}
```

Para confirmar que la operación se realizó correctamente, puede ejecutar `get-operation`. Para obtener más información, consulte [get-operation](#).

Para obtener más información, consulta [Eliminar espacios de nombres](#) en la Guía para desarrolladores de AWS Cloud Map.

- Para API obtener más información, consulta la Referencia [DeleteNamespace](#) de AWS CLI comandos.

## delete-service

En el siguiente ejemplo de código, se muestra cómo usar delete-service.

AWS CLI

Eliminación de un servicio

En el siguiente delete-service ejemplo, se elimina un servicio.

```
aws servicediscovery delete-service \  
  --id srv-p5zdwlg5uvvzjita
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Eliminar servicios](#) en la Guía para desarrolladores de AWS Cloud Map.

- Para API obtener más información, consulte [DeleteService](#) la Referencia de AWS CLI comandos.

## deregister-instance

En el siguiente ejemplo de código, se muestra cómo usar `deregister-instance`.

### AWS CLI

Para anular el registro de una instancia de servicio

En el siguiente `deregister-instance` ejemplo se anula el registro de una instancia de servicio.

```
aws servicediscovery deregister-instance \  
  --service-id srv-p5zdwlg5uvvzjita \  
  --instance-id myservice-53
```

Salida:

```
{  
  "OperationId": "4yejorelbukcjzpnr6t1mrghsjwpngf4-k98rnaiq"  
}
```

Para confirmar que la operación se realizó correctamente, puede ejecutar `get-operation`. Para obtener más información, consulte [get-operation](#).

Para obtener más información, consulta Cómo [anular el registro de instancias de servicio](#) en la Guía para desarrolladores de AWS Cloud Map.

- Para API obtener más información, consulta la Referencia [DeregisterInstance](#) de AWS CLI comandos.

## discover-instances

En el siguiente ejemplo de código, se muestra cómo usar `discover-instances`.

### AWS CLI

Para descubrir instancias registradas

En el siguiente `discover-instances` ejemplo, se descubren instancias registradas.

```
aws servicediscovery discover-instances \  
  --service-id srv-p5zdwlg5uvvzjita
```



```
--namespace-name example.com \  
--service-name myservice \  
--max-results 10 \  
--health-status ALL
```

Salida:

```
{  
  "Instances": [  
    {  
      "InstanceId": "myservice-53",  
      "NamespaceName": "example.com",  
      "ServiceName": "myservice",  
      "HealthStatus": "UNKNOWN",  
      "Attributes": {  
        "AWS_INSTANCE_IPV4": "172.2.1.3",  
        "AWS_INSTANCE_PORT": "808"  
      }  
    }  
  ]  
}
```

- Para API obtener más información, consulte [DiscoverInstances](#) la Referencia de AWS CLI comandos.

## get-operation

En el siguiente ejemplo de código, se muestra cómo usar `get-operation`.

AWS CLI

Para obtener el resultado de una operación

En el `get-operation` ejemplo siguiente se obtiene el resultado de una operación.

```
aws servicediscovery get-operation \  
--operation-id gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9302yzd
```

Salida:

```
{
```

```
"Operation": {
  "Id": "gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9302yzd",
  "Type": "CREATE_NAMESPACE",
  "Status": "SUCCESS",
  "CreateDate": 1587055860.121,
  "UpdateDate": 1587055900.469,
  "Targets": {
    "NAMESPACE": "ns-ylexjili4cdxy3xm"
  }
}
```

- Para API obtener más información, consulte [GetOperation](#) la Referencia de AWS CLI comandos.

## list-instances

En el siguiente ejemplo de código, se muestra cómo usar `list-instances`.

### AWS CLI

Para enumerar las instancias de servicio

En el siguiente `list-instances` ejemplo, se enumeran las instancias de servicio.

```
aws servicediscovery list-instances \
  --service-id srv-qzpwvt2tfqcegapy
```

Salida:

```
{
  "Instances": [
    {
      "Id": "i-06bdabbae60f65a4e",
      "Attributes": {
        "AWS_INSTANCE_IPV4": "172.2.1.3",
        "AWS_INSTANCE_PORT": "808"
      }
    }
  ]
}
```

Para obtener más información, consulta [Cómo ver una lista de instancias de servicio](#) en la Guía para desarrolladores de AWS Cloud Map.

- Para API obtener más información, consulte [ListInstances](#) la Referencia de AWS CLI comandos.

## list-namespaces

En el siguiente ejemplo de código, se muestra cómo usar `list-namespaces`.

### AWS CLI

Para enumerar los espacios de nombres

En el siguiente `list-namespaces` ejemplo, se enumeran los espacios de nombres.

```
aws servicediscovery list-namespaces
```

Salida:

```
{
  "Namespaces": [
    {
      "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-a3ccy2e7e3a7rile",
      "CreateDate": 1585354387.357,
      "Id": "ns-a3ccy2e7e3a7rile",
      "Name": "local",
      "Properties": {
        "DnsProperties": {
          "HostedZoneId": "Z06752353VBUDTC32S84S"
        },
        "HttpProperties": {
          "HttpName": "local"
        }
      },
      "Type": "DNS_PRIVATE"
    },
    {
      "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-pocfyjtrismwtvcxx",
      "CreateDate": 1586468974.698,
      "Description": "My second namespace",
      "Id": "ns-pocfyjtrismwtvcxx",
```

```

    "Name": "My-second-namespace",
    "Properties": {
      "DnsProperties": {},
      "HttpProperties": {
        "HttpName": "My-second-namespace"
      }
    },
    "Type": "HTTP"
  },
  {
    "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-
ylexjili4cdxy3xm",
    "CreateDate": 1587055896.798,
    "Id": "ns-ylexjili4cdxy3xm",
    "Name": "example.com",
    "Properties": {
      "DnsProperties": {
        "HostedZoneId": "Z09983722P0QME1B3KC8I"
      },
      "HttpProperties": {
        "HttpName": "example.com"
      }
    },
    "Type": "DNS_PRIVATE"
  }
]
}

```

Para obtener más información, consulta Cómo [ver una lista de espacios de nombres](#) en la Guía para desarrolladores de AWS Cloud Map.

- Para API obtener más información, consulta la Referencia [ListNamespaces](#) de AWS CLI comandos.

## list-services

En el siguiente ejemplo de código, se muestra cómo usar `list-services`.

### AWS CLI

Para enumerar los servicios

En el siguiente `list-services` ejemplo, se enumeran los servicios.

```
aws servicediscovery list-services
```

Salida:

```
{
  "Services": [
    {
      "Id": "srv-p5zdwlg5uvvzjita",
      "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
p5zdwlg5uvvzjita",
      "Name": "myservice",
      "DnsConfig": {
        "RoutingPolicy": "MULTIVALUE",
        "DnsRecords": [
          {
            "Type": "A",
            "TTL": 60
          }
        ]
      },
      "CreateDate": 1587081768.334
    }
  ]
}
```

Para obtener más información, consulta [Cómo ver una lista de servicios](#) en la Guía para desarrolladores de AWS Cloud Map.

- Para API obtener más información, consulte [ListServices](#) la Referencia de AWS CLI comandos.

## register-instance

En el siguiente ejemplo de código, se muestra cómo usar `register-instance`.

### AWS CLI

Para registrar una instancia de servicio

En el siguiente `register-instance` ejemplo, se registra una instancia de servicio.

```
aws servicediscovery register-instance \
```

```
--service-id srv-p5zdwlg5uvvzjita \  
--instance-id myservice-53 \  
--attributes=AWS_INSTANCE_IPV4=172.2.1.3,AWS_INSTANCE_PORT=808
```

Salida:

```
{  
  "OperationId": "4yejorelbukcjzpnr6t1mrghsjwpngf4-k95yg2u7"  
}
```

Para confirmar que la operación se realizó correctamente, puede ejecutar `target-operation`. Para obtener más información, consulte [get-operation](#).

Para obtener más información, consulta [Registrar instancias](#) en la Guía para desarrolladores de AWS Cloud Map.

- Para API obtener más información, consulte [RegisterInstance](#) la Referencia de AWS CLI comandos.

## AWS Cloud9 ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Cloud9.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-environment-ec2**

En el siguiente ejemplo de código, se muestra cómo usar `create-environment-ec2`.

## AWS CLI

Para crear un entorno de AWS desarrollo Cloud9 EC2

En el siguiente `create-environment-ec2` ejemplo, se crea un entorno de desarrollo de AWS Cloud9 con la configuración especificada, se lanza una instancia de Amazon Elastic Compute Cloud EC2 (Amazon) y, a continuación, se conecta desde la instancia al entorno.

```
aws cloud9 create-environment-ec2 \  
  --name my-demo-env \  
  --description "My demonstration development environment." \  
  --instance-type t2.micro --image-id amazonlinux-2023-x86_64 \  
  --subnet-id subnet-1fab8aEX \  
  --automatic-stop-time-minutes 60 \  
  --owner-arn arn:aws:iam::123456789012:user/MyDemoUser
```

Salida:

```
{  
  "environmentId": "8a34f51ce1e04a08882f1e811bd706EX"  
}
```

Para obtener más información, consulte la Guía del usuario sobre cómo [crear un EC2 entorno](#) en AWS Cloud9.

- Para API obtener más información, consulte la sección [CreateEnvironmentEc2](#) en la Referencia de AWS CLI comandos.

## `create-environment-membership`

En el siguiente ejemplo de código, se muestra cómo usar `create-environment-membership`.

## AWS CLI

Para añadir un miembro del entorno a un entorno de desarrollo de AWS Cloud9

En este ejemplo, se agrega el miembro del entorno especificado al entorno de desarrollo de AWS Cloud9 especificado.

Comando:

```
aws cloud9 create-environment-membership --environment-id 8a34f51ce1e04a08882f1e811bd706EX --user-arn arn:aws:iam::123456789012:user/AnotherDemoUser --permissions read-write
```

Salida:

```
{
  "membership": {
    "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
    "userId": "AIDAJ3LOROMOUXTBSU6EX",
    "userArn": "arn:aws:iam::123456789012:user/AnotherDemoUser",
    "permissions": "read-write"
  }
}
```

- Para API obtener más información, consulte [CreateEnvironmentMembership](#) la Referencia de AWS CLI comandos.

## delete-environment-membership

En el siguiente ejemplo de código, se muestra cómo usar `delete-environment-membership`.

### AWS CLI

Para eliminar un miembro del entorno de un entorno de desarrollo de AWS Cloud9

En este ejemplo, se elimina el miembro del entorno especificado del entorno de desarrollo AWS Cloud9 especificado.

Comando:

```
aws cloud9 delete-environment-membership --environment-id 8a34f51ce1e04a08882f1e811bd706EX --user-arn arn:aws:iam::123456789012:user/AnotherDemoUser
```

Salida:

```
None.
```

- Para API obtener más información, consulte la Referencia [DeleteEnvironmentMembership](#) de AWS CLI comandos.



## delete-environment

En el siguiente ejemplo de código, se muestra cómo usar `delete-environment`.

### AWS CLI

Para eliminar un entorno de AWS desarrollo de Cloud9

En este ejemplo, se elimina el entorno de desarrollo AWS Cloud9 especificado. Si una EC2 instancia de Amazon está conectada al entorno, también termina la instancia.

Comando:

```
aws cloud9 delete-environment --environment-id 8a34f51ce1e04a08882f1e811bd706EX
```

Salida:

```
None.
```

- Para API obtener más información, consulte [DeleteEnvironment](#) la Referencia de AWS CLI comandos.

## describe-environment-memberships

En el siguiente ejemplo de código, se muestra cómo usar `describe-environment-memberships`.

### AWS CLI

Para obtener información sobre los miembros del entorno de un entorno de desarrollo de AWS Cloud9

En este ejemplo, se obtiene información sobre los miembros del entorno de desarrollo AWS Cloud9 especificado.

Comando:

```
aws cloud9 describe-environment-memberships --environment-id 8a34f51ce1e04a08882f1e811bd706EX
```

Salida:

```
{
  "memberships": [
    {
      "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
      "userId": "AIDAJ3LOROMOXTBSU6EX",
      "userArn": "arn:aws:iam::123456789012:user/AnotherDemoUser",
      "permissions": "read-write"
    },
    {
      "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
      "userId": "AIDAJNUEDQAQWFELJDLEX",
      "userArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "permissions": "owner"
    }
  ]
}
```

Para obtener información sobre el propietario de un entorno de desarrollo de AWS Cloud9

En este ejemplo se obtiene información sobre el propietario del entorno de desarrollo AWS Cloud9 especificado.

Comando:

```
aws cloud9 describe-environment-memberships --environment-id 8a34f51ce1e04a08882f1e811bd706EX --permissions owner
```

Salida:

```
{
  "memberships": [
    {
      "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
      "userId": "AIDAJNUEDQAQWFELJDLEX",
      "userArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "permissions": "owner"
    }
  ]
}
```

Para obtener información sobre un miembro del entorno para varios entornos de desarrollo de AWS Cloud9

En este ejemplo, se obtiene información sobre el miembro del entorno especificado para varios entornos de desarrollo de AWS Cloud9.

Comando:

```
aws cloud9 describe-environment-memberships --user-arn arn:aws:iam::123456789012:user/MyDemoUser
```

Salida:

```
{
  "memberships": [
    {
      "environmentId": "10a75714bd494714929e7f5ec4125aEX",
      "lastAccess": 1516213427.0,
      "userId": "AIDAJNUEDQAQWFELJDLEX",
      "userArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "permissions": "owner"
    },
    {
      "environmentId": "1980b80e5f584920801c09086667f0EX",
      "lastAccess": 1516144884.0,
      "userId": "AIDAJNUEDQAQWFELJDLEX",
      "userArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "permissions": "owner"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeEnvironmentMemberships](#) la Referencia de AWS CLI comandos.

## **describe-environment-status**

En el siguiente ejemplo de código, se muestra cómo usar `describe-environment-status`.

AWS CLI

Para obtener información sobre el estado de un entorno de desarrollo de AWS Cloud9

En este ejemplo se obtiene la información de estado del entorno de desarrollo AWS Cloud9 especificado.

Comando:

```
aws cloud9 describe-environment-status --environment-id 685f892f431b45c2b28cb69eadcdb0EX
```

Salida:

```
{
  "status": "ready",
  "message": "Environment is ready to use"
}
```

- Para API obtener más información, consulte [DescribeEnvironmentStatus](#) la Referencia de AWS CLI comandos.

## describe-environments

En el siguiente ejemplo de código, se muestra cómo usar describe-environments.

AWS CLI

Para obtener información sobre los entornos de desarrollo de AWS Cloud9

En este ejemplo se obtiene información sobre los entornos de desarrollo de AWS Cloud9 especificados.

Comando:

```
aws cloud9 describe-environments --environment-ids 685f892f431b45c2b28cb69eadcdb0EX 349c86d4579e4e7298d500ff57a6b2EX
```

Salida:

```
{
  "environments": [
    {
      "id": "685f892f431b45c2b28cb69eadcdb0EX",

```

```
    "name": "my-demo-ec2-env",
    "description": "Created from CodeStar.",
    "type": "ec2",
    "arn": "arn:aws:cloud9:us-
east-1:123456789012:environment:685f892f431b45c2b28cb69eadcdb0EX",
    "ownerArn": "arn:aws:iam::123456789012:user/MyDemoUser",
    "lifecycle": {
      "status": "CREATED"
    }
  },
  {
    "id": "349c86d4579e4e7298d500ff57a6b2EX",
    "name": "my-demo-ssh-env",
    "description": "",
    "type": "ssh",
    "arn": "arn:aws:cloud9:us-
east-1:123456789012:environment:349c86d4579e4e7298d500ff57a6b2EX",
    "ownerArn": "arn:aws:iam::123456789012:user/MyDemoUser",
    "lifecycle": {
      "status": "CREATED"
    }
  }
]
}
```

- Para API obtener más información, consulte [DescribeEnvironments](#) la Referencia de AWS CLI comandos.

## list-environments

En el siguiente ejemplo de código, se muestra cómo usar `list-environments`.

### AWS CLI

Para obtener una lista de los identificadores del entorno de desarrollo de AWS Cloud9 disponibles

En este ejemplo se obtiene una lista de los identificadores del entorno de desarrollo de AWS Cloud9 disponibles.

Comando:

```
aws cloud9 list-environments
```

Salida:

```
{
  "environmentIds": [
    "685f892f431b45c2b28cb69eadcdb0EX",
    "1980b80e5f584920801c09086667f0EX"
  ]
}
```

- Para API obtener más información, consulte la Referencia [ListEnvironments](#) de AWS CLI comandos.

## update-environment-membership

En el siguiente ejemplo de código, se muestra cómo usar `update-environment-membership`.

AWS CLI

Para cambiar la configuración de un miembro del entorno existente para un entorno de desarrollo de AWS Cloud9

Este ejemplo cambia la configuración del miembro del entorno existente especificado para el entorno de desarrollo AWS Cloud9 especificado.

Comando:

```
aws cloud9 update-environment-membership --environment-
id 8a34f51ce1e04a08882f1e811bd706EX --user-arn arn:aws:iam::123456789012:user/
AnotherDemoUser --permissions read-only
```

Salida:

```
{
  "membership": {
    "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
    "userId": "AIDAJ3LOROMOUCTBSUGEX",
    "userArn": "arn:aws:iam::123456789012:user/AnotherDemoUser",
    "permissions": "read-only"
  }
}
```

- Para API obtener más información, consulte [UpdateEnvironmentMembership](#) la Referencia de AWS CLI comandos.

## update-environment

En el siguiente ejemplo de código, se muestra cómo usar `update-environment`.

### AWS CLI

Para cambiar la configuración de un entorno de desarrollo de AWS Cloud9 existente

Este ejemplo cambia la configuración especificada del entorno de desarrollo AWS Cloud9 existente especificado.

Comando:

```
aws cloud9 update-environment --environment-id 8a34f51ce1e04a08882f1e811bd706EX
--name my-changed-demo-env --description "My changed demonstration development
environment."
```

Salida:

```
None.
```

- Para API obtener más información, consulte [UpdateEnvironment](#) la Referencia de AWS CLI comandos.

## AWS CloudFormation ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS CloudFormation.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **activate-type**

En el siguiente ejemplo de código, se muestra cómo usar `activate-type`.

#### AWS CLI

Para activar un tipo

El siguiente `activate-type` ejemplo activa una extensión pública de terceros y la pone a disposición para su uso en plantillas de pila.

```
aws cloudformation activate-type \  
  --region us-west-2 \  
  --type RESOURCE \  
  --type-name Example::Test::1234567890abcdef0 \  
  --type-name-alias Example::Test::Alias
```

Salida:

```
{  
  "Arn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/Example-  
Test-Alias"  
}
```

Para obtener más información, consulte [Uso del AWS CloudFormation registro](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [ActivateType](#) la Referencia de AWS CLI comandos.

### **batch-describe-type-configurations**

En el siguiente ejemplo de código, se muestra cómo usar `batch-describe-type-configurations`.

#### AWS CLI

Para describir por lotes una configuración de tipos



El siguiente `batch-describe-type-configurations` ejemplo configura los datos del tipo.

```
aws cloudformation batch-describe-type-configurations \
  --region us-west-2 \
  --type-configuration-identifiers TypeArn="arn:aws:cloudformation:us-
west-2:123456789012:type/resource/Example-Test-
Type,TypeConfigurationAlias=MyConfiguration"
```

Salida:

```
{
  "Errors": [],
  "UnprocessedTypeConfigurations": [],
  "TypeConfigurations": [
    {
      "Arn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/
Example-Test-Type",
      "Alias": "MyConfiguration",
      "Configuration": "{\n      \"Example\": {\n          \"ApiKey\":
\"examplekey\", \n          \"ApplicationKey\": \"examplekey1\", \n
      \"ApiURL\": \"exampleurl\" \n      } \n    }",
      "LastUpdated": "2021-10-01T15:25:46.210000+00:00",
      "TypeArn": "arn:aws:cloudformation:us-east-1:123456789012:type/resource/
Example-Test-Type"
    }
  ]
}
```

Para obtener más información, consulte [Uso del AWS CloudFormation registro](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [BatchDescribeTypeConfigurations](#) la Referencia de AWS CLI comandos.

## cancel-update-stack

En el siguiente ejemplo de código, se muestra cómo usar `cancel-update-stack`.

### AWS CLI

Cancelación de una actualización de pila en progreso

El comando `cancel-update-stack` a continuación cancela una actualización de pila en la pila `myteststack`:

```
aws cloudformation cancel-update-stack --stack-name myteststack
```

- Para API obtener más información, consulte [CancelUpdateStack](#) la Referencia de AWS CLI comandos.

## **continue-update-rollback**

En el siguiente ejemplo de código, se muestra cómo usar `continue-update-rollback`.

AWS CLI

Nuevo intento de reversión de actualizaciones

El ejemplo de `continue-update-rollback` a continuación reanuda una operación de reversión a partir de un intento fallido previo de una actualización de pila.

```
aws cloudformation continue-update-rollback \  
  --stack-name my-stack
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [ContinueUpdateRollback](#) la Referencia de AWS CLI comandos.

## **create-change-set**

En el siguiente ejemplo de código, se muestra cómo usar `create-change-set`.

AWS CLI

Para crear un conjunto de cambios

En el siguiente `create-change-set` ejemplo, se crea un conjunto de cambios con `CAPABILITY_IAM` esta capacidad. El archivo `template.yaml` es una AWS CloudFormation plantilla de la carpeta actual que define una pila que incluye IAM recursos.

```
aws cloudformation create-change-set \  
  --stack-name my-stack
```

```
--stack-name my-application \  
--change-set-name my-change-set \  
--template-body file://template.yaml \  
--capabilities CAPABILITY_IAM
```

Salida:

```
{  
  "Id": "arn:aws:cloudformation:us-west-2:123456789012:changeSet/my-change-set/  
bc9555ba-a949-xmpl-bfb8-f41d04ec5784",  
  "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-application/  
d0a825a0-e4cd-xmpl-b9fb-061c69e99204"  
}
```

- Para API obtener más información, consulte [CreateChangeSet](#) la Referencia de AWS CLI comandos.

## create-stack-instances

En el siguiente ejemplo de código, se muestra cómo usar `create-stack-instances`.

### AWS CLI

Para crear instancias de pila

En el siguiente `create-stack-instances` ejemplo, se crean instancias de un conjunto de pilas en dos cuentas y cuatro regiones. La configuración de tolerancia a errores garantiza que se intente la actualización en todas las cuentas y regiones, incluso si no se pueden crear algunas pilas.

```
aws cloudformation create-stack-instances \  
--stack-set-name my-stack-set \  
--accounts 123456789012 223456789012 \  
--regions us-east-1 us-east-2 us-west-1 us-west-2 \  
--operation-preferences FailureToleranceCount=7
```

Salida:

```
{  
  "OperationId": "d7995c31-83c2-xmpl-a3d4-e9ca2811563f"
```

```
}
```

Para crear un conjunto de pilas, utilice el `create-stack-set` comando.

- Para API obtener más información, consulte [CreateStackInstances](#) la Referencia de AWS CLI comandos.

## **create-stack-set**

En el siguiente ejemplo de código, se muestra cómo usar `create-stack-set`.

### AWS CLI

Para crear un conjunto de pilas

En el siguiente `create-stack-set` ejemplo, se crea un conjunto de pilas mediante la plantilla de YAML archivo especificada. `template.yaml` es una AWS CloudFormation plantilla de la carpeta actual que define una pila.

```
aws cloudformation create-stack-set \  
  --stack-set-name my-stack-set \  
  --template-body file://template.yaml \  
  --description "SNS topic"
```

Salida:

```
{  
  "StackSetId": "my-stack-set:8d0f160b-d157-xmpl-a8e6-c0ce8e5d8cc1"  
}
```

Para añadir instancias de pila al conjunto de pilas, usa el `create-stack-instances` comando.

- Para API obtener más información, consulte [CreateStackSet](#) la Referencia de AWS CLI comandos.

## **create-stack**

En el siguiente ejemplo de código, se muestra cómo usar `create-stack`.

### AWS CLI

Para crear una AWS CloudFormation pila

En el siguiente ejemplo de `create-stacks`, se crea una pila con el nombre `myteststack` mediante la plantilla `sampletemplate.json`:

```
aws cloudformation create-stack --stack-name myteststack --template-body file://sampletemplate.json --parameters ParameterKey=KeyPairName,ParameterValue=TestKey  
ParameterKey=SubnetIDs,ParameterValue=SubnetID1\\,SubnetID2
```

Salida:

```
{  
  "StackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/  
myteststack/466df9e0-0dff-08e3-8e2f-5088487c4896"  
}
```

Para obtener más información, consulte `Stacks` en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [CreateStack](#) la Referencia de AWS CLI comandos.

## deactivate-type

En el siguiente ejemplo de código, se muestra cómo usar `deactivate-type`.

### AWS CLI

Para desactivar un tipo

En el siguiente `deactivate-type` ejemplo, se desactiva una extensión pública que se había activado anteriormente en esta cuenta y región.

```
aws cloudformation deactivate-type \  
  --region us-west-2 \  
  --type MODULE \  
  --type-name Example::Test::Type::MODULE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso del AWS CloudFormation registro](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [DeactivateType](#) la Referencia de AWS CLI comandos.

## delete-change-set

En el siguiente ejemplo de código, se muestra cómo usar `delete-change-set`.

### AWS CLI

Para eliminar un conjunto de cambios

En el siguiente `delete-change-set` ejemplo, se elimina un conjunto de cambios especificando el nombre del conjunto de cambios y el nombre de la pila.

```
aws cloudformation delete-change-set \  
  --stack-name my-stack \  
  --change-set-name my-change-set
```

Este comando no genera ninguna salida.

En el siguiente `delete-change-set` ejemplo, se elimina un conjunto de cambios especificando el conjunto ARN de cambios completo.

```
aws cloudformation delete-change-set \  
  --change-set-name arn:aws:cloudformation:us-east-2:123456789012:changeSet/my-change-set/4eca1a01-e285-xmpl-8026-9a1967bfb4b0
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteChangeSet](#) la Referencia de AWS CLI comandos.

## delete-stack-instances

En el siguiente ejemplo de código, se muestra cómo usar `delete-stack-instances`.

### AWS CLI

Para eliminar instancias de pila

En el siguiente `delete-stack-instances` ejemplo, se eliminan las instancias de un conjunto de pilas en dos cuentas de dos regiones y se cierran las pilas.

```
aws cloudformation delete-stack-instances \  
  --stack-name my-stack \  
  --region us-east-1 \  
  --account-id 123456789012 \  
  --region us-east-2 \  
  --account-id 123456789012
```

```
--stack-set-name my-stack-set \  
--accounts 123456789012 567890123456 \  
--regions us-east-1 us-west-1 \  
--no-retain-stacks
```

Salida:

```
{  
  "OperationId": "ad49f10c-fd1d-413f-a20a-8de6e2fa8f27"  
}
```

Para eliminar un conjunto de pilas vacío, usa el comando. `delete-stack-set`

- Para API obtener más información, consulte [DeleteStackInstances](#) la Referencia de AWS CLI comandos.

## **delete-stack-set**

En el siguiente ejemplo de código, se muestra cómo usar `delete-stack-set`.

AWS CLI

Para eliminar un conjunto de pilas

El siguiente comando elimina el conjunto de pilas vacío especificado. El conjunto de pilas debe estar vacío.

```
aws cloudformation delete-stack-set \  
--stack-set-name my-stack-set
```

Este comando no genera ninguna salida.

Para eliminar instancias del conjunto de pilas, usa el `delete-stack-instances` comando.

- Para API obtener más información, consulte [DeleteStackSet](#) la Referencia de AWS CLI comandos.

## **delete-stack**

En el siguiente ejemplo de código, se muestra cómo usar `delete-stack`.

## AWS CLI

Eliminación de una pila

En el siguiente ejemplo de `delete-stack`, se elimina la pila especificada.

```
aws cloudformation delete-stack \  
  --stack-name my-stack
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteStack](#) la Referencia de AWS CLI comandos.

## deploy

En el siguiente ejemplo de código, se muestra cómo usar `deploy`.

### AWS CLI

El siguiente comando despliega la plantilla nombrada `template.json` en una pila llamada `my-new-stack`:

```
aws cloudformation deploy --template-file /path_to_template/template.json \  
  --stack-name my-new-stack --parameter-overrides Key1=Value1 Key2=Value2 -- \  
  tags Key1=Value1 Key2=Value2
```

- Para API obtener más información, consulte [Deploy](#) in AWS CLI Command Reference.

## deregister-type

En el siguiente ejemplo de código, se muestra cómo usar `deregister-type`.

### AWS CLI

Para anular el registro de una versión de tipo

En el siguiente `deregister-type` ejemplo, se quita la versión de tipo especificada del uso activo en el CloudFormation registro, de modo que ya no se puede utilizar en CloudFormation las operaciones.

```
aws cloudformation deregister-type \  
  --stack-name my-stack --type-name my-type
```



```
--type RESOURCE \  
--type-name My::Logs::LogGroup \  
--version-id 00000002
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso del CloudFormation registro](#) en la AWS CloudFormation Guía del usuario.

- Para API obtener más información, consulte [DeregisterType](#) la Referencia de AWS CLI comandos.

## describe-account-limits

En el siguiente ejemplo de código, se muestra cómo usar `describe-account-limits`.

AWS CLI

Para obtener información sobre los límites de su cuenta

El siguiente comando recupera una lista de los límites regionales de la cuenta corriente.

```
aws cloudformation describe-account-limits
```

Salida:

```
{  
  "AccountLimits": [  
    {  
      "Name": "StackLimit",  
      "Value": 200  
    },  
    {  
      "Name": "StackOutputsLimit",  
      "Value": 60  
    },  
    {  
      "Name": "ConcurrentResourcesLimit",  
      "Value": 2500  
    }  
  ]  
}
```

- Para API obtener más información, consulte [DescribeAccountLimits](#) la Referencia de AWS CLI comandos.

## describe-change-set

En el siguiente ejemplo de código, se muestra cómo usar `describe-change-set`.

### AWS CLI

Para obtener información sobre un conjunto de cambios

En el siguiente `describe-change-set` ejemplo, se muestran los detalles del conjunto de cambios especificados por el nombre del conjunto de cambios y el nombre de la pila.

```
aws cloudformation describe-change-set \  
  --change-set-name my-change-set \  
  --stack-name my-stack
```

En el siguiente `describe-change-set` ejemplo, se muestran los detalles del conjunto de cambios especificados por el conjunto ARN de cambios completo:

```
aws cloudformation describe-change-set \  
  --change-set-name arn:aws:cloudformation:us-west-2:123456789012:changeSet/my-change-set/bc9555ba-a949-xmpl-bfb8-f41d04ec5784
```

Salida:

```
{  
  "Changes": [  
    {  
      "Type": "Resource",  
      "ResourceChange": {  
        "Action": "Modify",  
        "LogicalResourceId": "function",  
        "PhysicalResourceId": "my-function-SEZV4XMPL4S5",  
        "ResourceType": "AWS::Lambda::Function",  
        "Replacement": "False",  
        "Scope": [  
          "Properties"  
        ],  
        "Details": [  

```

```

        {
            "Target": {
                "Attribute": "Properties",
                "Name": "Timeout",
                "RequiresRecreation": "Never"
            },
            "Evaluation": "Static",
            "ChangeSource": "DirectModification"
        }
    ]
}
],
"ChangeSetName": "my-change-set",
"ChangeSetId": "arn:aws:cloudformation:us-west-2:123456789012:changeSet/my-
change-set/4eca1a01-e285-xmpl-8026-9a1967bfb4b0",
"StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/
d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
"StackName": "my-stack",
"Description": null,
"Parameters": null,
"CreationTime": "2019-10-02T05:20:56.651Z",
"ExecutionStatus": "AVAILABLE",
"Status": "CREATE_COMPLETE",
"StatusReason": null,
"NotificationARNs": [],
"RollbackConfiguration": {},
"Capabilities": [
    "CAPABILITY_IAM"
],
"Tags": null
}

```

- Para API obtener más información, consulte [DescribeChangeSet](#) la Referencia de AWS CLI comandos.

## describe-publisher

En el siguiente ejemplo de código, se muestra cómo usar describe-publisher.

### AWS CLI

Para describir un editor

El siguiente describe-publisher ejemplo configura la información de un editor.

```
aws cloudformation describe-publisher \  
  --region us-west-2 \  
  --publisher-id 000q6TfUovXsEMmgKowxDZLLwqr2QUsh
```

Salida:

```
{  
  "PublisherId": "000q6TfUovXsEMmgKowxDZLLwqr2QUshd2e75c8c",  
  "PublisherStatus": "VERIFIED",  
  "IdentityProvider": "AWS_Marketplace",  
  "PublisherProfile": "https://aws.amazon.com/marketplace/seller-profile?  
id=2c5dc1f0-17cd-4259-8e46-822a83gdteg"  
}
```

Para obtener más información, consulte [Uso del AWS CloudFormation registro](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [DescribePublisher](#) la Referencia de AWS CLI comandos.

## describe-stack-drift-detection-status

En el siguiente ejemplo de código, se muestra cómo usar describe-stack-drift-detection-status.

AWS CLI

Para comprobar el estado de una operación de detección de desviaciones

El siguiente describe-stack-drift-detection-status ejemplo muestra el estado de una operación de detección de desviaciones. Obtenga el by ID al ejecutar el detect-stack-drift comando.

```
aws cloudformation describe-stack-drift-detection-status \  
  --stack-drift-detection-id 1a229160-e4d9-xmpl-ab67-0a4f93df83d4
```

Salida:

```
{
```

```

    "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/
d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
    "StackDriftDetectionId": "1a229160-e4d9-xmpl-ab67-0a4f93df83d4",
    "StackDriftStatus": "DRIFTED",
    "DetectionStatus": "DETECTION_COMPLETE",
    "DriftedStackResourceCount": 1,
    "Timestamp": "2019-10-02T05:54:30.902Z"
}

```

- Para API obtener más información, consulte [DescribeStackDriftDetectionStatus](#) la Referencia de AWS CLI comandos.

## describe-stack-events

En el siguiente ejemplo de código, se muestra cómo usar `describe-stack-events`.

### AWS CLI

Descripción de los eventos de la pila

En el siguiente ejemplo de `describe-stack-events`, se muestran los dos eventos más recientes de la pila especificada.

```

aws cloudformation describe-stack-events \
  --stack-name my-stack \
  --max-items 2
{
  "StackEvents": [
    {
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "EventId": "4e1516d0-e4d6-xmpl-b94f-0a51958a168c",
      "StackName": "my-stack",
      "LogicalResourceId": "my-stack",
      "PhysicalResourceId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/my-stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "ResourceType": "AWS::CloudFormation::Stack",
      "Timestamp": "2019-10-02T05:34:29.556Z",
      "ResourceStatus": "UPDATE_COMPLETE"
    },
    {

```

```

    "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
    "EventId": "4dd3c810-e4d6-xmpl-bade-0aaf8b31ab7a",
    "StackName": "my-stack",
    "LogicalResourceId": "my-stack",
    "PhysicalResourceId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/my-stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
    "ResourceType": "AWS::CloudFormation::Stack",
    "Timestamp": "2019-10-02T05:34:29.127Z",
    "ResourceStatus": "UPDATE_COMPLETE_CLEANUP_IN_PROGRESS"
  },
],
"NextToken": "eyJJOZXh0VG9XMPLi0iBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQi0iAyfQ=="
}

```

- Para API obtener más información, consulte [DescribeStackEvents](#) la Referencia de AWS CLI comandos.

## describe-stack-instance

En el siguiente ejemplo de código, se muestra cómo usar `describe-stack-instance`.

### AWS CLI

Para describir una instancia de pila

El siguiente comando describe una instancia del conjunto de pilas especificado en la cuenta y la región especificadas. El conjunto de pilas se encuentra en la región y la cuenta actuales, y la instancia se encuentra en la `us-west-2` región de la cuenta `123456789012`. :

```

aws cloudformation describe-stack-instance \
  --stack-set-name my-stack-set \
  --stack-instance-account 123456789012 \
  --stack-instance-region us-west-2

```

Salida:

```

{
  "StackInstance": {
    "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",

```

```

    "Region": "us-west-2",
    "Account": "123456789012",
    "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/
StackSet-enable-config-e6cac20f-xmpl-46e9-8314-53e0d4591532/4287f9a0-e615-
xmpl-894a-12b31d3117be",
    "ParameterOverrides": [],
    "Status": "OUTDATED",
    "StatusReason": "ResourceLogicalId:ConfigBucket,
ResourceType:AWS::S3::Bucket, ResourceStatusReason:You have attempted to create
more buckets than allowed (Service: Amazon S3; Status Code: 400; Error Code:
TooManyBuckets; Request ID: F7F21CXMPL580224; S3 Extended Request ID: egd/
Fdt89BXMPlyiqbMNLjVk55Yqqvi3NYW2nKLUVWhUGEhNfCmZdyj967lhriaG/dWMobS040o=)."
  }
}

```

- Para API obtener más información, consulte [DescribeStackInstance](#) la Referencia de AWS CLI comandos.

## describe-stack-resource-drifts

En el siguiente ejemplo de código, se muestra cómo usar `describe-stack-resource-drifts`.

### AWS CLI

Para obtener información sobre los recursos que se derivaron de la definición de pila

El siguiente comando muestra información sobre los recursos desviados de la pila especificada. Para iniciar la detección de desviaciones, utilice el `detect-stack-drift` comando. :

```

aws cloudformation describe-stack-resource-drifts \
  --stack-name my-stack

```

El resultado muestra una función AWS Lambda que se modificó: out-of-band

```

{
  "StackResourceDrifts": [
    {
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "LogicalResourceId": "function",
      "PhysicalResourceId": "my-function-SEZV4XMPL4S5",

```

```

        "ResourceType": "AWS::Lambda::Function",
        "ExpectedProperties": "{\"Description\":\"Write a file to S3.\",
\\\"Environment\\\":{\\\"Variables\\\":{\\\"bucket\\\":\\\"my-stack-bucket-1vc62xmplgguf
\\\"}},\\\"Handler\\\":\\\"index.handler\\\",\\\"MemorySize\\\":128,\\\"Role\\\":
\\\"arn:aws:iam::123456789012:role/my-functionRole-HIZXMPLE0M9E\\\",\\\"Runtime\\\":
\\\"nodejs10.x\\\",\\\"Tags\\\":[{\\\"Key\\\":\\\"lambda:createdBy\\\",\\\"Value\\\":\\\"SAM\\\"}],\\\"Timeout
\\\":900,\\\"TracingConfig\\\":{\\\"Mode\\\":\\\"Active\\\"}}\",
        \"ActualProperties\": \"{\\\"Description\\\":\\\"Write a file to S3.\",
\\\"Environment\\\":{\\\"Variables\\\":{\\\"bucket\\\":\\\"my-stack-bucket-1vc62xmplgguf
\\\"}},\\\"Handler\\\":\\\"index.handler\\\",\\\"MemorySize\\\":256,\\\"Role\\\":
\\\"arn:aws:iam::123456789012:role/my-functionRole-HIZXMPLE0M9E\\\",\\\"Runtime\\\":
\\\"nodejs10.x\\\",\\\"Tags\\\":[{\\\"Key\\\":\\\"lambda:createdBy\\\",\\\"Value\\\":\\\"SAM\\\"}],\\\"Timeout
\\\":22,\\\"TracingConfig\\\":{\\\"Mode\\\":\\\"Active\\\"}}\",
        \"PropertyDifferences\": [
            {
                \"PropertyPath\": \"/MemorySize\",
                \"ExpectedValue\": \"128\",
                \"ActualValue\": \"256\",
                \"DifferenceType\": \"NOT_EQUAL\"
            },
            {
                \"PropertyPath\": \"/Timeout\",
                \"ExpectedValue\": \"900\",
                \"ActualValue\": \"22\",
                \"DifferenceType\": \"NOT_EQUAL\"
            }
        ],
        \"StackResourceDriftStatus\": \"MODIFIED\",
        \"Timestamp\": \"2019-10-02T05:54:44.064Z\"
    }
]
}

```

- Para API obtener más información, consulte [DescribeStackResourceDrifts](#) la Referencia de AWS CLI comandos.

## describe-stack-resource

En el siguiente ejemplo de código, se muestra cómo usar describe-stack-resource.

### AWS CLI

Obtención de información sobre un recurso de pila



En el siguiente ejemplo de `describe-stack-resource`, se muestran los detalles del recurso con el nombre `MyFunction` en la pila especificada.

```
aws cloudformation describe-stack-resource \  
  --stack-name MyStack \  
  --logical-resource-id MyFunction
```

Salida:

```
{  
  "StackResourceDetail": {  
    "StackName": "MyStack",  
    "StackId": "arn:aws:cloudformation:us-east-2:123456789012:stack/MyStack/  
d0a825a0-e4cd-xmpl-b9fb-061c69e99204",  
    "LogicalResourceId": "MyFunction",  
    "PhysicalResourceId": "my-function-SEZV4XMPL4S5",  
    "ResourceType": "AWS::Lambda::Function",  
    "LastUpdatedTimestamp": "2019-10-02T05:34:27.989Z",  
    "ResourceStatus": "UPDATE_COMPLETE",  
    "Metadata": "{}",  
    "DriftInformation": {  
      "StackResourceDriftStatus": "IN_SYNC"  
    }  
  }  
}
```

- Para API obtener más información, consulte [DescribeStackResource](#) la Referencia de AWS CLI comandos.

## describe-stack-resources

En el siguiente ejemplo de código, se muestra cómo usar `describe-stack-resources`.

### AWS CLI

Obtención de información sobre un recurso de pila

En el siguiente ejemplo de `describe-stack-resources`, se muestran los detalles de los recursos en la pila especificada.

```
aws cloudformation describe-stack-resources \  
  --stack-name MyStack
```

```
--stack-name my-stack
```

Salida:

```
{
  "StackResources": [
    {
      "StackName": "my-stack",
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "LogicalResourceId": "bucket",
      "PhysicalResourceId": "my-stack-bucket-1vc62xmplgguf",
      "ResourceType": "AWS::S3::Bucket",
      "Timestamp": "2019-10-02T04:34:11.345Z",
      "ResourceStatus": "CREATE_COMPLETE",
      "DriftInformation": {
        "StackResourceDriftStatus": "IN_SYNC"
      }
    },
    {
      "StackName": "my-stack",
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "LogicalResourceId": "function",
      "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
      "ResourceType": "AWS::Lambda::Function",
      "Timestamp": "2019-10-02T05:34:27.989Z",
      "ResourceStatus": "UPDATE_COMPLETE",
      "DriftInformation": {
        "StackResourceDriftStatus": "IN_SYNC"
      }
    },
    {
      "StackName": "my-stack",
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "LogicalResourceId": "functionRole",
      "PhysicalResourceId": "my-functionRole-HIZXMPLE0M9E",
      "ResourceType": "AWS::IAM::Role",
      "Timestamp": "2019-10-02T04:34:06.350Z",
      "ResourceStatus": "CREATE_COMPLETE",
      "DriftInformation": {
        "StackResourceDriftStatus": "IN_SYNC"
      }
    }
  ]
}
```

```

    }
  }
]
}

```

- Para API obtener más información, consulte [DescribeStackResources](#) la Referencia de AWS CLI comandos.

## describe-stack-set-operation

En el siguiente ejemplo de código, se muestra cómo usar `describe-stack-set-operation`.

### AWS CLI

Para obtener información sobre una operación de conjunto de pilas

El siguiente `describe-stack-set-operation` ejemplo muestra los detalles de una operación de actualización en el conjunto de pilas especificado.

```

aws cloudformation describe-stack-set-operation \
  --stack-set-name enable-config \
  --operation-id 35d45ebc-ed88-xmpl-ab59-0197a1fc83a0

```

Salida:

```

{
  "StackSetOperation": {
    "OperationId": "35d45ebc-ed88-xmpl-ab59-0197a1fc83a0",
    "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
    "Action": "UPDATE",
    "Status": "SUCCEEDED",
    "OperationPreferences": {
      "RegionOrder": [
        "us-east-1",
        "us-west-2",
        "eu-west-1",
        "us-west-1"
      ],
      "FailureToleranceCount": 7,
      "MaxConcurrentCount": 2
    }
  },
}

```

```

    "AdministrationRoleARN": "arn:aws:iam::123456789012:role/
AWSCloudFormationStackSetAdministrationRole",
    "ExecutionRoleName": "AWSCloudFormationStackSetExecutionRole",
    "CreationTimestamp": "2019-10-03T16:28:44.377Z",
    "EndTimestamp": "2019-10-03T16:42:08.607Z"
  }
}

```

- Para API obtener más información, consulte [DescribeStackSetOperation](#) la Referencia de AWS CLI comandos.

## describe-stack-set

En el siguiente ejemplo de código, se muestra cómo usar `describe-stack-set`.

### AWS CLI

Para obtener información sobre un conjunto de pilas

El siguiente `describe-stack-set` ejemplo muestra detalles sobre el conjunto de pilas especificado.

```

aws cloudformation describe-stack-set \
  --stack-set-name my-stack-set

```

Salida:

```

{
  "StackSet": {
    "StackSetName": "my-stack-set",
    "StackSetId": "my-stack-set:296a3360-xmpl-40af-be78-9341e95bf743",
    "Description": "Create an Amazon SNS topic",
    "Status": "ACTIVE",
    "TemplateBody": "AWSTemplateFormatVersion: '2010-09-09'\nDescription: An AWS
SNS topic\nResources:\n  topic:\n    Type: AWS::SNS::Topic",
    "Parameters": [],
    "Capabilities": [],
    "Tags": [],
    "StackSetARN": "arn:aws:cloudformation:us-west-2:123456789012:stackset/
enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
    "AdministrationRoleARN": "arn:aws:iam::123456789012:role/
AWSCloudFormationStackSetAdministrationRole",
    "ExecutionRoleName": "AWSCloudFormationStackSetExecutionRole"
  }
}

```

```
}  
}
```

- Para API obtener más información, consulte [DescribeStackSet](#) la Referencia de AWS CLI comandos.

## describe-stacks

En el siguiente ejemplo de código, se muestra cómo usar `describe-stacks`.

### AWS CLI

Para describir las AWS CloudFormation pilas

En el siguiente ejemplo de `describe-stacks`, se muestra la información del resumen para la pila `myteststack`:

```
aws cloudformation describe-stacks --stack-name myteststack
```

Salida:

```
{  
  "Stacks": [  
    {  
      "StackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/  
myteststack/466df9e0-0dff-08e3-8e2f-5088487c4896",  
      "Description": "AWS CloudFormation Sample Template S3_Bucket: Sample  
template showing how to create a publicly accessible S3 bucket. **WARNING** This  
template creates an S3 bucket. You will be billed for the AWS resources used if you  
create a stack from this template.",  
      "Tags": [],  
      "Outputs": [  
        {  
          "Description": "Name of S3 bucket to hold website content",  
          "OutputKey": "BucketName",  
          "OutputValue": "myteststack-s3bucket-jssofi1zie2w"  
        }  
      ],  
      "StackStatusReason": null,  
      "CreationTime": "2013-08-23T01:02:15.422Z",  
      "Capabilities": [],  
      "StackName": "myteststack",
```

```

        "StackStatus": "CREATE_COMPLETE",
        "DisableRollback": false
    }
]
}

```

Para obtener más información, consulte Stacks en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [DescribeStacks](#) la Referencia de AWS CLI comandos.

## describe-type-registration

En el siguiente ejemplo de código, se muestra cómo usar `describe-type-registration`.

### AWS CLI

Para mostrar la información de registro de tipos

En el siguiente `describe-type-registration` ejemplo, se muestra información sobre el registro del tipo especificado, incluidos el estado actual, el tipo y la versión del tipo.

```

aws cloudformation describe-type-registration \
  --registration-token a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

```

Salida:

```

{
  "ProgressStatus": "COMPLETE",
  "TypeArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/My-Logs-LogGroup",
  "Description": "Deployment is currently in DEPLOY_STAGE of status COMPLETED; ",
  "TypeVersionArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/My-Logs-LogGroup/00000001"
}

```

Para obtener más información, consulte [Uso del CloudFormation registro](#) en la AWS CloudFormation Guía del usuario.

- Para API obtener más información, consulte [DescribeTypeRegistration](#) la Referencia de AWS CLI comandos.

## describe-type

En el siguiente ejemplo de código, se muestra cómo usar `describe-type`.

### AWS CLI

Para mostrar la información de tipos

En el siguiente `describe-type` ejemplo, se muestra la información del tipo especificado.

```
aws cloudformation describe-type \  
  --type-name My::Logs::LogGroup \  
  --type RESOURCE
```

Salida:

```
{  
  "SourceUrl": "https://github.com/aws-cloudformation/aws-cloudformation-resource-providers-logs.git",  
  "Description": "Customized resource derived from AWS::Logs::LogGroup",  
  "TimeCreated": "2019-12-03T23:29:33.321Z",  
  "Visibility": "PRIVATE",  
  "TypeName": "My::Logs::LogGroup",  
  "LastUpdated": "2019-12-03T23:29:33.321Z",  
  "DeprecatedStatus": "LIVE",  
  "ProvisioningType": "FULLY_MUTABLE",  
  "Type": "RESOURCE",  
  "Arn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/My-Logs-LogGroup/00000001",  
  "Schema": "[details omitted]"  
}
```

Para obtener más información, consulte [Uso del CloudFormation registro](#) en la AWS CloudFormation Guía del usuario.

- Para API obtener más información, consulte [DescribeType](#) la Referencia de AWS CLI comandos.

## detect-stack-drift

En el siguiente ejemplo de código, se muestra cómo usar `detect-stack-drift`.

## AWS CLI

Para detectar recursos desviados

El siguiente `detect-stack-drift` ejemplo inicia la detección de desviaciones para la pila especificada.

```
aws cloudformation detect-stack-drift \  
  --stack-name my-stack
```

Salida:

```
{  
  "StackDriftDetectionId": "1a229160-e4d9-xmpl-ab67-0a4f93df83d4"  
}
```

A continuación, puede utilizar este identificador con el `describe-stack-resource-drifts` comando para describir los recursos desviados.

- Para API obtener más información, consulte [DetectStackDrift](#) la Referencia de AWS CLI comandos.

## `detect-stack-resource-drift`

En el siguiente ejemplo de código, se muestra cómo usar `detect-stack-resource-drift`.

## AWS CLI

Para detectar la desviación de un recurso

En el siguiente `detect-stack-resource-drift` ejemplo, se comprueba un recurso cuyo nombre figura `MyFunction` en una pila denominada `MyStack` deriva:

```
aws cloudformation detect-stack-resource-drift \  
  --stack-name MyStack \  
  --logical-resource-id MyFunction
```

El resultado muestra una función AWS Lambda que se modificó: `out-of-band`

```
{
```



```

"StackResourceDrift": {
  "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/MyStack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
  "LogicalResourceId": "MyFunction",
  "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
  "ResourceType": "AWS::Lambda::Function",
  "ExpectedProperties": "{\"Description\": \"Write a file to S3.\",
  \"Environment\": {\"Variables\": {\"bucket\": \"my-stack-bucket-1vc62xmplgguf\"}},
  \"Handler\": \"index.handler\", \"MemorySize\": 128, \"Role\": \"arn:aws:iam:123456789012:role/my-functionRole-HIZXMPLE0M9E\", \"Runtime\": \"nodejs10.x\",
  \"Tags\": [{\"Key\": \"lambda:createdBy\", \"Value\": \"SAM\"}], \"Timeout\": 900,
  \"TracingConfig\": {\"Mode\": \"Active\"}}",
  "ActualProperties": "{\"Description\": \"Write a file to S3.\", \"Environment\": {\"Variables\": {\"bucket\": \"my-stack-bucket-1vc62xmplgguf\"}},
  \"Handler\": \"index.handler\", \"MemorySize\": 256, \"Role\": \"arn:aws:iam:123456789012:role/my-functionRole-HIZXMPLE0M9E\", \"Runtime\": \"nodejs10.x\",
  \"Tags\": [{\"Key\": \"lambda:createdBy\", \"Value\": \"SAM\"}], \"Timeout\": 22, \"TracingConfig\": {\"Mode\": \"Active\"}}",
  "PropertyDifferences": [
    {
      "PropertyPath": "/MemorySize",
      "ExpectedValue": "128",
      "ActualValue": "256",
      "DifferenceType": "NOT_EQUAL"
    },
    {
      "PropertyPath": "/Timeout",
      "ExpectedValue": "900",
      "ActualValue": "22",
      "DifferenceType": "NOT_EQUAL"
    }
  ],
  "StackResourceDriftStatus": "MODIFIED",
  "Timestamp": "2019-10-02T05:58:47.433Z"
}
}

```

- Para API obtener más información, consulte [DetectStackResourceDrift](#) la Referencia de AWS CLI comandos.

## detect-stack-set-drift

En el siguiente ejemplo de código, se muestra cómo usar `detect-stack-set-drift`.

## AWS CLI

Para detectar la desviación en un conjunto de pilas y en todas las instancias de pilas asociadas

El siguiente `detect-stack-set-drift` ejemplo inicia las operaciones de detección de desviaciones en el conjunto de pilas especificado, incluidas todas las instancias de apilamiento asociadas a ese conjunto de pilas, y devuelve un identificador de operación que se puede utilizar para realizar un seguimiento del estado de la operación de desviación.

```
aws cloudformation detect-stack-set-drift \  
  --stack-set-name stack-set-drift-example
```

Salida:

```
{  
  "OperationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE111111"  
}
```

Para obtener más información, consulte [Detección de cambios de configuración no gestionados en conjuntos de pilas](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [DetectStackSetDrift](#) la Referencia de AWS CLI comandos.

## `estimate-template-cost`

En el siguiente ejemplo de código, se muestra cómo usar `estimate-template-cost`.

## AWS CLI

Cálculo del costo de una plantilla

En el siguiente ejemplo de `estimate-template-cost`, se genera un cálculo del costo para una plantilla con el nombre `template.yaml` en la carpeta actual.

```
aws cloudformation estimate-template-cost \  
  --template-body file://template.yaml
```

Salida:

```
{
```

```
"Url": "http://calculator.s3.amazonaws.com/calc5.html?
key=cloudformation/7870825a-xmpl-4def-92e7-c4f8dd360cca"
}
```

- Para API obtener más información, consulte [EstimateTemplateCost](#) la Referencia de AWS CLI comandos.

## execute-change-set

En el siguiente ejemplo de código, se muestra cómo usar `execute-change-set`.

### AWS CLI

Para ejecutar un conjunto de cambios

En el siguiente `execute-change-set` ejemplo, se ejecuta un conjunto de cambios especificado por el nombre del conjunto de cambios y el nombre de la pila.

```
aws cloudformation execute-change-set \
  --change-set-name my-change-set \
  --stack-name my-stack
```

En el siguiente `execute-change-set` ejemplo, se ejecuta un conjunto de cambios especificado por el conjunto ARN de cambios completo.

```
aws cloudformation execute-change-set \
  --change-set-name arn:aws:cloudformation:us-west-2:123456789012:changeSet/my-
change-set/bc9555ba-a949-xmpl-bfb8-f41d04ec5784
```

- Para API obtener más información, consulte [ExecuteChangeSet](#) la Referencia de AWS CLI comandos.

## get-stack-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-stack-policy`.

### AWS CLI

Para ver una política de apilamiento

En el siguiente `get-stack-policy` ejemplo, se muestra la política de apilamiento de la pila especificada. Para adjuntar una política a una pila, utilice el `set-stack-policy` comando.

```
aws cloudformation get-stack-policy \
  --stack-name my-stack
```

Salida:

```
{
  "StackPolicyBody": "{\n  \"Statement\" : [\n    {\n      \"Effect\" :\n  \"Allow\", \n      \"Action\" : \"Update:*\", \n      \"Principal\": \"*\", \n      \"Resource\" : \"*\" \n    }, \n    {\n      \"Effect\" : \"Deny\", \n      \"Action\" : \"Update:*\", \n      \"Principal\": \"*\", \n      \"Resource\" :\n  \"LogicalResourceId/bucket\" \n    } \n  ]\n}"
```

- Para API obtener más información, consulte [GetStackPolicy](#) la Referencia de AWS CLI comandos.

## get-template-summary

En el siguiente ejemplo de código, se muestra cómo usar `get-template-summary`.

AWS CLI

Para mostrar un resumen de la plantilla

El siguiente comando muestra información resumida sobre los recursos y los metadatos del archivo de plantilla especificado.

```
aws cloudformation get-template-summary \
  --template-body file://template.yaml
```

Salida:

```
{
  "Parameters": [],
  "Description": "A VPC and subnets.",
  "ResourceTypes": [
    "AWS::EC2::VPC",
    "AWS::EC2::Subnet",
```

```

    "AWS::EC2::Subnet",
    "AWS::EC2::RouteTable",
    "AWS::EC2::VPCEndpoint",
    "AWS::EC2::SubnetRouteTableAssociation",
    "AWS::EC2::SubnetRouteTableAssociation",
    "AWS::EC2::VPCEndpoint"
  ],
  "Version": "2010-09-09"
}

```

- Para API obtener más información, consulte [GetTemplateSummary](#) la Referencia de AWS CLI comandos.

## get-template

En el siguiente ejemplo de código, se muestra cómo usar `get-template`.

### AWS CLI

Para ver el cuerpo de la plantilla de una AWS CloudFormation pila

En el siguiente ejemplo de `get-template`, se muestra la plantilla para la pila `myteststack`:

```
aws cloudformation get-template --stack-name myteststack
```

Salida:

```

{
  "TemplateBody": {
    "AWSTemplateFormatVersion": "2010-09-09",
    "Outputs": {
      "BucketName": {
        "Description": "Name of S3 bucket to hold website content",
        "Value": {
          "Ref": "S3Bucket"
        }
      }
    }
  },
  "Description": "AWS CloudFormation Sample Template S3_Bucket: Sample
template showing how to create a publicly accessible S3 bucket. **WARNING** This
template creates an S3 bucket. You will be billed for the AWS resources used if you
create a stack from this template.",

```

```

    "Resources": {
      "S3Bucket": {
        "Type": "AWS::S3::Bucket",
        "Properties": {
          "AccessControl": "PublicRead"
        }
      }
    }
  }
}

```

- Para API obtener más información, consulte [GetTemplate](#) la Referencia de AWS CLI comandos.

## list-change-sets

En el siguiente ejemplo de código, se muestra cómo usar `list-change-sets`.

### AWS CLI

Para enumerar los conjuntos de cambios

En el siguiente `list-change-sets` ejemplo, se muestra una lista de los conjuntos de cambios pendientes de la pila especificada.

```

aws cloudformation list-change-sets \
  --stack-name my-stack

```

Salida:

```

{
  "Summaries": [
    {
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
      "StackName": "my-stack",
      "ChangeSetId": "arn:aws:cloudformation:us-west-2:123456789012:changeSet/
my-change-set/70160340-7914-xmpl-bcbf-128a1fa78b5d",
      "ChangeSetName": "my-change-set",
      "ExecutionStatus": "AVAILABLE",
      "Status": "CREATE_COMPLETE",
      "CreationTime": "2019-10-02T05:38:54.297Z"
    }
  ]
}

```

```
]
}
```

- Para API obtener más información, consulte [ListChangeSets](#) la Referencia de AWS CLI comandos.

## list-exports

En el siguiente ejemplo de código, se muestra cómo usar `list-exports`.

### AWS CLI

Para enumerar las exportaciones

En el siguiente `list-exports` ejemplo, se muestra una lista de las exportaciones de las pilas de la región actual.

```
aws cloudformation list-exports
```

Salida:

```
{
  "Exports": [
    {
      "ExportingStackId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/private-vpc/99764070-b56c-xmpl-bee8-062a88d1d800",
      "Name": "private-vpc-subnet-a",
      "Value": "subnet-07b410xmplddcfa03"
    },
    {
      "ExportingStackId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/private-vpc/99764070-b56c-xmpl-bee8-062a88d1d800",
      "Name": "private-vpc-subnet-b",
      "Value": "subnet-075ed3xmpllebd2fb1"
    },
    {
      "ExportingStackId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/private-vpc/99764070-b56c-xmpl-bee8-062a88d1d800",
      "Name": "private-vpc-vpcid",
      "Value": "vpc-011d7xmpl1100e9841"
    }
  ]
}
```

```
}
```

- Para API obtener más información, consulte [ListExports](#) la Referencia de AWS CLI comandos.

## list-imports

En el siguiente ejemplo de código, se muestra cómo usar `list-imports`.

### AWS CLI

Para enumerar las importaciones

En el siguiente `list-imports` ejemplo, se enumeran las pilas que importan la exportación especificada. Para obtener la lista de exportaciones disponibles, utilice el `list-exports` comando.

```
aws cloudformation list-imports \  
  --export-name private-vpc-vpcid
```

Salida:

```
{  
  "Imports": [  
    "my-database-stack"  
  ]  
}
```

- Para API obtener más información, consulte [ListImports](#) la Referencia de AWS CLI comandos.

## list-stack-instances

En el siguiente ejemplo de código, se muestra cómo usar `list-stack-instances`.

### AWS CLI

Para enumerar las instancias de una pila

En el siguiente `list-stack-instances` ejemplo, se enumeran las instancias creadas a partir del conjunto de pilas especificado.

```
aws cloudformation list-stack-instances \  
  --stack-name my-stack
```



```
--stack-set-name enable-config
```

El resultado del ejemplo incluye detalles sobre una pila que no se pudo actualizar debido a un error:

```
{
  "Summaries": [
    {
      "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
      "Region": "us-west-2",
      "Account": "123456789012",
      "StackId": "arn:aws:cloudformation:ap-northeast-1:123456789012:stack/StackSet-enable-config-35a6ac50-d9f8-4084-86e4-7da34d5de4c4/a1631cd0-e5fb-xmpl-b474-0aa20f14f06e",
      "Status": "CURRENT"
    },
    {
      "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
      "Region": "us-west-2",
      "Account": "123456789012",
      "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/StackSet-enable-config-e6cac20f-xmpl-46e9-8314-53e0d4591532/eab53680-e5fa-xmpl-ba14-0a522351f81e",
      "Status": "OUTDATED",
      "StatusReason": "ResourceLogicalId:ConfigDeliveryChannel, ResourceType:AWS::Config::DeliveryChannel, ResourceStatusReason:Failed to put delivery channel 'StackSet-enable-config-e6cac20f-xmpl-46e9-8314-53e0d4591532-ConfigDeliveryChannel-10JWJ7XD59WR0' because the maximum number of delivery channels: 1 is reached. (Service: AmazonConfig; Status Code: 400; Error Code: MaxNumberOfDeliveryChannelsExceededException; Request ID: d14b34a0-ef7c-xmpl-acf8-8a864370ae56).",
    }
  ]
}
```

- Para API obtener más información, consulte [ListStackInstances](#) la Referencia de AWS CLI comandos.

## list-stack-resources

En el siguiente ejemplo de código, se muestra cómo usar `list-stack-resources`.

## AWS CLI

### Enumeración de los recursos de una pila

En el siguiente comando, se muestra la lista de los recursos en la pila especificada.

```
aws cloudformation list-stack-resources \  
  --stack-name my-stack
```

Salida:

```
{  
  "StackResourceSummaries": [  
    {  
      "LogicalResourceId": "bucket",  
      "PhysicalResourceId": "my-stack-bucket-1vc62xmplgguf",  
      "ResourceType": "AWS::S3::Bucket",  
      "LastUpdatedTimestamp": "2019-10-02T04:34:11.345Z",  
      "ResourceStatus": "CREATE_COMPLETE",  
      "DriftInformation": {  
        "StackResourceDriftStatus": "IN_SYNC"  
      }  
    },  
    {  
      "LogicalResourceId": "function",  
      "PhysicalResourceId": "my-function-SEZV4XMPL4S5",  
      "ResourceType": "AWS::Lambda::Function",  
      "LastUpdatedTimestamp": "2019-10-02T05:34:27.989Z",  
      "ResourceStatus": "UPDATE_COMPLETE",  
      "DriftInformation": {  
        "StackResourceDriftStatus": "IN_SYNC"  
      }  
    },  
    {  
      "LogicalResourceId": "functionRole",  
      "PhysicalResourceId": "my-functionRole-HIZXMPLEOM9E",  
      "ResourceType": "AWS::IAM::Role",  
      "LastUpdatedTimestamp": "2019-10-02T04:34:06.350Z",  
      "ResourceStatus": "CREATE_COMPLETE",  
      "DriftInformation": {  
        "StackResourceDriftStatus": "IN_SYNC"  
      }  
    }  
  ]  
}
```

```
]
}
```

- Para API obtener más información, consulte [ListStackResources](#) la Referencia de AWS CLI comandos.

## list-stack-set-operation-results

En el siguiente ejemplo de código, se muestra cómo usar `list-stack-set-operation-results`.

### AWS CLI

Para enumerar los resultados de las operaciones del conjunto de pilas

El siguiente comando muestra los resultados de una operación de actualización en las instancias del conjunto de pilas especificado.

```
aws cloudformation list-stack-set-operation-results \
  --stack-set-name enable-config \
  --operation-id 35d45ebc-ed88-xmpl-ab59-0197a1fc83a0
```

Salida:

```
{
  "Summaries": [
    {
      "Account": "223456789012",
      "Region": "us-west-2",
      "Status": "SUCCEEDED",
      "AccountGateResult": {
        "Status": "SKIPPED",
        "StatusReason": "Function not found: arn:aws:lambda:eu-west-1:223456789012:function:AWSCloudFormationStackSetAccountGate"
      }
    },
    {
      "Account": "223456789012",
      "Region": "ap-south-1",
      "Status": "CANCELLED",
      "StatusReason": "Cancelled since failure tolerance has exceeded"
    }
  ]
}
```

```

    }
  ]
}

```

Nota: El SKIPPED estado AccountGateResult es el esperado para que las operaciones se realicen correctamente, a menos que se cree una función de puerta de cuenta.

- Para API obtener más información, consulte [ListStackSetOperationResults](#) la Referencia de AWS CLI comandos.

## list-stack-set-operations

En el siguiente ejemplo de código, se muestra cómo usar `list-stack-set-operations`.

### AWS CLI

Para enumerar las operaciones de conjuntos de pilas

En el siguiente `list-stack-set-operations` ejemplo, se muestra la lista de las operaciones más recientes del conjunto de pilas especificado.

```

aws cloudformation list-stack-set-operations \
  --stack-set-name my-stack-set

```

Salida:

```

{
  "Summaries": [
    {
      "OperationId": "35d45ebc-ed88-xmpl-ab59-0197a1fc83a0",
      "Action": "UPDATE",
      "Status": "SUCCEEDED",
      "CreationTimestamp": "2019-10-03T16:28:44.377Z",
      "EndTimestamp": "2019-10-03T16:42:08.607Z"
    },
    {
      "OperationId": "891aa98f-7118-xmpl-00b2-00954d1dd0d6",
      "Action": "UPDATE",
      "Status": "FAILED",
      "CreationTimestamp": "2019-10-03T15:43:53.916Z",
      "EndTimestamp": "2019-10-03T15:45:58.925Z"
    }
  ]
}

```

```
    }
  ]
}
```

- Para API obtener más información, consulte [ListStackSetOperations](#) la Referencia de AWS CLI comandos.

## list-stack-sets

En el siguiente ejemplo de código, se muestra cómo usar `list-stack-sets`.

### AWS CLI

Para enumerar conjuntos de pilas

En el siguiente `list-stack-sets` ejemplo, se muestra la lista de conjuntos de pilas de la región y la cuenta actuales.

```
aws cloudformation list-stack-sets
```

Salida:

```
{
  "Summaries": [
    {
      "StackSetName": "enable-config",
      "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
      "Description": "Enable AWS Config",
      "Status": "ACTIVE"
    }
  ]
}
```

- Para API obtener más información, consulte [ListStackSets](#) la Referencia de AWS CLI comandos.

## list-stacks

En el siguiente ejemplo de código, se muestra cómo usar `list-stacks`.

## AWS CLI

Para enumerar las AWS CloudFormation pilas

En el siguiente ejemplo de `list-stacks`, se muestra un resumen de todas las pilas que tienen el estado `CREATE_COMPLETE`:

```
aws cloudformation list-stacks --stack-status-filter CREATE_COMPLETE
```

Salida:

```
[
  {
    "StackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/
myteststack/466df9e0-0dff-08e3-8e2f-5088487c4896",
    "TemplateDescription": "AWS CloudFormation Sample Template S3_Bucket: Sample
template showing how to create a publicly accessible S3 bucket. **WARNING** This
template creates an S3 bucket. You will be billed for the AWS resources used if you
create a stack from this template.",
    "StackStatusReason": null,
    "CreationTime": "2013-08-26T03:27:10.190Z",
    "StackName": "myteststack",
    "StackStatus": "CREATE_COMPLETE"
  }
]
```

- Para API obtener más información, consulte [ListStacks](#) la Referencia de AWS CLI comandos.

## `list-type-registrations`

En el siguiente ejemplo de código, se muestra cómo usar `list-type-registrations`.

### AWS CLI

Para enumerar los registros completados de un tipo

En el siguiente `list-type-registrations` ejemplo, se muestra una lista de los registros de tipo completados para el tipo especificado.

```
aws cloudformation list-type-registrations \
  --type RESOURCE \
```

```
--type-name My::Logs::LogGroup \  
--registration-status-filter COMPLETE
```

Salida:

```
{  
  "RegistrationTokenList": [  
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333"  
  ]  
}
```

Para obtener más información, consulte [Uso del CloudFormation registro](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [ListTypeRegistrations](#) la Referencia de AWS CLI comandos.

## list-type-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-type-versions`.

AWS CLI

Para enumerar la versión de una extensión

El siguiente `list-type-versions` ejemplo devuelve información resumida sobre las versiones de una extensión.

```
aws cloudformation list-type-versions \  
  --endpoint https://example.com \  
  --region us-west-2 \  
  --type RESOURCE \  
  --type-name My::Resource::Example \  
  --publisher-id 123456789012
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso del AWS CloudFormation registro](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [ListTypeVersions](#) la Referencia de AWS CLI comandos.

## list-types

En el siguiente ejemplo de código, se muestra cómo usar `list-types`.

### AWS CLI

Para enumerar los tipos de recursos privados de una cuenta

En el siguiente `list-types` ejemplo, se muestra una lista de los tipos de recursos privados actualmente registrados en la AWS cuenta actual.

```
aws cloudformation list-types
```

Salida:

```
{
  "TypeSummaries": [
    {
      "Description": "WordPress blog resource for internal use",
      "LastUpdated": "2019-12-04T18:28:15.059Z",
      "TypeName": "My::WordPress::BlogExample",
      "TypeArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/My-WordPress-BlogExample",
      "DefaultVersionId": "00000005",
      "Type": "RESOURCE"
    },
    {
      "Description": "Customized resource derived from AWS::Logs::LogGroup",
      "LastUpdated": "2019-12-04T18:28:15.059Z",
      "TypeName": "My::Logs::LogGroup",
      "TypeArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/My-Logs-LogGroup",
      "DefaultVersionId": "00000003",
      "Type": "RESOURCE"
    }
  ]
}
```



Para obtener más información, consulte [Uso del CloudFormation registro](#) en la AWS CloudFormation Guía del usuario.

- Para API obtener más información, consulte [ListTypes](#) la Referencia de AWS CLI comandos.

## package

En el siguiente ejemplo de código, se muestra cómo usar package.

### AWS CLI

El siguiente comando exporta una plantilla nombrada `template.json` al cargar artefactos locales en un bucket de S3 `bucket-name` y escribe la plantilla exportada en: `packaged-template.json`

```
aws cloudformation package --template-file /path_to_template/template.json --s3-bucket bucket-name --output-template-file packaged-template.json --use-json
```

- Para API obtener más información, consulte [Package](#) in AWS CLI Command Reference.

## publish-type

En el siguiente ejemplo de código, se muestra cómo usar publish-type.

### AWS CLI

Para publicar una extensión

En el siguiente `publish-type` ejemplo, se publica la extensión especificada en el CloudFormation registro como extensión pública en esta región.

```
aws cloudformation publish-type \  
  --region us-west-2 \  
  --type RESOURCE \  
  --type-name Example::Test::1234567890abcdef0
```

Salida:

```
{
```

```
"PublicTypeArn": "arn:aws:cloudformation:us-west-2::type/
resource/000q6TfUovXsEMmgKowxDZLLwqr2QUshd2e75c8c/Example-
Test-1234567890abcdef0/1.0.0"
}
```

Para obtener más información, consulte [Uso del AWS CloudFormation registro](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [PublishType](#) la Referencia de AWS CLI comandos.

## register-publisher

En el siguiente ejemplo de código, se muestra cómo usar `register-publisher`.

### AWS CLI

Para registrar un editor

En el siguiente `register-publisher` ejemplo, se registra un editor y se acepta el parámetro de términos y condiciones.

```
aws cloudformation register-publisher \
  --region us-west-2 \
  --accept-terms-and-conditions
```

Salida:

```
{
  "PublisherId": "000q6TfUovXsEMmgKowxDZLLwqr2QUshd2e75c8c"
}
```

Para obtener más información, consulte [Uso del AWS CloudFormation registro](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [RegisterPublisher](#) la Referencia de AWS CLI comandos.

## register-type

En el siguiente ejemplo de código, se muestra cómo usar `register-type`.

## AWS CLI

Para registrar un tipo de recurso

El siguiente `register-type` ejemplo registra el tipo de recurso especificado como un tipo de recurso privado en la cuenta del usuario.

```
aws cloudformation register-type \  
  --type-name My::Organization::ResourceName \  
  --schema-handler-package s3://bucket_name/my-organization-resource_name.zip \  
  --type RESOURCE
```

Salida:

```
{  
  "RegistrationToken": "f5525280-104e-4d35-bef5-8f1f1example"  
}
```

Para obtener más información, consulte [Registro de proveedores de recursos](#) en la Guía del usuario de la interfaz de línea de CloudFormation comandos para el desarrollo de tipos.

- Para API obtener más información, consulte [RegisterType](#) la Referencia de AWS CLI comandos.

## set-stack-policy

En el siguiente ejemplo de código, se muestra cómo usar `set-stack-policy`.

### AWS CLI

Para aplicar una política de apilamiento

En el siguiente `set-stack-policy` ejemplo, se deshabilitan las actualizaciones del recurso especificado en la pila especificada. `stack-policy.json` es un JSON documento que define las operaciones permitidas en los recursos de la pila.

```
aws cloudformation set-stack-policy \  
  --stack-name my-stack \  
  --stack-policy-body file://stack-policy.json
```

Salida:

```
{
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : "Update:*",
      "Principal": "*",
      "Resource" : "*"
    },
    {
      "Effect" : "Deny",
      "Action" : "Update:*",
      "Principal": "*",
      "Resource" : "LogicalResourceId/bucket"
    }
  ]
}
```

- Para API obtener más información, consulte [SetStackPolicy](#) la Referencia de AWS CLI comandos.

## set-type-configuration

En el siguiente ejemplo de código, se muestra cómo usar `set-type-configuration`.

### AWS CLI

Para configurar los datos

En el siguiente `set-type-configuration` ejemplo, se especifican los datos de configuración de una CloudFormation extensión registrada en la cuenta y región determinadas.

```
aws cloudformation set-type-configuration \
  --region us-west-2 \
  --type RESOURCE \
  --type-name Example::Test::Type \
  --configuration-alias default \
  --configuration "{\"CredentialKey\": \"testUserCredential\"}"
```

Salida:

```
{
```

```
"ConfigurationArn": "arn:aws:cloudformation:us-west-2:123456789012:type-configuration/resource/Example-Test-Type/default"
}
```

Para obtener más información, consulte [Uso del AWS CloudFormation registro](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [SetTypeConfiguration](#) la Referencia de AWS CLI comandos.

## set-type-default-version

En el siguiente ejemplo de código, se muestra cómo usar `set-type-default-version`.

### AWS CLI

Para establecer la versión por defecto de un tipo

En el siguiente `set-type-default-version` ejemplo, se establece la versión de tipo especificada que se utilizará como predeterminada para este tipo.

```
aws cloudformation set-type-default-version \
  --type RESOURCE \
  --type-name My::Logs::LogGroup \
  --version-id 00000003
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso del CloudFormation registro](#) en la AWS CloudFormation Guía del usuario.

- Para API obtener más información, consulte [SetTypeDefaultVersion](#) la Referencia de AWS CLI comandos.

## signal-resource

En el siguiente ejemplo de código, se muestra cómo usar `signal-resource`.

### AWS CLI

Para señalar un recurso

En el siguiente `signal-resource` ejemplo, `success` se indica que se debe cumplir la condición de espera `MyWaitCondition` indicada en la pila denominada `my-stack`.

```
aws cloudformation signal-resource \  
  --stack-name my-stack \  
  --logical-resource-id MyWaitCondition \  
  --unique-id 1234 \  
  --status SUCCESS
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SignalResource](#) la Referencia de AWS CLI comandos.

## stop-stack-set-operation

En el siguiente ejemplo de código, se muestra cómo usar `stop-stack-set-operation`.

AWS CLI

Para detener una operación de conjunto de pilas

El siguiente `stop-stack-set-operation` ejemplo detiene una operación de actualización en curso en el conjunto de pilas especificado.

```
aws cloudformation stop-stack-set-operation \  
  --stack-set-name my-stack-set \  
  --operation-id 1261cd27-490b-xmpl-ab42-793a896c69e6
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte la Referencia [StopStackSetOperation](#) de AWS CLI comandos.

## test-type

En el siguiente ejemplo de código, se muestra cómo usar `test-type`.

AWS CLI

Para probar una extensión

En el siguiente `test-type` ejemplo, se comprueba una extensión registrada para asegurarse de que cumple todos los requisitos necesarios para su publicación en el CloudFormation registro.

```
aws cloudformation test-type \  
  --arn arn:aws:cloudformation:us-west-2:123456789012:type/resource/Sample-Test-Resource123/00000001
```

Salida:

```
{  
  "TypeVersionArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/  
Sample-Test-Resource123/00000001"  
}
```

Para obtener más información, consulte [Uso del AWS CloudFormation registro](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [TestType](#) la Referencia de AWS CLI comandos.

## update-stack-instances

En el siguiente ejemplo de código, se muestra cómo usar `update-stack-instances`.

### AWS CLI

Para actualizar las instancias de pila

En el siguiente `update-stack-instances` ejemplo, se vuelve a intentar actualizar las instancias de pila de dos cuentas de dos regiones con la configuración más reciente. La configuración de tolerancia a errores especificada garantiza que la actualización se intente realizar en todas las cuentas y regiones, incluso si algunas pilas no se pueden actualizar.

```
aws cloudformation update-stack-instances \  
  --stack-set-name my-stack-set \  
  --accounts 123456789012 567890123456 \  
  --regions us-east-1 us-west-2 \  
  --operation-preferences FailureToleranceCount=3
```

Salida:

```
{
```

```
"OperationId": "103ebdf2-21ea-xmpl-8892-de5e30733132"  
}
```

- Para API obtener más información, consulte [UpdateStackInstances](#) la Referencia de AWS CLI comandos.

## update-stack-set

En el siguiente ejemplo de código, se muestra cómo usar `update-stack-set`.

### AWS CLI

Para actualizar un conjunto de pilas

En el siguiente `update-stack-set` ejemplo, se agrega una etiqueta con el nombre de la clave `Owner` y un valor de `IT` a las instancias de pila del conjunto de pilas especificado.

```
aws cloudformation update-stack-set \  
  --stack-set-name my-stack-set \  
  --use-previous-template \  
  --tags Key=Owner, Value=IT
```

Salida:

```
{  
  "OperationId": "e2b60321-6cab-xmpl-bde7-530c6f47950e"  
}
```

- Para API obtener más información, consulte [UpdateStackSet](#) la Referencia de AWS CLI comandos.

## update-stack

En el siguiente ejemplo de código, se muestra cómo usar `update-stack`.

### AWS CLI

Para actualizar las AWS CloudFormation pilas

En el siguiente ejemplo de `update-stack`, se actualiza la plantilla y los parámetros de entrada de la pila `mystack`:



```
aws cloudformation update-stack --stack-name mystack --  
template-url https://s3.amazonaws.com/sample/updated.template --  
parameters ParameterKey=KeyPairName,ParameterValue=SampleKeyPair  
ParameterKey=SubnetIDs,ParameterValue=SampleSubnetID1\\,SampleSubnetID2
```

En el siguiente ejemplo de `update-stack`, se actualiza únicamente el valor de parámetro `SubnetIDs` para la pila `mystack`. Si no se especifica un valor de parámetro, se utilizará el valor predeterminado que esté especificado en la plantilla:

```
aws cloudformation update-stack --stack-name mystack --  
template-url https://s3.amazonaws.com/sample/updated.template  
--parameters ParameterKey=KeyPairName,UsePreviousValue=true  
ParameterKey=SubnetIDs,ParameterValue=SampleSubnetID1\\,UpdatedSampleSubnetID2
```

En el siguiente ejemplo de `update-stack`, se agregan dos temas de notificaciones de pila en la pila `mystack`:

```
aws cloudformation update-stack --stack-name mystack --use-previous-template --  
notification-arns "arn:aws:sns:use-east-1:123456789012:mytopic1" "arn:aws:sns:us-  
east-1:123456789012:mytopic2"
```

Para obtener más información, consulta [las actualizaciones de las AWS CloudFormation pilas](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [UpdateStack](#) la Referencia de AWS CLI comandos.

## update-termination-protection

En el siguiente ejemplo de código, se muestra cómo usar `update-termination-protection`.

### AWS CLI

Para habilitar la protección de terminación

El siguiente `update-termination-protection` ejemplo habilita la protección de terminación en la pila especificada.

```
aws cloudformation update-termination-protection \  
--stack-name my-stack \  
--
```

**--enable-termination-protection**

Salida:

```
{
  "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/
d0a825a0-e4cd-xmpl-b9fb-061c69e99204"
}
```

- Para API obtener más información, consulte [UpdateTerminationProtection](#) la Referencia de AWS CLI comandos.

**validate-template**

En el siguiente ejemplo de código, se muestra cómo usar `validate-template`.

**AWS CLI**

Para validar una AWS CloudFormation plantilla

En el siguiente ejemplo de `validate-template`, se valida la plantilla `sampletemplate.json`:

```
aws cloudformation validate-template --template-body file://sampletemplate.json
```

Salida:

```
{
  "Description": "AWS CloudFormation Sample Template S3_Bucket: Sample template
showing how to create a publicly accessible S3 bucket. **WARNING** This template
creates an S3 bucket. You will be billed for the AWS resources used if you create a
stack from this template.",
  "Parameters": [],
  "Capabilities": []
}
```

Para obtener más información, consulte [Trabajar con AWS CloudFormation plantillas](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [ValidateTemplate](#) la Referencia de AWS CLI comandos.

## CloudFront ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with CloudFront.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-cloud-front-origin-access-identity**

En el siguiente ejemplo de código, se muestra cómo usar `create-cloud-front-origin-access-identity`.

#### AWS CLI

Para crear una identidad de acceso de CloudFront origen

El siguiente ejemplo crea una identidad de acceso de CloudFront origen (OAI) proporcionando la OAI configuración como argumento de línea de comandos:

```
aws cloudfront create-cloud-front-origin-access-identity \  
  --cloud-front-origin-access-identity-config \  
    CallerReference="cli-example",Comment="Example OAI"
```

Puede lograr lo mismo proporcionando la OAI configuración en un JSON archivo, como se muestra en el siguiente ejemplo:

```
aws cloudfront create-cloud-front-origin-access-identity \  
  --cloud-front-origin-access-identity-config file://OAI-config.json
```

El archivo `OAI-config.json` es un JSON documento del directorio actual que contiene lo siguiente:

```
{
  "CallerReference": "cli-example",
  "Comment": "Example OAI"
}
```

Tanto si proporciona la OAI configuración con un argumento de línea de comandos como con un JSON archivo, el resultado es el mismo:

```
{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/origin-access-identity/cloudfront/E74FTE3AEXAMPLE",
  "ETag": "E2QWRUHEXAMPLE",
  "CloudFrontOriginAccessIdentity": {
    "Id": "E74FTE3AEXAMPLE",
    "S3CanonicalUserId":
"cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
    "CloudFrontOriginAccessIdentityConfig": {
      "CallerReference": "cli-example",
      "Comment": "Example OAI"
    }
  }
}
```

- Para API obtener más información, consulte [CreateCloudFrontOriginAccessIdentity](#) la Referencia de AWS CLI comandos.

## **create-distribution-with-tags**

En el siguiente ejemplo de código, se muestra cómo usar `create-distribution-with-tags`.

### AWS CLI

Para crear una CloudFront distribución con etiquetas

El siguiente ejemplo crea una distribución con dos etiquetas proporcionando la configuración de distribución y las etiquetas en un JSON archivo denominado `dist-config-with-tags.json`:

```
aws cloudfront create-distribution-with-tags \
```

```
--distribution-config-with-tags file://dist-config-with-tags.json
```

El archivo `dist-config-with-tags.json` es un JSON documento de la carpeta actual que contiene lo siguiente. Observe el `Tags` objeto situado en la parte superior del archivo, que contiene dos etiquetas:

Name = ExampleDistributionProject = ExampleProject

```
{
  "Tags": {
    "Items": [
      {
        "Key": "Name",
        "Value": "ExampleDistribution"
      },
      {
        "Key": "Project",
        "Value": "ExampleProject"
      }
    ]
  },
  "DistributionConfig": {
    "CallerReference": "cli-example",
    "Aliases": {
      "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
      "Quantity": 1,
      "Items": [
        {
          "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
          "DomainName": "awsexamplebucket.s3.amazonaws.com",
          "OriginPath": "",
          "CustomHeaders": {
            "Quantity": 0
          },
          "S3OriginConfig": {
            "OriginAccessIdentity": ""
          }
        }
      ]
    }
  },
}
```

```
"OriginGroups": {
  "Quantity": 0
},
"DefaultCacheBehavior": {
  "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
  "ForwardedValues": {
    "QueryString": false,
    "Cookies": {
      "Forward": "none"
    },
  },
  "Headers": {
    "Quantity": 0
  },
  "QueryStringCacheKeys": {
    "Quantity": 0
  }
},
"TrustedSigners": {
  "Enabled": false,
  "Quantity": 0
},
"ViewerProtocolPolicy": "allow-all",
"MinTTL": 0,
"AllowedMethods": {
  "Quantity": 2,
  "Items": [
    "HEAD",
    "GET"
  ],
  "CachedMethods": {
    "Quantity": 2,
    "Items": [
      "HEAD",
      "GET"
    ]
  }
},
"SmoothStreaming": false,
"DefaultTTL": 86400,
"MaxTTL": 31536000,
"Compress": false,
"LambdaFunctionAssociations": {
  "Quantity": 0
},
}
```

```

        "FieldLevelEncryptionId": ""
    },
    "CacheBehaviors": {
        "Quantity": 0
    },
    "CustomErrorResponses": {
        "Quantity": 0
    },
    "Comment": "",
    "Logging": {
        "Enabled": false,
        "IncludeCookies": false,
        "Bucket": "",
        "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
}
}

```

Salida:

```

{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/EDFDVBD6EXAMPLE",
  "ETag": "E2QWRUHEXAMPLE",
  "Distribution": {
    "Id": "EDFDVBD6EXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",

```

```
"Status": "InProgress",
"LastModifiedTime": "2019-12-04T23:35:41.433Z",
"InProgressInvalidationBatches": 0,
"DomainName": "d111111abcdef8.cloudfront.net",
"ActiveTrustedSigners": {
  "Enabled": false,
  "Quantity": 0
},
"DistributionConfig": {
  "CallerReference": "cli-example",
  "Aliases": {
    "Quantity": 0
  },
  "DefaultRootObject": "index.html",
  "Origins": {
    "Quantity": 1,
    "Items": [
      {
        "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
        "DomainName": "awsexamplebucket.s3.amazonaws.com",
        "OriginPath": "",
        "CustomHeaders": {
          "Quantity": 0
        },
        "S3OriginConfig": {
          "OriginAccessIdentity": ""
        }
      }
    ]
  },
  "OriginGroups": {
    "Quantity": 0
  },
  "DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
    "ForwardedValues": {
      "QueryString": false,
      "Cookies": {
        "Forward": "none"
      },
      "Headers": {
        "Quantity": 0
      },
      "QueryStringCacheKeys": {
```



```
        "Quantity": 0
      }
    },
    "TrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ],
      "CachedMethods": {
        "Quantity": 2,
        "Items": [
          "HEAD",
          "GET"
        ]
      }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
      "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
  },
  "CacheBehaviors": {
    "Quantity": 0
  },
  "CustomErrorResponses": {
    "Quantity": 0
  },
  "Comment": "",
  "Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
  }
}
```

```

    },
    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
      "CloudFrontDefaultCertificate": true,
      "MinimumProtocolVersion": "TLSv1",
      "CertificateSource": "cloudfront"
    },
    "Restrictions": {
      "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
      }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
  }
}
}
}

```

- Para API obtener más información, consulte [CreateDistributionWithTags](#) la Referencia de AWS CLI comandos.

## create-distribution

En el siguiente ejemplo de código, se muestra cómo usar `create-distribution`.

### AWS CLI

Para crear una CloudFront distribución

En el siguiente ejemplo se crea una distribución para un bucket de S3 denominado `awsexamplebucket` y también se especifica `index.html` como objeto raíz predeterminado mediante argumentos de línea de comandos:

```

aws cloudfront create-distribution \
  --origin-domain-name awsexamplebucket.s3.amazonaws.com \
  --default-root-object index.html

```

En lugar de utilizar argumentos de línea de comandos, puede proporcionar la configuración de distribución en un JSON archivo, como se muestra en el siguiente ejemplo:

```
aws cloudfront create-distribution \  
--distribution-config file://dist-config.json
```

El archivo `dist-config.json` es un JSON documento de la carpeta actual que contiene lo siguiente:

```
{  
  "CallerReference": "cli-example",  
  "Aliases": {  
    "Quantity": 0  
  },  
  "DefaultRootObject": "index.html",  
  "Origins": {  
    "Quantity": 1,  
    "Items": [  
      {  
        "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",  
        "DomainName": "awsexamplebucket.s3.amazonaws.com",  
        "OriginPath": "",  
        "CustomHeaders": {  
          "Quantity": 0  
        },  
        "S3OriginConfig": {  
          "OriginAccessIdentity": ""  
        }  
      }  
    ]  
  },  
  "OriginGroups": {  
    "Quantity": 0  
  },  
  "DefaultCacheBehavior": {  
    "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",  
    "ForwardedValues": {  
      "QueryString": false,  
      "Cookies": {  
        "Forward": "none"  
      },  
    },  
    "Headers": {  
      "Quantity": 0  
    },  
    "QueryStringCacheKeys": {  
      "Quantity": 0  
    }  
  }  
}
```

```
    }
  },
  "TrustedSigners": {
    "Enabled": false,
    "Quantity": 0
  },
  "ViewerProtocolPolicy": "allow-all",
  "MinTTL": 0,
  "AllowedMethods": {
    "Quantity": 2,
    "Items": [
      "HEAD",
      "GET"
    ],
    "CachedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ]
    }
  },
  "SmoothStreaming": false,
  "DefaultTTL": 86400,
  "MaxTTL": 31536000,
  "Compress": false,
  "LambdaFunctionAssociations": {
    "Quantity": 0
  },
  "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
  "Quantity": 0
},
"CustomErrorResponses": {
  "Quantity": 0
},
"Comment": "",
"Logging": {
  "Enabled": false,
  "IncludeCookies": false,
  "Bucket": "",
  "Prefix": ""
},
},
```

```

    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
}

```

Tanto si proporciona la información de distribución con un argumento de línea de comandos como con un JSON archivo, el resultado es el mismo:

```

{
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/
EMLARXS9EXAMPLE",
    "ETag": "E9LHASXEXAMPLE",
    "Distribution": {
        "Id": "EMLARXS9EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/EMLARXS9EXAMPLE",
        "Status": "InProgress",
        "LastModifiedTime": "2019-11-22T00:55:15.705Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
    },
    "DistributionConfig": {
        "CallerReference": "cli-example",
        "Aliases": {
            "Quantity": 0
        },
        "DefaultRootObject": "index.html",
        "Origins": {

```

```
"Quantity": 1,
"Items": [
  {
    "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
    "DomainName": "awsexamplebucket.s3.amazonaws.com",
    "OriginPath": "",
    "CustomHeaders": {
      "Quantity": 0
    },
    "S3OriginConfig": {
      "OriginAccessIdentity": ""
    }
  }
],
"OriginGroups": {
  "Quantity": 0
},
"DefaultCacheBehavior": {
  "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
  "ForwardedValues": {
    "QueryString": false,
    "Cookies": {
      "Forward": "none"
    },
    "Headers": {
      "Quantity": 0
    },
    "QueryStringCacheKeys": {
      "Quantity": 0
    }
  },
  "TrustedSigners": {
    "Enabled": false,
    "Quantity": 0
  },
  "ViewerProtocolPolicy": "allow-all",
  "MinTTL": 0,
  "AllowedMethods": {
    "Quantity": 2,
    "Items": [
      "HEAD",
      "GET"
    ]
  },

```

```
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
        "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponses": {
    "Quantity": 0
},
"Comment": "",
"Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
},
"Restrictions": {
    "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
    }
},
"WebACLId": "",
```

```

        "HttpVersion": "http2",
        "IsIPV6Enabled": true
    }
}
}

```

- Para API obtener más información, consulte [CreateDistribution](#) la Referencia de AWS CLI comandos.

## create-field-level-encryption-config

En el siguiente ejemplo de código, se muestra cómo usar `create-field-level-encryption-config`.

### AWS CLI

Para crear una configuración de CloudFront cifrado a nivel de campo

El siguiente ejemplo crea una configuración de cifrado a nivel de campo proporcionando los parámetros de configuración en un archivo denominado `file-config.json`. Para poder crear una configuración de cifrado a nivel de campo, debe tener un perfil de cifrado a nivel de campo. Para crear un perfil, consulte el comando `-profile`. `create-field-level-encryption`

Para obtener más información sobre el cifrado a CloudFront nivel de campo, consulte [Uso del cifrado a nivel de campo para proteger los datos confidenciales en](#) la Guía para desarrolladores de Amazon. CloudFront

```

aws cloudfront create-field-level-encryption-config \
  --field-level-encryption-config file://file-config.json

```

El archivo `file-config.json` es un JSON documento de la carpeta actual que contiene lo siguiente:

```

{
  "CallerReference": "cli-example",
  "Comment": "Example FLE configuration",
  "QueryArgProfileConfig": {
    "ForwardWhenQueryArgProfileIsUnknown": true,
    "QueryArgProfiles": {
      "Quantity": 0
    }
  }
}

```



```

},
"ContentTypeProfileConfig": {
  "ForwardWhenContentTypeIsUnknown": true,
  "ContentTypeProfiles": {
    "Quantity": 1,
    "Items": [
      {
        "Format": "URLEncoded",
        "ProfileId": "P280MFCLSY0CVU",
        "ContentType": "application/x-www-form-urlencoded"
      }
    ]
  }
}
}
}

```

**Salida:**

```

{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/field-level-encryption/
C3KM2WVD605UAY",
  "ETag": "E2P4Z4VU7TY5SG",
  "FieldLevelEncryption": {
    "Id": "C3KM2WVD605UAY",
    "LastModifiedTime": "2019-12-10T21:30:18.974Z",
    "FieldLevelEncryptionConfig": {
      "CallerReference": "cli-example",
      "Comment": "Example FLE configuration",
      "QueryArgProfileConfig": {
        "ForwardWhenQueryArgProfileIsUnknown": true,
        "QueryArgProfiles": {
          "Quantity": 0,
          "Items": []
        }
      }
    }
  },
  "ContentTypeProfileConfig": {
    "ForwardWhenContentTypeIsUnknown": true,
    "ContentTypeProfiles": {
      "Quantity": 1,
      "Items": [
        {
          "Format": "URLEncoded",
          "ProfileId": "P280MFCLSY0CVU",

```

```
        "ContentType": "application/x-www-form-urlencoded"
      }
    ]
  }
}
```

- Para API obtener más información, consulte [CreateFieldLevelEncryptionConfig](#) la Referencia de AWS CLI comandos.

## create-field-level-encryption-profile

En el siguiente ejemplo de código, se muestra cómo usar `create-field-level-encryption-profile`.

### AWS CLI

Para crear un perfil de CloudFront cifrado a nivel de campo

El siguiente ejemplo crea un perfil de cifrado a nivel de campo proporcionando los parámetros de un archivo denominado `fle-profile-config.json`. Para poder crear un perfil de cifrado a nivel de campo, debe disponer de una clave pública. CloudFront Para crear una clave CloudFront pública, consulte el `create-public-key` comando.

Para obtener más información sobre el cifrado a CloudFront nivel de campo, consulte [Uso del cifrado a nivel de campo para proteger los datos confidenciales en](#) la Guía para desarrolladores de Amazon. CloudFront

```
aws cloudfront create-field-level-encryption-profile \
  --field-level-encryption-profile-config file://fle-profile-config.json
```

El archivo `fle-profile-config.json` es un JSON documento de la carpeta actual que contiene lo siguiente:

```
{
  "Name": "ExampleFLEProfile",
  "CallerReference": "cli-example",
  "Comment": "FLE profile for AWS CLI example",
  "EncryptionEntities": {
```

```

    "Quantity": 1,
    "Items": [
      {
        "PublicKeyId": "K2K8NC4HVFE3M0",
        "ProviderId": "ExampleFLEProvider",
        "FieldPatterns": {
          "Quantity": 1,
          "Items": [
            "ExampleSensitiveField"
          ]
        }
      ]
    ]
  }
}

```

**Salida:**

```

{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/field-level-encryption-
profile/PPK0UOSIF5WSV",
  "ETag": "E2QWRUHEXAMPLE",
  "FieldLevelEncryptionProfile": {
    "Id": "PPK0UOSIF5WSV",
    "LastModifiedTime": "2019-12-10T01:03:16.537Z",
    "FieldLevelEncryptionProfileConfig": {
      "Name": "ExampleFLEProfile",
      "CallerReference": "cli-example",
      "Comment": "FLE profile for AWS CLI example",
      "EncryptionEntities": {
        "Quantity": 1,
        "Items": [
          {
            "PublicKeyId": "K2K8NC4HVFE3M0",
            "ProviderId": "ExampleFLEProvider",
            "FieldPatterns": {
              "Quantity": 1,
              "Items": [
                "ExampleSensitiveField"
              ]
            }
          ]
        ]
      }
    }
  ]
}

```

```

    }
  }
}

```

- Para API obtener más información, consulte [CreateFieldLevelEncryptionProfile](#) la Referencia de AWS CLI comandos.

## create-invalidatio

En el siguiente ejemplo de código, se muestra cómo usar `create-invalidatio`.

### AWS CLI

Para crear una invalidación para una distribución CloudFront

El siguiente `create-invalidatio` ejemplo crea una invalidación para los archivos especificados de la distribución especificada CloudFront :

```

aws cloudfront create-invalidatio \
  --distribution-id EDFDVBD6EXAMPLE \
  --paths "/example-path/example-file.jpg" "/example-path/example-file2.png"

```

Salida:

```

{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/EDFDVBD6EXAMPLE/invalidation/I1JLWSDAP8FU89",
  "Invalidation": {
    "Id": "I1JLWSDAP8FU89",
    "Status": "InProgress",
    "CreateTime": "2019-12-05T18:24:51.407Z",
    "InvalidationBatch": {
      "Paths": {
        "Quantity": 2,
        "Items": [
          "/example-path/example-file2.png",
          "/example-path/example-file.jpg"
        ]
      }
    },
    "CallerReference": "cli-1575570291-670203"
  }
}

```

```
}  
}
```

En el ejemplo anterior, generó AWS CLI automáticamente un `aleatorioCallerReference`. Para especificar los suyos propios `CallerReference` o para evitar pasar los parámetros de invalidación como argumentos de línea de comandos, puede utilizar un JSON archivo. El siguiente ejemplo crea una invalidación para dos archivos, proporcionando los parámetros de invalidación en un JSON archivo denominado: `inv-batch.json`

```
aws cloudfront create-invalidation \  
  --distribution-id EDFDVBD6EXAMPLE \  
  --invalidation-batch file://inv-batch.json
```

Contenidos de `inv-batch.json`:

```
{  
  "Paths": {  
    "Quantity": 2,  
    "Items": [  
      "/example-path/example-file.jpg",  
      "/example-path/example-file2.png"  
    ]  
  },  
  "CallerReference": "cli-example"  
}
```

Salida:

```
{  
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/  
EDFDVBD6EXAMPLE/invalidation/I2J0I21PCUY0IK",  
  "Invalidation": {  
    "Id": "I2J0I21PCUY0IK",  
    "Status": "InProgress",  
    "CreateTime": "2019-12-05T18:40:49.413Z",  
    "InvalidationBatch": {  
      "Paths": {  
        "Quantity": 2,  
        "Items": [  
          "/example-path/example-file.jpg",  
          "/example-path/example-file2.png"  
        ]  
      }  
    }  
  }  
}
```

```

    },
    "CallerReference": "cli-example"
  }
}
}

```

- Para API obtener más información, consulte la Referencia [CreateInvalidation](#) de AWS CLI comandos.

## create-public-key

En el siguiente ejemplo de código, se muestra cómo usar `create-public-key`.

### AWS CLI

Para crear una clave CloudFront pública

En el siguiente ejemplo, se crea una clave CloudFront pública proporcionando los parámetros de un JSON archivo denominado `pub-key-config.json`. Para poder usar este comando, debe tener una clave pública PEM codificada. Para obtener más información, consulta [Crear un par de RSA claves](#) en la Guía para CloudFront desarrolladores de Amazon.

```

aws cloudfront create-public-key \
  --public-key-config file://pub-key-config.json

```

El archivo `pub-key-config.json` es un JSON documento de la carpeta actual que contiene lo siguiente. Tenga en cuenta que la clave pública está codificada en PEM formato.

```

{
  "CallerReference": "cli-example",
  "Name": "ExampleKey",
  "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxPMbCA2Ks01nd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1U0WcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAz03ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95y1UQd9LfYTBzVZYG2tSesplc0kjM3\n2Uu
+oMwxQAw1NINnSLPinMVsutJy6Zq1V3McwNWe4T+STGtWhrPNqJEn45sIcCx4\nnq
+kGZ2NQ0FyIyT2eiLK0X5RgB/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nnrwIDAQAB\n-----END
PUBLIC KEY-----\n",
  "Comment": "example public key"
}

```

**Salida:**

```
{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/public-key/
KDFB19YGCR002",
  "ETag": "E2QWRUHEXAMPLE",
  "PublicKey": {
    "Id": "KDFB19YGCR002",
    "CreatedTime": "2019-12-05T18:51:43.781Z",
    "PublicKeyConfig": {
      "CallerReference": "cli-example",
      "Name": "ExampleKey",
      "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxPmBCA2Ks01nd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLae/1U0WcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBaz03ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95y1UQd9LfYTBzVZYG2tSesplc0kjM3\n2Uu
+oMwxQAw1NINnSLPinMVsutJy6Zq1V3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nnq
+kGZ2NQ0FyIyT2eiLK0X5RgB/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nnrwIDAQAB\n-----END
PUBLIC KEY-----\n",
      "Comment": "example public key"
    }
  }
}
```

- Para API obtener más información, consulte [CreatePublicKey](#) la Referencia de AWS CLI comandos.

**delete-cloud-front-origin-access-identity**

En el siguiente ejemplo de código, se muestra cómo usar `delete-cloud-front-origin-access-identity`.

**AWS CLI**

Para eliminar una identidad de acceso de CloudFront origen

En el siguiente ejemplo, se elimina la identidad de acceso de origen (OAI) junto con el IDE74FTE3AEXAMPLE. Para eliminar un OAI, debes tener el OAI ID y ETag. El OAI ID se devuelve en el resultado de los comandos `create-cloud-front-origin-access-identity` y `list-cloud-front-origin-access-identity`. Para obtenerla ETag, utilice el comando `-access-identity` o `-. get-cloud-front-origin`

get-cloud-front-origin access-identity-config Usa la `--if-match` opción para proporcionar los. OAI ETag

```
aws cloudfront delete-cloud-front-origin-access-identity \  
  --id E74FTE3AEXAMPLE \  
  --if-match E2QWRUHEXAMPLE
```

Cuando tenga éxito, este comando no tiene salida.

- Para API obtener más información, consulte [DeleteCloudFrontOriginAccessIdentity](#) la Referencia de AWS CLI comandos.

## delete-distribution

En el siguiente ejemplo de código, se muestra cómo usar `delete-distribution`.

### AWS CLI

Para eliminar una CloudFront distribución

En el siguiente ejemplo, se elimina la CloudFront distribución con el ID. EDFDVBD6EXAMPLE. Antes de eliminar una distribución, debe deshabilitarla. Para deshabilitar una distribución, utilice el comando `update-distribution`. Para obtener más información, consulte los ejemplos de `update-distribution`.

Cuando una distribución está deshabilitada, puede eliminarla. Para eliminar una distribución, debe usar la opción `--if-match` para proporcionar la ETag de la distribución. Para obtenerla ETag, utilice el comando `get-distribution` or. `get-distribution-config`

```
aws cloudfront delete-distribution \  
  --id EDFDVBD6EXAMPLE \  
  --if-match E2QWRUHEXAMPLE
```

Cuando tenga éxito, este comando no tiene salida.

- Para API obtener más información, consulte la Referencia [DeleteDistribution](#) de AWS CLI comandos.



## delete-field-level-encryption-config

En el siguiente ejemplo de código, se muestra cómo usar `delete-field-level-encryption-config`.

### AWS CLI

Para eliminar una configuración de CloudFront cifrado a nivel de campo

En el siguiente ejemplo, se elimina la configuración de cifrado a CloudFront nivel de campo con el ID. `C3KM2WVD605UAY` Para eliminar una configuración de cifrado a nivel de campo, debe tener su ID y. ETag El ID se devuelve en el resultado de los comandos `create-field-level-encryption-config` y `list-field-level-encryption-configs`. Para obtenerlos ETag, utilice el comando `get-field-level-encryption` or `get-field-level-encryption-config`. Utilice la `--if-match` opción para proporcionar las configuraciones. ETag

```
aws cloudfront delete-field-level-encryption-config \  
  --id C3KM2WVD605UAY \  
  --if-match E26M4BIAV81ZF6
```

Cuando tenga éxito, este comando no tiene salida.

- Para API obtener más información, consulte [DeleteFieldLevelEncryptionConfig](#) la Referencia de AWS CLI comandos.

## delete-field-level-encryption-profile

En el siguiente ejemplo de código, se muestra cómo usar `delete-field-level-encryption-profile`.

### AWS CLI

Para eliminar un perfil de CloudFront cifrado a nivel de campo

En el siguiente ejemplo, se elimina el perfil de cifrado a CloudFront nivel de campo con el ID. `PPK0U0SIF5WSV` Para eliminar un perfil de cifrado a nivel de campo, debe tener su ID y. ETag El ID se devuelve en el resultado de los comandos `create-field-level-encryption-profile` y `list-field-level-encryption-profiles`. Para obtenerlo ETag, utilice el comando `get-field-level-encryption-profile` o `get-field-level-encryption-profile-config`. Utilice la `--if-match` opción para proporcionar los perfiles. ETag

```
aws cloudfront delete-field-level-encryption-profile \  
  --id PPK0UOSIF5WSV \  
  --if-match EJETYFJ9CL66D
```

Cuando tenga éxito, este comando no tiene salida.

- Para API obtener más información, consulte [DeleteFieldLevelEncryptionProfile](#) la Referencia de AWS CLI comandos.

## delete-public-key

En el siguiente ejemplo de código, se muestra cómo usar `delete-public-key`.

### AWS CLI

Para eliminar una clave CloudFront pública

En el siguiente ejemplo, se elimina la clave CloudFront pública con el `IDKDFB19YGCR002`. Para eliminar una clave pública, debe tener su ID y ETag. El ID se devuelve en la salida de los `list-public-keys` comandos `create-public-key` y. Para obtener el ETag, utilice el `get-public-key-config` comando `get-public-key` o. Utilice la `--if-match` opción para proporcionar las claves públicas ETag.

```
aws cloudfront delete-public-key \  
  --id KDFB19YGCR002 \  
  --if-match E2QWRUHEXAMPLE
```

Cuando tenga éxito, este comando no tiene salida.

- Para API obtener más información, consulte [DeletePublicKey](#) la Referencia de AWS CLI comandos.

## get-cloud-front-origin-access-identity-config

En el siguiente ejemplo de código, se muestra cómo usar `get-cloud-front-origin-access-identity-config`.

### AWS CLI

Para obtener una configuración de identidad de acceso a CloudFront Origin

En el siguiente ejemplo, se obtienen metadatos sobre la identidad de acceso de CloudFront origen (OAI) con el IDE74FTE3AEXAMPLE, incluido el suyoETag. El OAI ID se devuelve en el resultado de los comandos `create-cloud-front-origin -access-identity` y `list-cloud-front-origin -access-identity`.

```
aws cloudfront get-cloud-front-origin-access-identity-config --id E74FTE3AEXAMPLE
```

Salida:

```
{
  "ETag": "E2QWRUHEXAMPLE",
  "CloudFrontOriginAccessIdentityConfig": {
    "CallerReference": "cli-example",
    "Comment": "Example OAI"
  }
}
```

- Para obtener API más información, consulte la Referencia de comandos.

[GetCloudFrontOriginAccessIdentityConfig](#) AWS CLI

## get-cloud-front-origin-access-identity

En el siguiente ejemplo de código, se muestra cómo usar `get-cloud-front-origin-access-identity`.

AWS CLI

Para obtener una identidad de acceso de CloudFront origen

En el siguiente ejemplo, se obtiene la identidad de acceso de CloudFront origen (OAI) con el IDE74FTE3AEXAMPLE, incluido su ETag ID canónico de S3 asociado. El OAI ID se devuelve en el resultado de los comandos `-access-identity` y `create-cloud-front-origin -access-identity`. `list-cloud-front-origin`

```
aws cloudfront get-cloud-front-origin-access-identity --id E74FTE3AEXAMPLE
```

Salida:

```
{
  "ETag": "E2QWRUHEXAMPLE",
```

```

    "CloudFrontOriginAccessIdentity": {
      "Id": "E74FTE3AEXAMPLE",
      "S3CanonicalUserId":
"cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
      "CloudFrontOriginAccessIdentityConfig": {
        "CallerReference": "cli-example",
        "Comment": "Example OAI"
      }
    }
  }
}

```

- Para obtener API más información, consulte la Referencia de comandos.

[GetCloudFrontOriginAccessIdentity](#) AWS CLI

## get-distribution-config

En el siguiente ejemplo de código, se muestra cómo usar `get-distribution-config`.

### AWS CLI

Para obtener una configuración CloudFront de distribución

En el siguiente ejemplo, se obtienen metadatos sobre la CloudFront distribución con el identificador `EDFDVBD6EXAMPLE`, incluido el suyo `ETag`. El ID de distribución se devuelve en los comandos `create-distribution` y `list-distributions`.

```
aws cloudfront get-distribution-config --id EDFDVBD6EXAMPLE
```

Salida:

```

{
  "ETag": "E2QWRUHEXAMPLE",
  "DistributionConfig": {
    "CallerReference": "cli-example",
    "Aliases": {
      "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
      "Quantity": 1,
      "Items": [
        {

```

```
        "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
        "DomainName": "awsexamplebucket.s3.amazonaws.com",
        "OriginPath": "",
        "CustomHeaders": {
            "Quantity": 0
        },
        "S3OriginConfig": {
            "OriginAccessIdentity": ""
        }
    }
]
},
"OriginGroups": {
    "Quantity": 0
},
"DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
    "ForwardedValues": {
        "QueryString": false,
        "Cookies": {
            "Forward": "none"
        },
    },
    "Headers": {
        "Quantity": 0
    },
    "QueryStringCacheKeys": {
        "Quantity": 0
    }
},
"TrustedSigners": {
    "Enabled": false,
    "Quantity": 0
},
"ViewerProtocolPolicy": "allow-all",
"MinTTL": 0,
"AllowedMethods": {
    "Quantity": 2,
    "Items": [
        "HEAD",
        "GET"
    ],
},
"CachedMethods": {
    "Quantity": 2,
    "Items": [
```

```

        "HEAD",
        "GET"
    ]
    }
},
"SmoothStreaming": false,
"DefaultTTL": 86400,
"MaxTTL": 31536000,
"Compress": false,
"LambdaFunctionAssociations": {
    "Quantity": 0
},
"FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponses": {
    "Quantity": 0
},
"Comment": "",
"Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
},
"Restrictions": {
    "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
    }
},
"WebACLId": "",
"HttpVersion": "http2",
"IsIPV6Enabled": true
}

```

```
}
```

- Para API obtener más información, consulte [GetDistributionConfig](#) la Referencia de AWS CLI comandos.

## get-distribution

En el siguiente ejemplo de código, se muestra cómo usar `get-distribution`.

### AWS CLI

Para obtener una CloudFront distribución

En el siguiente ejemplo, se obtiene la CloudFront distribución con el identificador `EDFDVBD6EXAMPLE`, incluido el suyo `ETag`. El ID de distribución se devuelve en los comandos `create-distribution` y `list-distributions`.

```
aws cloudfront get-distribution --id EDFDVBD6EXAMPLE
```

Salida:

```
{
  "ETag": "E2QWRUHEXAMPLE",
  "Distribution": {
    "Id": "EDFDVBD6EXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",
    "Status": "Deployed",
    "LastModifiedTime": "2019-12-04T23:35:41.433Z",
    "InProgressInvalidationBatches": 0,
    "DomainName": "d111111abcdef8.cloudfront.net",
    "ActiveTrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
  },
  "DistributionConfig": {
    "CallerReference": "cli-example",
    "Aliases": {
      "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
      "Quantity": 1,
```

```
    "Items": [
      {
        "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
        "DomainName": "awsexamplebucket.s3.amazonaws.com",
        "OriginPath": "",
        "CustomHeaders": {
          "Quantity": 0
        },
        "S3OriginConfig": {
          "OriginAccessIdentity": ""
        }
      }
    ],
  },
  "OriginGroups": {
    "Quantity": 0
  },
  "DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
    "ForwardedValues": {
      "QueryString": false,
      "Cookies": {
        "Forward": "none"
      },
      "Headers": {
        "Quantity": 0
      },
      "QueryStringCacheKeys": {
        "Quantity": 0
      }
    },
    "TrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ],
      "CachedMethods": {
```



```
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ]
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
        "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponses": {
    "Quantity": 0
},
"Comment": "",
"Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
},
"Restrictions": {
    "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
    }
},
"WebACLId": "",
"HttpVersion": "http2",
```

```

    "IsIPV6Enabled": true
  }
}
}

```

- Para API obtener más información, consulte [GetDistribution](#) la Referencia de AWS CLI comandos.

## get-field-level-encryption-config

En el siguiente ejemplo de código, se muestra cómo usar `get-field-level-encryption-config`.

### AWS CLI

Para obtener metadatos sobre una configuración de cifrado a CloudFront nivel de campo

El siguiente ejemplo obtiene metadatos sobre la configuración de cifrado a CloudFront nivel de campo con el IDC3KM2WVD605UAY, incluido su: ETag

```
aws cloudfront get-field-level-encryption-config --id C3KM2WVD605UAY
```

Salida:

```

{
  "ETag": "E2P4Z4VU7TY5SG",
  "FieldLevelEncryptionConfig": {
    "CallerReference": "cli-example",
    "Comment": "Example FLE configuration",
    "QueryArgProfileConfig": {
      "ForwardWhenQueryArgProfileIsUnknown": true,
      "QueryArgProfiles": {
        "Quantity": 0,
        "Items": []
      }
    },
    "ContentTypeProfileConfig": {
      "ForwardWhenContentTypeIsUnknown": true,
      "ContentTypeProfiles": {
        "Quantity": 1,
        "Items": [
          {

```

```
        "Format": "URLEncoded",
        "ProfileId": "P280MFCLSY0CVU",
        "ContentType": "application/x-www-form-urlencoded"
      }
    ]
  }
}
```

- Para API obtener más información, consulte la Referencia [GetFieldLevelEncryptionConfig](#) de AWS CLI comandos.

## get-field-level-encryption-profile-config

En el siguiente ejemplo de código, se muestra cómo usar `get-field-level-encryption-profile-config`.

### AWS CLI

Para obtener una configuración de CloudFront perfil de cifrado a nivel de campo

El siguiente ejemplo obtiene metadatos sobre el perfil de cifrado a CloudFront nivel de campo con un identificador `PPK0U0SIF5WSV`, incluido su: ETag

```
aws cloudfront get-field-level-encryption-profile-config --id PPK0U0SIF5WSV
```

Salida:

```
{
  "ETag": "E1QQG65FS2L2GC",
  "FieldLevelEncryptionProfileConfig": {
    "Name": "ExampleFLEProfile",
    "CallerReference": "cli-example",
    "Comment": "FLE profile for AWS CLI example",
    "EncryptionEntities": {
      "Quantity": 1,
      "Items": [
        {
          "PublicKeyId": "K2K8NC4HVFE3M0",
          "ProviderId": "ExampleFLEProvider",
          "FieldPatterns": {
```

```
      "Quantity": 1,  
      "Items": [  
        "ExampleSensitiveField"  
      ]  
    }  
  }  
]  
}
```

- Para API obtener más información, consulte la Referencia [GetFieldLevelEncryptionProfileConfig](#) de AWS CLI comandos.

## get-field-level-encryption-profile

En el siguiente ejemplo de código, se muestra cómo usar `get-field-level-encryption-profile`.

### AWS CLI

Para obtener un perfil de CloudFront cifrado a nivel de campo

En el ejemplo siguiente se obtiene el perfil de cifrado a CloudFront nivel de campo con un identificador `PPK0U0SIF5WSV`, que incluye su: ETag

```
aws cloudfront get-field-level-encryption-profile --id PPK0U0SIF5WSV
```

Salida:

```
{  
  "ETag": "E1QQG65FS2L2GC",  
  "FieldLevelEncryptionProfile": {  
    "Id": "PPK0U0SIF5WSV",  
    "LastModifiedTime": "2019-12-10T01:03:16.537Z",  
    "FieldLevelEncryptionProfileConfig": {  
      "Name": "ExampleFLEProfile",  
      "CallerReference": "cli-example",  
      "Comment": "FLE profile for AWS CLI example",  
      "EncryptionEntities": {  
        "Quantity": 1,  
        "Items": [  
          "ExampleSensitiveField"  
        ]  
      }  
    }  
  }  
}
```

```

    {
      "PublicKeyId": "K2K8NC4HVFE3M0",
      "ProviderId": "ExampleFLEProvider",
      "FieldPatterns": {
        "Quantity": 1,
        "Items": [
          "ExampleSensitiveField"
        ]
      }
    }
  ]
}

```

- Para API obtener más información, consulte la Referencia [GetFieldLevelEncryptionProfile](#) de AWS CLI comandos.

## get-field-level-encryption

En el siguiente ejemplo de código, se muestra cómo usar `get-field-level-encryption`.

### AWS CLI

Para obtener una configuración de CloudFront cifrado a nivel de campo

En el siguiente ejemplo, se obtiene la configuración de cifrado a CloudFront nivel de campo con el IDC3KM2WVD605UAY, incluido su: ETag

```
aws cloudfront get-field-level-encryption --id C3KM2WVD605UAY
```

Salida:

```

{
  "ETag": "E2P4Z4VU7TY5SG",
  "FieldLevelEncryption": {
    "Id": "C3KM2WVD605UAY",
    "LastModifiedTime": "2019-12-10T21:30:18.974Z",
    "FieldLevelEncryptionConfig": {
      "CallerReference": "cli-example",
      "Comment": "Example FLE configuration",

```

```

    "QueryArgProfileConfig": {
      "ForwardWhenQueryArgProfileIsUnknown": true,
      "QueryArgProfiles": {
        "Quantity": 0,
        "Items": []
      }
    },
    "ContentTypeProfileConfig": {
      "ForwardWhenContentTypeIsUnknown": true,
      "ContentTypeProfiles": {
        "Quantity": 1,
        "Items": [
          {
            "Format": "URLEncoded",
            "ProfileId": "P280MFCLSYOCVU",
            "ContentType": "application/x-www-form-urlencoded"
          }
        ]
      }
    }
  }
}

```

- Para API obtener más información, consulte la Referencia [GetFieldLevelEncryption](#) de AWS CLI comandos.

## get-invalidation

En el siguiente ejemplo de código, se muestra cómo usar `get-invalidation`.

### AWS CLI

Para obtener una CloudFront invalidación

En el siguiente ejemplo, se obtiene la invalidación con el identificador `I2J0I21PCUY0IK` de la CloudFront distribución con el identificador: `EDFDVBD6EXAMPLE`

```
aws cloudfront get-invalidation --id I2J0I21PCUY0IK --distribution-id EDFDVBD6EXAMPLE
```

Salida:

```
{
  "Invalidation": {
    "Status": "Completed",
    "InvalidationBatch": {
      "Paths": {
        "Items": [
          "/example-path/example-file.jpg",
          "/example-path/example-file-2.jpg"
        ],
        "Quantity": 2
      },
      "CallerReference": "cli-example"
    },
    "Id": "I2J0I21PCUY0IK",
    "CreateTime": "2019-12-05T18:40:49.413Z"
  }
}
```

- Para API obtener más información, consulte [GetInvalidation](#) la Referencia de AWS CLI comandos.

## get-public-key-config

En el siguiente ejemplo de código, se muestra cómo usar `get-public-key-config`.

### AWS CLI

Para obtener una configuración de clave CloudFront pública

En el siguiente ejemplo, se obtienen metadatos sobre la clave CloudFront pública con el `IDKDFB19YGCR002`, incluido el suyo `ETag`. El ID de clave pública se devuelve en los `list-public-keys` comandos `create-public-key` y.

```
aws cloudfront get-public-key-config --id KDFB19YGCR002
```

Salida:

```
{
  "ETag": "E2QWRUHEXAMPLE",
  "PublicKeyConfig": {
    "CallerReference": "cli-example",
    "Name": "ExampleKey",
```

```

    "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEAxPmBCA2Ks0lnd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1U0WcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBaz03ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95y1UQd9LfYTBzVZYG2tSesplc0kjM3\n2Uu
+oMwXQAw1NINnSLPinMVsutJy6Zq1V3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nnq
+kGZ2NQ0FyIyT2eiLK0X5Rgb/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nnrwIDAQAB\n-----END
PUBLIC KEY-----\n",
    "Comment": "example public key"
  }
}

```

- Para API obtener más información, consulte [GetPublicKeyConfig](#) la Referencia de AWS CLI comandos.

## get-public-key

En el siguiente ejemplo de código, se muestra cómo usar `get-public-key`.

### AWS CLI

Para obtener una clave CloudFront pública

En el siguiente ejemplo, se obtiene la clave CloudFront pública con el identificador `KDFB19YGCR002`, incluido el suyo ETag. El identificador de clave pública se devuelve en los `list-public-keys` comandos `create-public-key` y.

```
aws cloudfront get-public-key --id KDFB19YGCR002
```

Salida:

```

{
  "ETag": "E2QWRUHEXAMPLE",
  "PublicKey": {
    "Id": "KDFB19YGCR002",
    "CreatedTime": "2019-12-05T18:51:43.781Z",
    "PublicKeyConfig": {
      "CallerReference": "cli-example",
      "Name": "ExampleKey",
      "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEAxPmBCA2Ks0lnd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1U0WcmZX2u70dYcSIzB1ofZtcn4cJ

```



```

\nenHBAz03ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95y1UQd9LFYTBzVZYG2tSesplc0kjM3\n2Uu
+oMwxQAw1NINnSLPinMVsutJy6Zq1V3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nnq
+kGZ2NQ0FyIyT2eiLK0X5Rgb/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nnrwIDAQAB\n-----END
PUBLIC KEY-----\n",
    "Comment": "example public key"
  }
}
}

```

- Para API obtener más información, consulte [GetPublicKey](#) la Referencia de AWS CLI comandos.

## list-cloud-front-origin-access-identities

En el siguiente ejemplo de código, se muestra cómo usar `list-cloud-front-origin-access-identities`.

### AWS CLI

Para enumerar las identidades de acceso de CloudFront origen

En el siguiente ejemplo, se obtiene una lista de las identidades de acceso de CloudFront origen (OAI) de su AWS cuenta:

```
aws cloudfront list-cloud-front-origin-access-identities
```

Salida:

```

{
  "CloudFrontOriginAccessIdentityList": {
    "Items": [
      {
        "Id": "E74FTE3AEXAMPLE",
        "S3CanonicalUserId":
"cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
        "Comment": "Example OAI"
      },
      {
        "Id": "EH1HDMBEXAMPLE",
        "S3CanonicalUserId":
"1489f6f2e6faacaae7ff64c4c3e6956c24f78788abfc1718c3527c263bf7a17EXAMPLE",

```

```

        "Comment": "Test OAI"
      },
      {
        "Id": "E2X2C9TEXAMPLE",
        "S3CanonicalUserId":
"cbfeebb915a64749f9be546a45b3fcfd3a31c779673c13c4dd460911ae402c2EXAMPLE",
        "Comment": "Example OAI #2"
      }
    ]
  }
}

```

- Para API obtener más información, consulta [ListCloudFrontOriginAccessIdentities](#) la Referencia de AWS CLI comandos.

## list-distributions

En el siguiente ejemplo de código, se muestra cómo usar `list-distributions`.

### AWS CLI

Para enumerar las CloudFront distribuciones

En el siguiente ejemplo, se obtiene una lista de las CloudFront distribuciones de su AWS cuenta:

```
aws cloudfront list-distributions
```

Salida:

```

{
  "DistributionList": {
    "Items": [
      {
        "Id": "EMLARXS9EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/
EMLARXS9EXAMPLE",
        "Status": "InProgress",
        "LastModifiedTime": "2019-11-22T00:55:15.705Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
          "Enabled": false,

```

```

        "Quantity": 0
    },
    "DistributionConfig": {
        "CallerReference": "cli-example",
        "Aliases": {
            "Quantity": 0
        },
        "DefaultRootObject": "index.html",
        "Origins": {
            "Quantity": 1,
            "Items": [
                {
                    "Id": "awsexamplebucket.s3.amazonaws.com-cli-
example",
                    "DomainName": "awsexamplebucket.s3.amazonaws.com",
                    "OriginPath": "",
                    "CustomHeaders": {
                        "Quantity": 0
                    },
                    "S3OriginConfig": {
                        "OriginAccessIdentity": ""
                    }
                }
            ]
        },
        "OriginGroups": {
            "Quantity": 0
        },
        "DefaultCacheBehavior": {
            "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-
example",
            "ForwardedValues": {
                "QueryString": false,
                "Cookies": {
                    "Forward": "none"
                },
                "Headers": {
                    "Quantity": 0
                },
                "QueryStringCacheKeys": {
                    "Quantity": 0
                }
            },
            "TrustedSigners": {

```

```
        "Enabled": false,
        "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ],
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
        "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponses": {
    "Quantity": 0
},
"Comment": "",
"Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
```



```
        "S3OriginConfig": {
            "OriginAccessIdentity": ""
        }
    ],
},
"OriginGroups": {
    "Quantity": 0
},
"DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket1.s3.amazonaws.com-cli-
example",
    "ForwardedValues": {
        "QueryString": false,
        "Cookies": {
            "Forward": "none"
        },
        "Headers": {
            "Quantity": 0
        },
        "QueryStringCacheKeys": {
            "Quantity": 0
        }
    },
    "TrustedSigners": {
        "Enabled": false,
        "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ],
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    }
},
```

```

        "SmoothStreaming": false,
        "DefaultTTL": 86400,
        "MaxTTL": 31536000,
        "Compress": false,
        "LambdaFunctionAssociations": {
            "Quantity": 0
        },
        "FieldLevelEncryptionId": ""
    },
    "CacheBehaviors": {
        "Quantity": 0
    },
    "CustomErrorResponses": {
        "Quantity": 0
    },
    "Comment": "",
    "Logging": {
        "Enabled": false,
        "IncludeCookies": false,
        "Bucket": "",
        "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
}
},
{
    "Id": "E1X5IZQEXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/
E1X5IZQEXAMPLE",

```

```
"Status": "Deployed",
"LastModifiedTime": "2019-11-06T21:31:48.864Z",
"DomainName": "d2e04y12345678.cloudfront.net",
"Aliases": {
  "Quantity": 0
},
"Origins": {
  "Quantity": 1,
  "Items": [
    {
      "Id": "awsexamplebucket2",
      "DomainName": "awsexamplebucket2.s3.us-
west-2.amazonaws.com",
      "OriginPath": "",
      "CustomHeaders": {
        "Quantity": 0
      },
      "S3OriginConfig": {
        "OriginAccessIdentity": ""
      }
    }
  ]
},
"OriginGroups": {
  "Quantity": 0
},
"DefaultCacheBehavior": {
  "TargetOriginId": "awsexamplebucket2",
  "ForwardedValues": {
    "QueryString": false,
    "Cookies": {
      "Forward": "none"
    }
  },
  "Headers": {
    "Quantity": 0
  },
  "QueryStringCacheKeys": {
    "Quantity": 0
  }
},
"TrustedSigners": {
  "Enabled": false,
  "Quantity": 0
},
}
```



```
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ],
      "CachedMethods": {
        "Quantity": 2,
        "Items": [
          "HEAD",
          "GET"
        ]
      }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
      "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
  },
  "CacheBehaviors": {
    "Quantity": 0
  },
  "CustomErrorResponses": {
    "Quantity": 0
  },
  "Comment": "",
  "PriceClass": "PriceClass_All",
  "Enabled": true,
  "ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
  },
  "Restrictions": {
    "GeoRestriction": {
      "RestrictionType": "none",
      "Quantity": 0
    }
  }
}
```

```

        },
        "WebACLId": "",
        "HttpVersion": "HTTP1_1",
        "IsIPV6Enabled": true
    }
]
}
}

```

- Para API obtener más información, consulte [ListDistributions](#) la Referencia de AWS CLI comandos.

## list-field-level-encryption-configs

En el siguiente ejemplo de código, se muestra cómo usar `list-field-level-encryption-configs`.

### AWS CLI

Para enumerar las configuraciones de CloudFront cifrado a nivel de campo

En el siguiente ejemplo, se obtiene una lista de las configuraciones de cifrado a CloudFront nivel de campo de su cuenta: AWS

```
aws cloudfront list-field-level-encryption-configs
```

Salida:

```

{
  "FieldLevelEncryptionList": {
    "MaxItems": 100,
    "Quantity": 1,
    "Items": [
      {
        "Id": "C3KM2WVD605UAY",
        "LastModifiedTime": "2019-12-10T21:30:18.974Z",
        "Comment": "Example FLE configuration",
        "QueryArgProfileConfig": {
          "ForwardWhenQueryArgProfileIsUnknown": true,
          "QueryArgProfiles": {
            "Quantity": 0,
            "Items": []
          }
        }
      }
    ]
  }
}

```

```

    }
  },
  "ContentTypeProfileConfig": {
    "ForwardWhenContentTypeIsUnknown": true,
    "ContentTypeProfiles": {
      "Quantity": 1,
      "Items": [
        {
          "Format": "URLEncoded",
          "ProfileId": "P280MFCLSY0CVU",
          "ContentType": "application/x-www-form-urlencoded"
        }
      ]
    }
  }
}

```

- Para API obtener más información, consulte la Referencia [ListFieldLevelEncryptionConfigs](#) de AWS CLI comandos.

## list-field-level-encryption-profiles

En el siguiente ejemplo de código, se muestra cómo usar `list-field-level-encryption-profiles`.

### AWS CLI

Para enumerar los CloudFront perfiles de cifrado a nivel de campo

En el siguiente ejemplo, se obtiene una lista de los perfiles de cifrado a CloudFront nivel de campo de su cuenta: AWS

```
aws cloudfront list-field-level-encryption-profiles
```

Salida:

```

{
  "FieldLevelEncryptionProfileList": {
    "MaxItems": 100,

```

```
"Quantity": 2,
"Items": [
  {
    "Id": "P280MFCLSY0CVU",
    "LastModifiedTime": "2019-12-05T01:05:39.896Z",
    "Name": "ExampleFLEProfile",
    "EncryptionEntities": {
      "Quantity": 1,
      "Items": [
        {
          "PublicKeyId": "K2K8NC4HVFE3M0",
          "ProviderId": "ExampleFLEProvider",
          "FieldPatterns": {
            "Quantity": 1,
            "Items": [
              "ExampleSensitiveField"
            ]
          }
        }
      ]
    },
    "Comment": "FLE profile for AWS CLI example"
  },
  {
    "Id": "PPK0UOSIF5WSV",
    "LastModifiedTime": "2019-12-10T01:03:16.537Z",
    "Name": "ExampleFLEProfile2",
    "EncryptionEntities": {
      "Quantity": 1,
      "Items": [
        {
          "PublicKeyId": "K2ABC10EXAMPLE",
          "ProviderId": "ExampleFLEProvider2",
          "FieldPatterns": {
            "Quantity": 1,
            "Items": [
              "ExampleSensitiveField2"
            ]
          }
        }
      ]
    },
    "Comment": "FLE profile #2 for AWS CLI example"
  }
]
```

```
    ]
  }
}
```

- Para API obtener más información, consulte la Referencia [ListFieldLevelEncryptionProfiles](#) de AWS CLI comandos.

## list-invalidations

En el siguiente ejemplo de código, se muestra cómo usar `list-invalidations`.

### AWS CLI

Para enumerar las CloudFront invalidaciones

En el siguiente ejemplo, se obtiene una lista de las invalidaciones de la CloudFront distribución con el identificador: EDFDVBD6EXAMPLE

```
aws cloudfront list-invalidations --distribution-id EDFDVBD6EXAMPLE
```

Salida:

```
{
  "InvalidationList": {
    "Marker": "",
    "Items": [
      {
        "Status": "Completed",
        "Id": "YNY2LI2BVJ4NJU",
        "CreateTime": "2019-08-31T21:15:52.042Z"
      }
    ],
    "IsTruncated": false,
    "MaxItems": 100,
    "Quantity": 1
  }
}
```

- Para API obtener más información, consulte la Referencia [ListInvalidations](#) de AWS CLI comandos.

## list-public-keys

En el siguiente ejemplo de código, se muestra cómo usar `list-public-keys`.

### AWS CLI

Para enumerar las claves CloudFront públicas

En el siguiente ejemplo, se obtiene una lista de las claves CloudFront públicas de tu AWS cuenta:

```
aws cloudfront list-public-keys
```

Salida:

```
{
  "PublicKeyList": {
    "MaxItems": 100,
    "Quantity": 2,
    "Items": [
      {
        "Id": "K2K8NC4HVFE3M0",
        "Name": "ExampleKey",
        "CreatedTime": "2019-12-05T01:04:28.818Z",
        "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAA0CAQ8AMIIBCgKCAQEApMbCA2Ks01nd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1U0WcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAz03ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95y1UQd9LFYTBzVZYG2tSesplc0kjM3\n2Uu
+oMWxQAw1NINnSLPinMVsutJy6Zq1V3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nnq
+kGZ2NQ0FyIyT2eiLK0X5RgB/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nnrwIDAQAB\n-----END
PUBLIC KEY-----\n",
        "Comment": "example public key"
      },
      {
        "Id": "K1S0LWQ2L5HTBU",
        "Name": "ExampleKey2",
        "CreatedTime": "2019-12-09T23:28:11.110Z",
        "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAA0CAQ8AMIIBCgKCAQEAp0CAg88A8+f4dujn9Izt
\n26LxtgAkn2opGgo/NKpMiaisyw5qlg3f1gol17FV6pYN178iJg3E08JBbwt1H
+cR9\nLGSf60NDeVhm760c39Np/vWg0dsGQcRbi9WmKZeS0DqjQGzVZWqPmito3FzWV6b
\nfVY5N36U/RdbVAJm95Km+qaMY1bIdF40t72bi3IkKYV5h1B2XoDjlQ9F6ajQKyTB
\nMHa3SN8q+3ZjQ4sJJ7D1V6r4wR8jDcFVD5NckWJmngIVnk0QM37NYeoDnka0uTpu\nnha/
```

```
+3b8t0b2z3LBVHPkp85zJRA0XacSwf5rZtPYKBNFsixTa2n55k2r218m0kMC4\nUwIDAQAB\n-----END
PUBLIC KEY-----",
      "Comment": "example public key #2"
    }
  ]
}
}
```

- Para API obtener más información, consulte [ListPublicKeys](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de una CloudFront distribución

En el siguiente ejemplo, se obtiene una lista de las etiquetas de una CloudFront distribución:

```
aws cloudfront list-tags-for-resource \
  --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE
```

Salida:

```
{
  "Tags": {
    "Items": [
      {
        "Key": "DateCreated",
        "Value": "2019-12-04"
      },
      {
        "Key": "Name",
        "Value": "Example name"
      },
      {
        "Key": "Project",
        "Value": "Example project"
      }
    ]
  }
}
```

```

    ]
  }
}

```

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## sign

En el siguiente ejemplo de código, se muestra cómo usar `sign`.

### AWS CLI

Para firmar un CloudFront URL

El siguiente ejemplo firma un CloudFront URL. Para firmar un URL, necesita el identificador del par de claves (denominado identificador de clave de acceso en la consola AWS de administración) y la clave privada del par de claves del firmante de CloudFront confianza. Para obtener más información sobre cómo URLs publicar [contenido privado con cookies firmadas URLs y firmadas](#) en la Guía para CloudFront desarrolladores de Amazon.

```

aws cloudfront sign \
  --url https://d111111abcdef8.cloudfront.net/private-content/private-file.html \
  --key-pair-id APKAEIBAERJR2EXAMPLE \
  --private-key file://cf-signer-priv-key.pem \
  --date-less-than 2020-01-01

```

Salida:

```

https://d111111abcdef8.cloudfront.net/private-content/private-
file.html?Expires=1577836800&Signature=nEXK7Kby47XKeZQKvc6pwkif6oZc-
JWSpDkH0UH7EBGGqvgurkecCbgL5VfUAXyLQuJxFwRQWscz-
owcq9KpmewCXrXQbPaJZNi9XSNwf4YKurPDQYaRQawKoeenH0GFteRf9ELK-
Bs3n1jTLjtbgzIUt7QJNKXcWr8AuUYikzGdJ4-qzx6WnxXfH~fxg4-
GGl6l2kgCpXUB6Jx6K~Y3kpV0dzUP0IqFLHANJojbhxqrVejomZZ2XrquDvNUCCIbePGnR3d24UPaLXG4FK0qNEaWDIB
GNvjRJxqWf93uMobeM0iVYahb-e0KIitiQewGcm0eLZQ__&Key-Pair-Id=APKAEIBAERJR2EXAMPLE

```

- Para API obtener más información, consulte la Referencia de AWS CLI comandos de inicio de [sesión](#).



## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar una CloudFront distribución

El siguiente `tag-resource` ejemplo agrega dos etiquetas a la CloudFront distribución especificada.

```
aws cloudfront tag-resource \  
  --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE \  
  --tags 'Items=[{Key=Name,Value="Example name"},{Key=Project,Value="Example project"}]'
```

En lugar de utilizar argumentos de línea de comandos, puede proporcionar las etiquetas en un JSON archivo, como se muestra en el siguiente ejemplo:

```
aws cloudfront tag-resource \  
  --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE \  
  --tags file://tags.json
```

Contenidos de `tags.json`:

```
{  
  "Items": [  
    {  
      "Key": "Name",  
      "Value": "Example name"  
    },  
    {  
      "Key": "Project",  
      "Value": "Example project"  
    }  
  ]  
}
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar etiquetas de una CloudFront distribución

El siguiente ejemplo elimina dos etiquetas de una CloudFront distribución mediante argumentos de línea de comandos:

```
aws cloudfront untag-resource \  
  --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE \  
  --tag-keys Items=Name,Project
```

En lugar de utilizar argumentos de línea de comandos, puede proporcionar las claves de etiquetas en un JSON archivo, como se muestra en el siguiente ejemplo:

```
aws cloudfront untag-resource \  
  --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE \  
  --tag-keys file://tag-keys.json
```

El archivo `tag-keys.json` es un JSON documento de la carpeta actual que contiene lo siguiente:

```
{  
  "Items": [  
    "Name",  
    "Project"  
  ]  
}
```

Cuando tenga éxito, este comando no tiene salida.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-cloud-front-origin-access-identity

En el siguiente ejemplo de código, se muestra cómo usar `update-cloud-front-origin-access-identity`.

## AWS CLI

Para actualizar una identidad de acceso a un CloudFront origen

En el siguiente ejemplo, se actualiza la identidad de acceso de origen (OAI) con la IDE74FTE3AEXAMPLE. El único campo que puedes actualizar OAI es elComment.

Para actualizar unOAI, debes tener OAI su ID yETag. El OAI ID se devuelve en el resultado de los comandos create-cloud-front-origin -access-identity y list-cloud-front-origin -access-identity. Para obtenerlaETag, utilice el comando -access-identity o -. get-cloud-front-origin get-cloud-front-origin access-identity-config Usa la --if-match opción para proporcionar los. OAI ETag

```
aws cloudfront update-cloud-front-origin-access-identity \  
  --id E74FTE3AEXAMPLE \  
  --if-match E2QWRUHEXAMPLE \  
  --cloud-front-origin-access-identity-config \  
    CallerReference=cli-example,Comment="Example OAI Updated"
```

Puede lograr lo mismo proporcionando la OAI configuración en un JSON archivo, como se muestra en el siguiente ejemplo:

```
aws cloudfront update-cloud-front-origin-access-identity \  
  --id E74FTE3AEXAMPLE \  
  --if-match E2QWRUHEXAMPLE \  
  --cloud-front-origin-access-identity-config file://OAI-config.json
```

El archivo OAI-config.json es un JSON documento del directorio actual que contiene lo siguiente:

```
{  
  "CallerReference": "cli-example",  
  "Comment": "Example OAI Updated"  
}
```

Tanto si proporciona la OAI configuración con un argumento de línea de comandos como con un JSON archivo, el resultado es el mismo:

```
{  
  "ETag": "E9LHASXEXAMPLE",  
  "CloudFrontOriginAccessIdentity": {  
    "Id": "E74FTE3AEXAMPLE",
```

```

    "S3CanonicalUserId":
    "cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
    "CloudFrontOriginAccessIdentityConfig": {
        "CallerReference": "cli-example",
        "Comment": "Example OAI Updated"
    }
}
}

```

- Para API obtener más información, consulte [UpdateCloudFrontOriginAccessIdentity](#) la Referencia de AWS CLI comandos.

## update-distribution

En el siguiente ejemplo de código, se muestra cómo usar `update-distribution`.

### AWS CLI

Para actualizar el objeto raíz predeterminado de una CloudFront distribución

En el siguiente ejemplo, se actualiza el objeto raíz predeterminado `index.html` para la CloudFront distribución con el ID `EDFDVBD6EXAMPLE`:

```

aws cloudfront update-distribution --id EDFDVBD6EXAMPLE \
  --default-root-object index.html

```

Salida:

```

{
  "ETag": "E2QWRUHEXAMPLE",
  "Distribution": {
    "Id": "EDFDVBD6EXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",
    "Status": "InProgress",
    "LastModifiedTime": "2019-12-06T18:55:39.870Z",
    "InProgressInvalidationBatches": 0,
    "DomainName": "d111111abcdef8.cloudfront.net",
    "ActiveTrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
  },
  "DistributionConfig": {

```

```
"CallerReference": "6b10378d-49be-4c4b-a642-419ccaf8f3b5",
"Aliases": {
  "Quantity": 0
},
"DefaultRootObject": "index.html",
"Origins": {
  "Quantity": 1,
  "Items": [
    {
      "Id": "example-website",
      "DomainName": "www.example.com",
      "OriginPath": "",
      "CustomHeaders": {
        "Quantity": 0
      },
      "CustomOriginConfig": {
        "HTTPPort": 80,
        "HTTPSPort": 443,
        "OriginProtocolPolicy": "match-viewer",
        "OriginSslProtocols": {
          "Quantity": 2,
          "Items": [
            "SSLv3",
            "TLSv1"
          ]
        },
        "OriginReadTimeout": 30,
        "OriginKeepaliveTimeout": 5
      }
    }
  ]
},
"OriginGroups": {
  "Quantity": 0
},
"DefaultCacheBehavior": {
  "TargetOriginId": "example-website",
  "ForwardedValues": {
    "QueryString": false,
    "Cookies": {
      "Forward": "none"
    }
  },
  "Headers": {
    "Quantity": 1,
```

```
        "Items": [
            "*"
        ]
    },
    "QueryStringCacheKeys": {
        "Quantity": 0
    }
},
"TrustedSigners": {
    "Enabled": false,
    "Quantity": 0
},
"ViewerProtocolPolicy": "allow-all",
"MinTTL": 0,
"AllowedMethods": {
    "Quantity": 2,
    "Items": [
        "HEAD",
        "GET"
    ],
    "CachedMethods": {
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ]
    }
},
"SmoothStreaming": false,
"DefaultTTL": 86400,
"MaxTTL": 31536000,
"Compress": false,
"LambdaFunctionAssociations": {
    "Quantity": 0
},
"FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponse": {
    "Quantity": 0
},
"Comment": "",
```

```

    "Logging": {
      "Enabled": false,
      "IncludeCookies": false,
      "Bucket": "",
      "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
      "CloudFrontDefaultCertificate": true,
      "MinimumProtocolVersion": "TLSv1",
      "CertificateSource": "cloudfront"
    },
    "Restrictions": {
      "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
      }
    },
    "WebACLId": "",
    "HttpVersion": "http1.1",
    "IsIPV6Enabled": true
  }
}
}
}

```

Para actualizar una CloudFront distribución

El siguiente ejemplo deshabilita la CloudFront distribución con el ID EMLARXS9EXAMPLE proporcionando la configuración de distribución en un JSON archivo denominado `dist-config-disable.json`. Para actualizar una distribución, debe usar la opción `--if-match` para proporcionar la ETag de la distribución. Para obtenerla ETag, utilice el comando `get-distribution` or `get-distribution-config`

Una vez que utilice el siguiente ejemplo para deshabilitar una distribución, puede utilizar el comando `delete-distribution` para eliminarla.

```

aws cloudfront update-distribution \
  --id EMLARXS9EXAMPLE \
  --if-match E2QWRUHEXAMPLE \
  --distribution-config file://dist-config-disable.json

```

El archivo `dist-config-disable.json` es un JSON documento de la carpeta actual que contiene lo siguiente. Observe que el campo `Enabled` está establecido en `false`.

```
{
  "CallerReference": "cli-1574382155-496510",
  "Aliases": {
    "Quantity": 0
  },
  "DefaultRootObject": "index.html",
  "Origins": {
    "Quantity": 1,
    "Items": [
      {
        "Id": "awsexamplebucket.s3.amazonaws.com-1574382155-273939",
        "DomainName": "awsexamplebucket.s3.amazonaws.com",
        "OriginPath": "",
        "CustomHeaders": {
          "Quantity": 0
        },
        "S3OriginConfig": {
          "OriginAccessIdentity": ""
        }
      }
    ]
  },
  "OriginGroups": {
    "Quantity": 0
  },
  "DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-1574382155-273939",
    "ForwardedValues": {
      "QueryString": false,
      "Cookies": {
        "Forward": "none"
      },
      "Headers": {
        "Quantity": 0
      },
      "QueryStringCacheKeys": {
        "Quantity": 0
      }
    },
    "TrustedSigners": {
```



```
        "Enabled": false,
        "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ],
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
        "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponses": {
    "Quantity": 0
},
"Comment": "",
"Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": false,
"ViewerCertificate": {
```

```

    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
  },
  "Restrictions": {
    "GeoRestriction": {
      "RestrictionType": "none",
      "Quantity": 0
    }
  },
  "WebACLId": "",
  "HttpVersion": "http2",
  "IsIPV6Enabled": true
}

```

**Salida:**

```

{
  "ETag": "E9LHASXEXAMPLE",
  "Distribution": {
    "Id": "EMLARXS9EXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EMLARXS9EXAMPLE",
    "Status": "InProgress",
    "LastModifiedTime": "2019-12-06T18:32:35.553Z",
    "InProgressInvalidationBatches": 0,
    "DomainName": "d111111abcdef8.cloudfront.net",
    "ActiveTrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    }
  },
  "DistributionConfig": {
    "CallerReference": "cli-1574382155-496510",
    "Aliases": {
      "Quantity": 0
    }
  },
  "DefaultRootObject": "index.html",
  "Origins": {
    "Quantity": 1,
    "Items": [
      {
        "Id": "awsexamplebucket.s3.amazonaws.com-1574382155-273939",
        "DomainName": "awsexamplebucket.s3.amazonaws.com",
        "OriginPath": "",

```

```
        "CustomHeaders": {
            "Quantity": 0
        },
        "S3OriginConfig": {
            "OriginAccessIdentity": ""
        }
    }
]
},
"OriginGroups": {
    "Quantity": 0
},
"DefaultCacheBehavior": {
    "TargetOriginId":
"awsexamplebucket.s3.amazonaws.com-1574382155-273939",
    "ForwardedValues": {
        "QueryString": false,
        "Cookies": {
            "Forward": "none"
        },
        "Headers": {
            "Quantity": 0
        },
        "QueryStringCacheKeys": {
            "Quantity": 0
        }
    },
    "TrustedSigners": {
        "Enabled": false,
        "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ],
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    }
}
```

```
        ]
      }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
      "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
  },
  "CacheBehaviors": {
    "Quantity": 0
  },
  "CustomErrorResponses": {
    "Quantity": 0
  },
  "Comment": "",
  "Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
  },
  "PriceClass": "PriceClass_All",
  "Enabled": false,
  "ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
  },
  "Restrictions": {
    "GeoRestriction": {
      "RestrictionType": "none",
      "Quantity": 0
    }
  },
  "WebACLId": "",
  "HttpVersion": "http2",
  "IsIPV6Enabled": true
}
}
```

```
}

```

- Para API obtener más información, consulte [UpdateDistribution](#) la Referencia de AWS CLI comandos.

## update-field-level-encryption-config

En el siguiente ejemplo de código, se muestra cómo usar `update-field-level-encryption-config`.

### AWS CLI

Para actualizar una configuración de CloudFront cifrado a nivel de campo

El siguiente ejemplo actualiza el `Comment` campo de la configuración de cifrado a nivel de campo con el ID `C3KM2WVD605UAY` proporcionando los parámetros de un archivo. JSON

Para actualizar una configuración de cifrado a nivel de campo, debe tener el ID de la configuración y. ETag El ID se devuelve en el resultado de los comandos `create-field-level-encryption -config` y `list-field-level-encryption -configs`. Para obtenerlos ETag, utilice el comando `get-field-level-encryption` or `get-field-level-encryption -config`. Utilice la `--if-match` opción para proporcionar las configuraciones. ETag

```
aws cloudfront update-field-level-encryption-config \
  --id C3KM2WVD605UAY \
  --if-match E2P4Z4VU7TY5SG \
  --field-level-encryption-config file://fle-config.json
```

El archivo `file-config.json` es un JSON documento del directorio actual que contiene lo siguiente:

```
{
  "CallerReference": "cli-example",
  "Comment": "Updated example FLE configuration",
  "QueryArgProfileConfig": {
    "ForwardWhenQueryArgProfileIsUnknown": true,
    "QueryArgProfiles": {
      "Quantity": 0
    }
  }
},
```

```

"ContentTypeProfileConfig": {
  "ForwardWhenContentTypeIsUnknown": true,
  "ContentTypeProfiles": {
    "Quantity": 1,
    "Items": [
      {
        "Format": "URLEncoded",
        "ProfileId": "P280MFCLSY0CVU",
        "ContentType": "application/x-www-form-urlencoded"
      }
    ]
  }
}
}

```

**Salida:**

```

{
  "ETag": "E26M4BIAV81ZF6",
  "FieldLevelEncryption": {
    "Id": "C3KM2WVD605UAY",
    "LastModifiedTime": "2019-12-10T22:26:26.170Z",
    "FieldLevelEncryptionConfig": {
      "CallerReference": "cli-example",
      "Comment": "Updated example FLE configuration",
      "QueryArgProfileConfig": {
        "ForwardWhenQueryArgProfileIsUnknown": true,
        "QueryArgProfiles": {
          "Quantity": 0,
          "Items": []
        }
      }
    },
    "ContentTypeProfileConfig": {
      "ForwardWhenContentTypeIsUnknown": true,
      "ContentTypeProfiles": {
        "Quantity": 1,
        "Items": [
          {
            "Format": "URLEncoded",
            "ProfileId": "P280MFCLSY0CVU",
            "ContentType": "application/x-www-form-urlencoded"
          }
        ]
      }
    }
  }
}

```

```

    }
  }
}

```

- Para API obtener más información, consulte [UpdateFieldLevelEncryptionConfig](#) la Referencia de AWS CLI comandos.

## update-field-level-encryption-profile

En el siguiente ejemplo de código, se muestra cómo usar `update-field-level-encryption-profile`.

### AWS CLI

Para actualizar un perfil de CloudFront cifrado a nivel de campo

El siguiente ejemplo actualiza el perfil de cifrado a nivel de campo con el ID. `PPK0U0SIF5WSV` En este ejemplo `Comment`, se actualiza el `Name` y del perfil y se agrega un segundo `FieldPatterns` elemento proporcionando los parámetros de un JSON archivo.

Para actualizar un perfil de cifrado a nivel de campo, debe tener el ID del perfil y. `ETag` El ID se devuelve en el resultado de los comandos `create-field-level-encryption-profile` y `list-field-level-encryption-profiles`. Para obtenerlo `ETag`, utilice el comando `get-field-level-encryption-profile` o `get-field-level-encryption-profile-config`. Utilice la `--if-match` opción para proporcionar los perfiles. `ETag`

```

aws cloudfront update-field-level-encryption-profile \
  --id PPK0U0SIF5WSV \
  --if-match E1QQG65FS2L2GC \
  --field-level-encryption-profile-config file://fle-profile-config.json

```

El archivo `fle-profile-config.json` es un JSON documento del directorio actual que contiene lo siguiente:

```

{
  "Name": "ExampleFLEProfileUpdated",
  "CallerReference": "cli-example",
  "Comment": "Updated FLE profile for AWS CLI example",

```

```

"EncryptionEntities": {
  "Quantity": 1,
  "Items": [
    {
      "PublicKeyId": "K2K8NC4HVFE3M0",
      "ProviderId": "ExampleFLEProvider",
      "FieldPatterns": {
        "Quantity": 2,
        "Items": [
          "ExampleSensitiveField",
          "SecondExampleSensitiveField"
        ]
      }
    }
  ]
}

```

Salida:

```

{
  "ETag": "EJETYFJ9CL66D",
  "FieldLevelEncryptionProfile": {
    "Id": "PPK0UOSIF5WSV",
    "LastModifiedTime": "2019-12-10T19:05:58.296Z",
    "FieldLevelEncryptionProfileConfig": {
      "Name": "ExampleFLEProfileUpdated",
      "CallerReference": "cli-example",
      "Comment": "Updated FLE profile for AWS CLI example",
      "EncryptionEntities": {
        "Quantity": 1,
        "Items": [
          {
            "PublicKeyId": "K2K8NC4HVFE3M0",
            "ProviderId": "ExampleFLEProvider",
            "FieldPatterns": {
              "Quantity": 2,
              "Items": [
                "ExampleSensitiveField",
                "SecondExampleSensitiveField"
              ]
            }
          }
        ]
      }
    }
  }
}

```



```
}  
  }  
} ]
```

- Para API obtener más información, consulte [UpdateFieldLevelEncryptionProfile](#) la Referencia de AWS CLI comandos.

## CloudSearch Ejemplos de Amazon que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Amazon CloudSearch.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **upload-documents**

En el siguiente ejemplo de código, se muestra cómo usar upload-documents.

AWS CLI

El siguiente upload-documents comando carga un lote de JSON documentos a un CloudSearch dominio de Amazon:

```
aws cloudsearchdomain upload-documents --endpoint-url https://doc-my-domain.us-west-1.cloudsearch.amazonaws.com --content-type application/json --documents document-batch.json
```

Salida:

```
{
  "status": "success",
  "adds": 5000,
  "deletes": 0
}
```

- Para API obtener más información, consulte [UploadDocuments](#) la Referencia de AWS CLI comandos.

## CloudTrail ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with CloudTrail.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### add-tags

En el siguiente ejemplo de código, se muestra cómo usar add-tags.

AWS CLI

Para añadir etiquetas a la ruta

El siguiente add-tags comando añade etiquetas paraTrail1:

```
aws cloudtrail add-tags --resource-id arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1 --tags-  
list Key=name,Value=Alice Key=location,Value=us
```

- Para API obtener más información, consulte [AddTags](#) la Referencia de AWS CLI comandos.

## create-subscription

En el siguiente ejemplo de código, se muestra cómo usar `create-subscription`.

### AWS CLI

Para crear y configurar AWS los recursos de una ruta

El siguiente `create-subscription` comando crea un nuevo bucket de S3 y un SNS tema para `Trail1`:

```
aws cloudtrail create-subscription --name Trail1 --s3-new-bucket my-bucket --sns-new-topic my-topic
```

Salida:

```
Setting up new S3 bucket my-bucket...
Setting up new SNS topic my-topic...
Creating/updating CloudTrail configuration...
CloudTrail configuration:
{
  "trailList": [
    {
      "IncludeGlobalServiceEvents": true,
      "Name": "Trail1",
      "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1",
      "LogFileValidationEnabled": false,
      "IsMultiRegionTrail": false,
      "S3BucketName": "my-bucket",
      "SnsTopicName": "my-topic",
      "HomeRegion": "us-east-1"
    }
  ],
  "ResponseMetadata": {
    "HTTPStatusCode": 200,
    "RequestId": "f39e51f6-c615-11e5-85bd-d35ca21ee3e2"
  }
}
Starting CloudTrail service...
Logs will be delivered to my-bucket
```

- Para API obtener más información, consulte [CreateSubscription](#) la Referencia de AWS CLI comandos.

## create-trail

En el siguiente ejemplo de código, se muestra cómo usar `create-trail`.

### AWS CLI

Para crear una ruta

El siguiente `create-trail` comando crea un sendero multirregional denominado `Trail1` y especifica un bucket de S3:

```
aws cloudtrail create-trail --name Trail1 --s3-bucket-name my-bucket --is-multi-region-trail
```

Salida:

```
{
  "IncludeGlobalServiceEvents": true,
  "Name": "Trail1",
  "TrailARN": "arn:aws:cloudtrail:us-west-2:123456789012:trail/Trail1",
  "LogFileValidationEnabled": false,
  "IsMultiRegionTrail": true,
  "S3BucketName": "my-bucket"
}
```

- Para API obtener más información, consulte [CreateTrail](#) la Referencia de AWS CLI comandos.

## delete-trail

En el siguiente ejemplo de código, se muestra cómo usar `delete-trail`.

### AWS CLI

Para eliminar un rastro

El siguiente `delete-trail` comando elimina un sendero denominado `Trail1`:

```
aws cloudtrail delete-trail --name Trail1
```

- Para API obtener más información, consulte [DeleteTrail](#) la Referencia de AWS CLI comandos.

## describe-trails

En el siguiente ejemplo de código, se muestra cómo usar `describe-trails`.

### AWS CLI

Para describir un sendero

El siguiente `describe-trails` comando devuelve la configuración de `Trail1` y `Trail2`:

```
aws cloudtrail describe-trails --trail-name-list Trail1 Trail2
```

Salida:

```
{
  "trailList": [
    {
      "IncludeGlobalServiceEvents": true,
      "Name": "Trail1",
      "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1",
      "LogFileValidationEnabled": false,
      "IsMultiRegionTrail": false,
      "S3BucketName": "my-bucket",
      "CloudWatchLogsRoleArn": "arn:aws:iam::123456789012:role/
CloudTrail_CloudWatchLogs_Role",
      "CloudWatchLogsLogGroupArn": "arn:aws:logs:us-east-1:123456789012:log-
group:CloudTrail:*",
      "SnsTopicName": "my-topic",
      "HomeRegion": "us-east-1"
    },
    {
      "IncludeGlobalServiceEvents": true,
      "Name": "Trail2",
      "S3KeyPrefix": "my-prefix",
      "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail2",
      "LogFileValidationEnabled": false,
      "IsMultiRegionTrail": false,
      "S3BucketName": "my-bucket",
      "KmsKeyId": "arn:aws:kms:us-
east-1:123456789012:key/4c5ae5ac-3c13-421e-8335-c7868ef6a769",
```

```
        "HomeRegion": "us-east-1"
    }
]
}
```

- Para API obtener más información, consulte [DescribeTrails](#) la Referencia de AWS CLI comandos.

## get-event-selectors

En el siguiente ejemplo de código, se muestra cómo usar `get-event-selectors`.

### AWS CLI

Para ver los ajustes del selector de eventos de una ruta

El siguiente `get-event-selectors` comando devuelve la configuración de `Trail1`:

```
aws cloudtrail get-event-selectors --trail-name Trail1
```

Salida:

```
{
  "EventSelectors": [
    {
      "IncludeManagementEvents": true,
      "DataResources": [],
      "ReadWriteType": "All"
    }
  ],
  "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1"
}
```

- Para API obtener más información, consulte [GetEventSelectors](#) la Referencia de AWS CLI comandos.

## get-trail-status

En el siguiente ejemplo de código, se muestra cómo usar `get-trail-status`.

## AWS CLI

Para obtener el estado de un sendero

El siguiente `get-trail-status` comando devuelve los detalles de entrega y registro deTrail1:

```
aws cloudtrail get-trail-status --name Trail1
```

Salida:

```
{
  "LatestNotificationTime": 1454022144.869,
  "LatestNotificationAttemptSucceeded": "2016-01-28T23:02:24Z",
  "LatestDeliveryAttemptTime": "2016-01-28T23:02:24Z",
  "LatestDeliveryTime": 1454022144.869,
  "TimeLoggingStarted": "2015-11-06T18:36:38Z",
  "LatestDeliveryAttemptSucceeded": "2016-01-28T23:02:24Z",
  "IsLogging": true,
  "LatestCloudWatchLogsDeliveryTime": 1454022144.918,
  "StartLoggingTime": 1446834998.695,
  "StopLoggingTime": 1446834996.933,
  "LatestNotificationAttemptTime": "2016-01-28T23:02:24Z",
  "TimeLoggingStopped": "2015-11-06T18:36:36Z"
}
```

- Para API obtener más información, consulte [GetTrailStatus](#) la Referencia de AWS CLI comandos.

## list-public-keys

En el siguiente ejemplo de código, se muestra cómo usar `list-public-keys`.

## AWS CLI

Para enumerar todas las claves públicas de un sendero

El siguiente `list-public-keys` comando devuelve todas las claves públicas cuyas claves privadas se utilizaron para firmar los archivos de resumen dentro del intervalo de tiempo especificado:

```
aws cloudtrail list-public-keys --start-time 2016-01-01T20:30:00.000Z
```

Salida:

```
{
  "PublicKeyList": [
    {
      "ValidityStartTime": 1453076702.0,
      "ValidityEndTime": 1455668702.0,
      "Value": "MIIBCgKCAQEA1SS3c192HDycr/MTj0mo0has8habjrraXw+Kz1WF0axSI2tcF
+3iJ9BKQAVSKxGwxwu3m0wG3J
+kU11xboEcEPHYoIYMBgfSw7KGnuDKwkLzsQWhUJ0cIb0HASox1vv/5fNXkrHhGbDCHeVXm804c83nvHUEFYThr1PfyP
+4WGDk+BGH5m9iuiAKkipEHWmU18/P7XpfpWQuk4h8g3pXZ0rNXr081bh4d39svj7Uqdhv0XoBISp9t/
EXYuePGEtBdrKD9Dz+VHwyUPtBQvYr9BnkF88qBnaPNhS44rzwIDAQAB",
      "Fingerprint": "7f3f401420072e50a65a141430817ab3"
    }
  ]
}
```

- Para API obtener más información, consulte [ListPublicKeys](#) la Referencia de AWS CLI comandos.

## list-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-tags`.

AWS CLI

Para enumerar las etiquetas de un sendero

El siguiente `list-tags` comando muestra las etiquetas de `Trail1` y `Trail2`:

```
aws cloudtrail list-tags --resource-id-list arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1 arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail2
```

Salida:

```
{
  "ResourceTagList": [
```



```
{
  "ResourceId": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1",
  "TagsList": [
    {
      "Value": "Alice",
      "Key": "name"
    },
    {
      "Value": "us",
      "Key": "location"
    }
  ]
},
{
  "ResourceId": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail2",
  "TagsList": [
    {
      "Value": "Bob",
      "Key": "name"
    }
  ]
}
]
```

- Para API obtener más información, consulte [ListTags](#) la Referencia de AWS CLI comandos.

## lookup-events

En el siguiente ejemplo de código, se muestra cómo usar `lookup-events`.

### AWS CLI

Para buscar eventos para una ruta

El siguiente `lookup-events` comando busca los eventos de API actividad por el atributo `eventName`:

```
aws cloudtrail lookup-events --lookup-attributes AttributeKey=EventName,AttributeValue=ConsoleLogin
```

Salida:

```
{
  "Events": [
    {
      "EventId": "654ccbc0-ba0d-486a-9076-dbf7274677a7",
      "Username": "my-session-name",
      "EventTime": "2021-11-18T09:41:02-08:00",
      "CloudTrailEvent": "{\"eventVersion\":\"1.02\",\"userIdentity\":{\"type\":\"AssumedRole\",\"principalId\":\"AROAJIKPFTA72SWU4L7T4:my-session-name\",\"arn\":\"arn:aws:sts::123456789012:assumed-role/my-role/my-session-name\",\"accountId\":\"123456789012\",\"sessionContext\":{\"attributes\":{\"mfaAuthenticated\":\"false\"},\"creationDate\":\"2016-01-26T21:42:12Z\"},\"sessionIssuer\":{\"type\":\"Role\",\"principalId\":\"AROAJIKPFTA72SWU4L7T4\",\"arn\":\"arn:aws:iam::123456789012:role/my-role\",\"accountId\":\"123456789012\",\"userName\":\"my-role\"}}},\"eventTime\":\"2016-01-26T21:42:12Z\",\"eventSource\":\"signin.amazonaws.com\",\"eventName\":\"ConsoleLogin\",\"awsRegion\":\"us-east-1\",\"sourceIPAddress\":\"72.21.198.70\",\"userAgent\":\"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/47.0.2526.111 Safari/537.36\",\"requestParameters\":null,\"responseElements\":{\"ConsoleLogin\":{\"Success\"}},\"additionalEventData\":{\"MobileVersion\":\"No\",\"MFAUsed\":\"No\"},\"eventID\":\"654ccbc0-ba0d-486a-9076-dbf7274677a7\",\"eventType\":\"AwsConsoleSignIn\",\"recipientAccountId\":\"123456789012\"}",
      "EventName": "ConsoleLogin",
      "Resources": []
    }
  ]
}
```

- Para API obtener más información, consulte [LookupEvents](#) la Referencia de AWS CLI comandos.

## put-event-selectors

En el siguiente ejemplo de código, se muestra cómo usar `put-event-selectors`.

### AWS CLI

Ejemplo 1: configurar un registro para registrar los eventos de administración y los eventos de datos mediante selectores de eventos avanzados

Puedes añadir selectores de eventos avanzados y condiciones para tus selectores de eventos avanzados, hasta un máximo de 500 valores para todas las condiciones y selectores de un sendero. Puedes usar selectores de eventos avanzados para registrar todos los tipos de eventos

de datos disponibles. Puede utilizar selectores de eventos avanzados o selectores de eventos básicos, pero no ambos. Si aplica selectores de eventos avanzados a un registro de seguimiento, se sobrescriben todos los selectores de eventos básicos existentes.

En el siguiente ejemplo, se crea un selector de eventos avanzado para una ruta denominada `myTrail` para registrar todos los eventos de administración, registrar S3 PutObject y las DeleteObject API llamadas para todos los bucket de S3 excepto uno, registrar las API llamadas de datos para una función de Lambda denominada `myFunction` y registrar las API llamadas de publicación en un SNS tema denominado `myTopic`

```
aws cloudtrail put-event-selectors \
  --trail-name myTrail \
  --advanced-event-selectors '[{"Name": "Log all management events",
  "FieldSelectors": [{"Field": "eventCategory", "Equals": ["Management"]} ] },
{"Name": "Log PutObject and DeleteObject events for all but one
bucket", "FieldSelectors": [{"Field": "eventCategory", "Equals": ["Data"]} ],
{"Field": "resources.type", "Equals": ["AWS::S3::Object"]} ], {"Field":
"eventName", "Equals": ["PutObject", "DeleteObject"]} ], {"Field": "resources.ARN",
"NotStartsWith": ["arn:aws:s3:::sample_bucket_name/"]} ]}], {"Name": "Log
data events for a specific Lambda function", "FieldSelectors": [{"Field":
"eventCategory", "Equals": ["Data"]} ], {"Field": "resources.type",
"Equals": ["AWS::Lambda::Function"]} ], {"Field": "resources.ARN", "Equals":
["arn:aws:lambda:us-east-1:123456789012:function:myFunction"]} ]}], {"Name":
"Log all Publish API calls on a specific SNS topic", "FieldSelectors":
[{"Field": "eventCategory", "Equals": ["Data"]} ], {"Field": "resources.type",
"Equals": ["AWS::SNS::Topic"]} ], {"Field": "eventName", "Equals":
["Publish"]} ], {"Field": "resources.ARN", "Equals": ["arn:aws:sns:us-
east-1:123456789012:myTopic.fifo"]} ]}]'
```

Salida:

```
{
  "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/myTrail",
  "AdvancedEventSelectors": [
    {
      "Name": "Log all management events",
      "FieldSelectors": [
        {
          "Field": "eventCategory",
          "Equals": [
            "Management"
          ]
        }
      ]
    }
  ]
}
```

```
    }
  ]
},
{
  "Name": "Log PutObject and DeleteObject events for all but one bucket",
  "FieldSelectors": [
    {
      "Field": "eventCategory",
      "Equals": [
        "Data"
      ]
    },
    {
      "Field": "resources.type",
      "Equals": [
        "AWS::S3::Object"
      ]
    },
    {
      "Field": "eventName",
      "Equals": [
        "PutObject",
        "DeleteObject"
      ]
    },
    {
      "Field": "resources.ARN",
      "NotStartsWith": [
        "arn:aws:s3:::sample_bucket_name/"
      ]
    }
  ]
},
{
  "Name": "Log data events for a specific Lambda function",
  "FieldSelectors": [
    {
      "Field": "eventCategory",
      "Equals": [
        "Data"
      ]
    },
    {
      "Field": "resources.type",
```

```
        "Equals": [
            "AWS::Lambda::Function"
        ]
    },
    {
        "Field": "resources.ARN",
        "Equals": [
            "arn:aws:lambda:us-east-1:123456789012:function:myFunction"
        ]
    }
]
},
{
    "Name": "Log all Publish API calls on a specific SNS topic",
    "FieldSelectors": [
        {
            "Field": "eventCategory",
            "Equals": [
                "Data"
            ]
        },
        {
            "Field": "resources.type",
            "Equals": [
                "AWS::SNS::Topic"
            ]
        },
        {
            "Field": "eventName",
            "Equals": [
                "Publish"
            ]
        },
        {
            "Field": "resources.ARN",
            "Equals": [
                "arn:aws:sns:us-east-1:123456789012:myTopic.fifo"
            ]
        }
    ]
}
]
}
```

Para obtener más información, consulte [Registrar eventos mediante selectores de eventos avanzados](#) en la Guía del AWS CloudTrail usuario.

Ejemplo 2: configurar los selectores de eventos para que un registro registre todos los eventos de administración y datos

Puede configurar hasta cinco selectores de eventos para un registro de seguimiento y hasta 250 recursos de datos para un registro de seguimiento. Los selectores de eventos también se denominan selectores de eventos básicos. Puede usar selectores de eventos para registrar eventos de administración y eventos de datos para objetos S3, funciones Lambda y tablas de DynamoDB. Para registrar eventos de datos para otros tipos de recursos, debe usar selectores de eventos avanzados.

En el siguiente ejemplo, se crea un selector de eventos para una ruta cuyo nombre `TrailName` incluye todos los eventos de administración, los eventos de datos para dos combinaciones de bucket y prefijo de Amazon S3 y los eventos de datos para una sola función de AWS Lambda denominada `hello-world-python-function`

```
aws cloudtrail put-event-selectors \
  --trail-name TrailName \
  --event-selectors '[{"ReadWriteType": "All", "IncludeManagementEvents":
true, "DataResources": [{"Type": "AWS::S3::Object", "Values":
["arn:aws:s3:::mybucket/prefix", "arn:aws:s3:::mybucket2/prefix2"]},
{"Type": "AWS::Lambda::Function", "Values": ["arn:aws:lambda:us-
west-2:999999999999:function:hello-world-python-function"]}]]'
```

Salida:

```
{
  "EventSelectors": [
    {
      "IncludeManagementEvents": true,
      "DataResources": [
        {
          "Values": [
            "arn:aws:s3:::mybucket/prefix",
            "arn:aws:s3:::mybucket2/prefix2"
          ],
          "Type": "AWS::S3::Object"
        },
        {
          "Values": [
```

```

        "arn:aws:lambda:us-west-2:123456789012:function:hello-world-
python-function"
        ],
        "Type": "AWS::Lambda::Function"
    },
    ],
    "ReadWriteType": "All"
}
],
"TrailARN": "arn:aws:cloudtrail:us-east-2:123456789012:trail/TrailName"
}

```

Para obtener más información, consulte [Registrar eventos mediante selectores de eventos básicos](#) en la Guía del usuario.AWS CloudTrail

Ejemplo 3: configurar los selectores de eventos de una ruta para registrar los eventos de administración, todos los eventos de datos de S3 en los objetos de S3 y todos los eventos de datos de Lambda en las funciones de su cuenta

En el siguiente ejemplo, se crea un selector de eventos para una ruta denominada `TrailName2` que incluye todos los eventos de administración y todos los eventos de datos de todos los buckets de Amazon S3 y las funciones de AWS Lambda de la cuenta. AWS

```

aws cloudtrail put-event-selectors \
  --trail-name TrailName2 \
  --event-selectors '[{"ReadWriteType": "All", "IncludeManagementEvents":
true, "DataResources": [{"Type": "AWS::S3::Object", "Values": ["arn:aws:s3"]},
{"Type": "AWS::Lambda::Function", "Values": ["arn:aws:lambda"]}]]'

```

Salida:

```

{
  "EventSelectors": [
    {
      "IncludeManagementEvents": true,
      "DataResources": [
        {
          "Values": [
            "arn:aws:s3"
          ],
          "Type": "AWS::S3::Object"
        }
      ]
    }
  ]
}

```

```

        {
            "Values": [
                "arn:aws:lambda"
            ],
            "Type": "AWS::Lambda::Function"
        },
    ],
    "ReadWriteType": "All"
}
],
"TrailARN": "arn:aws:cloudtrail:us-east-2:123456789012:trail/TrailName2"
}

```

Para obtener más información, consulte [Registrar eventos mediante selectores de eventos básicos](#) en la Guía del AWS CloudTrail usuario.

- Para API obtener más información, consulte [PutEventSelectors](#) la Referencia de AWS CLI comandos.

## remove-tags

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags`.

### AWS CLI

Para eliminar las etiquetas de un sendero

El siguiente `remove-tags` comando elimina las etiquetas especificadas para `Trail1`:

```
aws cloudtrail remove-tags --resource-id arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1 --tags-list Key=name Key=location
```

- Para API obtener más información, consulte [RemoveTags](#) la Referencia de AWS CLI comandos.

## start-logging

En el siguiente ejemplo de código, se muestra cómo usar `start-logging`.

### AWS CLI

Para empezar a registrar una ruta



El siguiente `start-logging` comando activa el registro de `Trail1`:

```
aws cloudtrail start-logging --name Trail1
```

- Para API obtener más información, consulte [StartLogging](#) la Referencia de AWS CLI comandos.

## stop-logging

En el siguiente ejemplo de código, se muestra cómo usar `stop-logging`.

### AWS CLI

Para dejar de registrar un rastro

El siguiente `stop-logging` comando desactiva el registro de `Trail1`:

```
aws cloudtrail stop-logging --name Trail1
```

- Para API obtener más información, consulte [StopLogging](#) la Referencia de AWS CLI comandos.

## update-subscription

En el siguiente ejemplo de código, se muestra cómo usar `update-subscription`.

### AWS CLI

Para actualizar los ajustes de configuración de una ruta

El siguiente `update-subscription` comando actualiza la ruta para especificar un SNS tema y un bucket de S3 nuevos:

```
aws cloudtrail update-subscription --name Trail1 --s3-new-bucket my-bucket-new --  
sns-new-topic my-topic-new
```

Salida:

```
Setting up new S3 bucket my-bucket-new...  
Setting up new SNS topic my-topic-new...  
Creating/Updating CloudTrail configuration...  
CloudTrail configuration:  
{
```

```
"trailList": [  
  {  
    "IncludeGlobalServiceEvents": true,  
    "Name": "Trail1",  
    "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1",  
    "LogFileValidationEnabled": false,  
    "IsMultiRegionTrail": false,  
    "S3BucketName": "my-bucket-new",  
    "SnsTopicName": "my-topic-new",  
    "HomeRegion": "us-east-1"  
  }  
],  
"ResponseMetadata": {  
  "HTTPStatusCode": 200,  
  "RequestId": "31126f8a-c616-11e5-9cc6-2fd637936879"  
}  
}
```

- Para API obtener más información, consulte [UpdateSubscription](#) la Referencia de AWS CLI comandos.

## update-trail

En el siguiente ejemplo de código, se muestra cómo usar `update-trail`.

### AWS CLI

Para actualizar una ruta

El siguiente `update-trail` comando actualiza un rastro para usar un depósito existente para la entrega de registros:

```
aws cloudtrail update-trail --name Trail1 --s3-bucket-name my-bucket
```

Salida:

```
{  
  "IncludeGlobalServiceEvents": true,  
  "Name": "Trail1",  
  "TrailARN": "arn:aws:cloudtrail:us-west-2:123456789012:trail/Trail1",  
  "LogFileValidationEnabled": false,  
  "IsMultiRegionTrail": true,  
}
```

```
"S3BucketName": "my-bucket"
}
```

- Para API obtener más información, consulte [UpdateTrail](#) la Referencia de AWS CLI comandos.

## validate-logs

En el siguiente ejemplo de código, se muestra cómo usar `validate-logs`.

### AWS CLI

Para validar un archivo de registro

El siguiente `validate-logs` comando valida los registros de `Trail1`:

```
aws cloudtrail validate-logs --trail-arn arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1 --start-time 20160129T19:00:00Z
```

Salida:

```
Validating log files for trail arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1 between 2016-01-29T19:00:00Z and 2016-01-29T22:15:43Z
Results requested for 2016-01-29T19:00:00Z to 2016-01-29T22:15:43Z
Results found for 2016-01-29T19:24:57Z to 2016-01-29T21:24:57Z:
3/3 digest files valid
15/15 log files valid
```

- Para API obtener más información, consulte [ValidateLogs](#) la Referencia de AWS CLI comandos.

## CloudWatch ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with CloudWatch.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **delete-alarms**

En el siguiente ejemplo de código, se muestra cómo usar `delete-alarms`.

#### AWS CLI

Eliminación de una alarma

En el siguiente ejemplo, se utiliza el `delete-alarms` comando para eliminar la CloudWatch alarma de Amazon denominada «myalarm»:

```
aws cloudwatch delete-alarms --alarm-names myalarm
```

Salida:

```
This command returns to the prompt if successful.
```

- Para API obtener más información, consulte [DeleteAlarms](#) la Referencia de AWS CLI comandos.

### **describe-alarm-history**

En el siguiente ejemplo de código, se muestra cómo usar `describe-alarm-history`.

#### AWS CLI

Recuperación del historial de una alarma

En el siguiente ejemplo, se utiliza el `describe-alarm-history` comando para recuperar el historial de la CloudWatch alarma de Amazon denominada «myalarm»:

```
aws cloudwatch describe-alarm-history --alarm-name "myalarm" --history-item-type StateUpdate
```

Salida:

```

{
  "AlarmHistoryItems": [
    {
      "Timestamp": "2014-04-09T18:59:06.442Z",
      "HistoryItemType": "StateUpdate",
      "AlarmName": "myalarm",
      "HistoryData": "{\"version\":\"1.0\",\"oldState\":{\"stateValue\":\"ALARM\",\"stateReason\":\"testing purposes\"},\"newState\":{\"stateValue\":\"OK\",\"stateReason\":\"Threshold Crossed: 2 datapoints were not greater than the threshold (70.0). The most recent datapoints: [38.958, 40.292].\",\"stateReasonData\":{\"version\":\"1.0\",\"queryDate\":\"2014-04-09T18:59:06.419+0000\",\"startDate\":\"2014-04-09T18:44:00.000+0000\",\"statistic\":\"Average\",\"period\":300,\"recentDatapoints\":[38.958,40.292],\"threshold\":70.0}}}\",
      "HistorySummary": "Alarm updated from ALARM to OK"
    },
    {
      "Timestamp": "2014-04-09T18:59:05.805Z",
      "HistoryItemType": "StateUpdate",
      "AlarmName": "myalarm",
      "HistoryData": "{\"version\":\"1.0\",\"oldState\":{\"stateValue\":\"OK\",\"stateReason\":\"Threshold Crossed: 2 datapoints were not greater than the threshold (70.0). The most recent datapoints: [38.839999999999996, 39.714].\",\"stateReasonData\":{\"version\":\"1.0\",\"queryDate\":\"2014-03-11T22:45:41.569+0000\",\"startDate\":\"2014-03-11T22:30:00.000+0000\",\"statistic\":\"Average\",\"period\":300,\"recentDatapoints\":[38.839999999999996,39.714],\"threshold\":70.0}},\"newState\":{\"stateValue\":\"ALARM\",\"stateReason\":\"testing purposes\"}}\",
      "HistorySummary": "Alarm updated from OK to ALARM"
    }
  ]
}

```

- Para API obtener más información, consulte [DescribeAlarmHistory](#) la Referencia de AWS CLI comandos.

## describe-alarms-for-metric

En el siguiente ejemplo de código, se muestra cómo usar `describe-alarms-for-metric`.

### AWS CLI

Visualización de información sobre las alarmas asociadas a una métrica

En el siguiente ejemplo, se usa el `describe-alarms-for-metric` comando para mostrar información sobre cualquier alarma asociada a la EC2 CPUUtilization métrica de Amazon y a la instancia con el ID `i-0c986c72`.

```
aws cloudwatch describe-alarms-for-metric --metric-name CPUUtilization --
namespace AWS/EC2 --dimensions Name=InstanceId,Value=i-0c986c72
```

Salida:

```
{
  "MetricAlarms": [
    {
      "EvaluationPeriods": 10,
      "AlarmArn": "arn:aws:cloudwatch:us-
east-1:111122223333:alarm:myHighCpuAlarm2",
      "StateUpdatedTimestamp": "2013-10-30T03:03:51.479Z",
      "AlarmConfigurationUpdatedTimestamp": "2013-10-30T03:03:50.865Z",
      "ComparisonOperator": "GreaterThanOrEqualToThreshold",
      "AlarmActions": [
        "arn:aws:sns:us-east-1:111122223333:NotifyMe"
      ],
      "Namespace": "AWS/EC2",
      "AlarmDescription": "CPU usage exceeds 70 percent",
      "StateReasonData": "{\"version\":\"1.0\",\"queryDate\":
\"2013-10-30T03:03:51.479+0000\",\"startDate\":\"2013-10-30T02:08:00.000+0000\",
\"statistic\":\"Average\",\"period\":300,\"recentDatapoints\":
[40.698,39.612,42.432,39.796,38.816,42.28,42.854,40.088,40.760000000000005,41.316],
\"threshold\":70.0}",
      "Period": 300,
      "StateValue": "OK",
      "Threshold": 70.0,
      "AlarmName": "myHighCpuAlarm2",
      "Dimensions": [
        {
          "Name": "InstanceId",
          "Value": "i-0c986c72"
        }
      ],
      "Statistic": "Average",
      "StateReason": "Threshold Crossed: 10 datapoints were not greater than
or equal to the threshold (70.0). The most recent datapoints: [40.760000000000005,
41.316].",
      "InsufficientDataActions": [],
    }
  ]
}
```

```

    "OKActions": [],
    "ActionsEnabled": true,
    "MetricName": "CPUUtilization"
  },
  {
    "EvaluationPeriods": 2,
    "AlarmArn": "arn:aws:cloudwatch:us-
east-1:111122223333:alarm:myHighCpuAlarm",
    "StateUpdatedTimestamp": "2014-04-09T18:59:06.442Z",
    "AlarmConfigurationUpdatedTimestamp": "2014-04-09T22:26:05.958Z",
    "ComparisonOperator": "GreaterThanThreshold",
    "AlarmActions": [
      "arn:aws:sns:us-east-1:111122223333:HighCPUAlarm"
    ],
    "Namespace": "AWS/EC2",
    "AlarmDescription": "CPU usage exceeds 70 percent",
    "StateReasonData": "{\"version\":\"1.0\",\"queryDate\":
\\\"2014-04-09T18:59:06.419+0000\\\",\\\"startDate\\\":\\\"2014-04-09T18:44:00.000+0000\\\",
\\\"statistic\\\":\\\"Average\\\",\\\"period\\\":300,\\\"recentDatapoints\\\":[38.958,40.292],
\\\"threshold\\\":70.0}\",
    "Period": 300,
    "StateValue": "OK",
    "Threshold": 70.0,
    "AlarmName": "myHighCpuAlarm",
    "Dimensions": [
      {
        "Name": "InstanceId",
        "Value": "i-0c986c72"
      }
    ],
    "Statistic": "Average",
    "StateReason": "Threshold Crossed: 2 datapoints were not greater than
the threshold (70.0). The most recent datapoints: [38.958, 40.292].",
    "InsufficientDataActions": [],
    "OKActions": [],
    "ActionsEnabled": false,
    "MetricName": "CPUUtilization"
  }
]
}

```

- Para API obtener más información, consulte [DescribeAlarmsForMetric](#) la Referencia de AWS CLI comandos.

## describe-alarms

En el siguiente ejemplo de código, se muestra cómo usar `describe-alarms`.

### AWS CLI

Visualización de información acerca de una alarma

En el siguiente ejemplo, se utiliza el comando `describe-alarms` para proporcionar información sobre la alarma denominada "mialarma":

```
aws cloudwatch describe-alarms --alarm-names "myAlarm"
```

Salida:

```
{
  "MetricAlarms": [
    {
      "EvaluationPeriods": 2,
      "AlarmArn": "arn:aws:cloudwatch:us-east-1:123456789012:alarm:myalarm",
      "StateUpdatedTimestamp": "2014-04-09T18:59:06.442Z",
      "AlarmConfigurationUpdatedTimestamp": "2012-12-27T00:49:54.032Z",
      "ComparisonOperator": "GreaterThanThreshold",
      "AlarmActions": [
        "arn:aws:sns:us-east-1:123456789012:myHighCpuAlarm"
      ],
      "Namespace": "AWS/EC2",
      "AlarmDescription": "CPU usage exceeds 70 percent",
      "StateReasonData": "{\"version\":\"1.0\",\"queryDate\":\
\\\"2014-04-09T18:59:06.419+0000\\\",\\\"startDate\\\":\\\"2014-04-09T18:44:00.000+0000\\\",
\\\"statistic\\\":\\\"Average\\\",\\\"period\\\":300,\\\"recentDatapoints\\\":[38.958,40.292],
\\\"threshold\\\":70.0}\",
      "Period": 300,
      "StateValue": "OK",
      "Threshold": 70.0,
      "AlarmName": "myalarm",
      "Dimensions": [
        {
          "Name": "InstanceId",
          "Value": "i-0c986c72"
        }
      ],
      "Statistic": "Average",
    }
  ]
}
```



```
        "StateReason": "Threshold Crossed: 2 datapoints were not greater than  
the threshold (70.0). The most recent datapoints: [38.958, 40.292].",  
        "InsufficientDataActions": [],  
        "OKActions": [],  
        "ActionsEnabled": true,  
        "MetricName": "CPUUtilization"  
    }  
]  
}
```

- Para API obtener más información, consulte [DescribeAlarms](#) la Referencia de AWS CLI comandos.

## disable-alarm-actions

En el siguiente ejemplo de código, se muestra cómo usar `disable-alarm-actions`.

### AWS CLI

Desactivación de acciones en una alarma

En el siguiente ejemplo, se utiliza el comando `disable-alarm-actions` para deshabilitar todas las acciones de la alarma denominada `mialarma`:

```
aws cloudwatch disable-alarm-actions --alarm-names myAlarm
```

Este comando vuelve a la petición si se ejecuta correctamente.

- Para API obtener más información, consulte [DisableAlarmActions](#) la Referencia de AWS CLI comandos.

## enable-alarm-actions

En el siguiente ejemplo de código, se muestra cómo usar `enable-alarm-actions`.

### AWS CLI

Activación de todas las acciones de una alarma

En el siguiente ejemplo, se utiliza el comando `enable-alarm-actions` para activar todas las acciones de la alarma denominada `myalarm`.

```
aws cloudwatch enable-alarm-actions --alarm-names myAlarm
```

Este comando vuelve a la petición si se ejecuta correctamente.

- Para API obtener más información, consulte [EnableAlarmActions](#) la Referencia de AWS CLI comandos.

## get-metric-statistics

En el siguiente ejemplo de código, se muestra cómo usar `get-metric-statistics`.

### AWS CLI

Para obtener la CPU utilización por EC2 instancia

En el siguiente ejemplo, se usa el `get-metric-statistics` comando para obtener la CPU utilización de una EC2 instancia con el identificador `i-abcdef`.

```
aws cloudwatch get-metric-statistics --metric-name CPUUtilization --start-time 2014-04-08T23:18:00Z --end-time 2014-04-09T23:18:00Z --period 3600 --namespace AWS/EC2 --statistics Maximum --dimensions Name=InstanceId,Value=i-abcdef
```

Salida:

```
{
  "Datapoints": [
    {
      "Timestamp": "2014-04-09T11:18:00Z",
      "Maximum": 44.79,
      "Unit": "Percent"
    },
    {
      "Timestamp": "2014-04-09T20:18:00Z",
      "Maximum": 47.92,
      "Unit": "Percent"
    },
    {
      "Timestamp": "2014-04-09T19:18:00Z",
      "Maximum": 50.85,
      "Unit": "Percent"
    },
    {
```

```
    "Timestamp": "2014-04-09T09:18:00Z",
    "Maximum": 47.92,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T03:18:00Z",
    "Maximum": 76.84,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T21:18:00Z",
    "Maximum": 48.96,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T14:18:00Z",
    "Maximum": 47.92,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T08:18:00Z",
    "Maximum": 47.92,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T16:18:00Z",
    "Maximum": 45.55,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T06:18:00Z",
    "Maximum": 47.92,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T13:18:00Z",
    "Maximum": 45.08,
    "Unit": "Percent"
  },
  {
    "Timestamp": "2014-04-09T05:18:00Z",
    "Maximum": 47.92,
    "Unit": "Percent"
  },
  },
```

```
{
  "Timestamp": "2014-04-09T18:18:00Z",
  "Maximum": 46.88,
  "Unit": "Percent"
},
{
  "Timestamp": "2014-04-09T17:18:00Z",
  "Maximum": 52.08,
  "Unit": "Percent"
},
{
  "Timestamp": "2014-04-09T07:18:00Z",
  "Maximum": 47.92,
  "Unit": "Percent"
},
{
  "Timestamp": "2014-04-09T02:18:00Z",
  "Maximum": 51.23,
  "Unit": "Percent"
},
{
  "Timestamp": "2014-04-09T12:18:00Z",
  "Maximum": 47.67,
  "Unit": "Percent"
},
{
  "Timestamp": "2014-04-08T23:18:00Z",
  "Maximum": 46.88,
  "Unit": "Percent"
},
{
  "Timestamp": "2014-04-09T10:18:00Z",
  "Maximum": 51.91,
  "Unit": "Percent"
},
{
  "Timestamp": "2014-04-09T04:18:00Z",
  "Maximum": 47.13,
  "Unit": "Percent"
},
{
  "Timestamp": "2014-04-09T15:18:00Z",
  "Maximum": 48.96,
  "Unit": "Percent"
}
```

```

    },
    {
      "Timestamp": "2014-04-09T00:18:00Z",
      "Maximum": 48.16,
      "Unit": "Percent"
    },
    {
      "Timestamp": "2014-04-09T01:18:00Z",
      "Maximum": 49.18,
      "Unit": "Percent"
    }
  ],
  "Label": "CPUUtilization"
}

```

### Especificación de varias dimensiones

En el siguiente ejemplo, se ilustra cómo especificar varias dimensiones. Cada dimensión se especifica mediante un par nombre/valor, con una coma entre el nombre y el valor. Cuando existen varias dimensiones se separan con un espacio. Si una métrica incluye varias dimensiones, debe especificar un valor para cada dimensión definida.

Para ver más ejemplos del uso del `get-metric-statistics` comando, consulta [Obtener estadísticas para una métrica en la Guía para CloudWatch desarrolladores de Amazon](#).

```

aws cloudwatch get-metric-statistics --metric-name Buffers --namespace MyNameSpace
  --dimensions Name=InstanceID,Value=i-abcdef Name=InstanceType,Value=m1.small --
start-time 2016-10-15T04:00:00Z --end-time 2016-10-19T07:00:00Z --statistics Average
  --period 60

```

- Para API obtener más información, consulte [GetMetricStatistics](#) la Referencia de AWS CLI comandos.

### **list-metrics**

En el siguiente ejemplo de código, se muestra cómo usar `list-metrics`.

#### AWS CLI

Para enumerar las métricas de Amazon SNS

En el siguiente `list-metrics` ejemplo, se muestran las métricas de Amazon SNS.

```
aws cloudwatch list-metrics \  
  --namespace "AWS/SNS"
```

Salida:

```
{  
  "Metrics": [  
    {  
      "Namespace": "AWS/SNS",  
      "Dimensions": [  
        {  
          "Name": "TopicName",  
          "Value": "NotifyMe"  
        }  
      ],  
      "MetricName": "PublishSize"  
    },  
    {  
      "Namespace": "AWS/SNS",  
      "Dimensions": [  
        {  
          "Name": "TopicName",  
          "Value": "CF0"  
        }  
      ],  
      "MetricName": "PublishSize"  
    },  
    {  
      "Namespace": "AWS/SNS",  
      "Dimensions": [  
        {  
          "Name": "TopicName",  
          "Value": "NotifyMe"  
        }  
      ],  
      "MetricName": "NumberOfNotificationsFailed"  
    },  
    {  
      "Namespace": "AWS/SNS",  
      "Dimensions": [  
        {  
          "Name": "TopicName",  
          "Value": "NotifyMe"  
        }  
      ]  
    }  
  ]  
}
```

```
    }
  ],
  "MetricName": "NumberOfNotificationsDelivered"
},
{
  "Namespace": "AWS/SNS",
  "Dimensions": [
    {
      "Name": "TopicName",
      "Value": "NotifyMe"
    }
  ],
  "MetricName": "NumberOfMessagesPublished"
},
{
  "Namespace": "AWS/SNS",
  "Dimensions": [
    {
      "Name": "TopicName",
      "Value": "CF0"
    }
  ],
  "MetricName": "NumberOfMessagesPublished"
},
{
  "Namespace": "AWS/SNS",
  "Dimensions": [
    {
      "Name": "TopicName",
      "Value": "CF0"
    }
  ],
  "MetricName": "NumberOfNotificationsDelivered"
},
{
  "Namespace": "AWS/SNS",
  "Dimensions": [
    {
      "Name": "TopicName",
      "Value": "CF0"
    }
  ],
  "MetricName": "NumberOfNotificationsFailed"
}
```

```
]
}
```

- Para API obtener más información, consulte [ListMetrics](#) la Referencia de AWS CLI comandos.

## put-metric-alarm

En el siguiente ejemplo de código, se muestra cómo usar `put-metric-alarm`.

### AWS CLI

Para enviar un mensaje de correo electrónico de Amazon Simple Notification Service cuando el CPU uso supere el 70 por ciento

El siguiente ejemplo utiliza el `put-metric-alarm` comando para enviar un mensaje de correo electrónico de Amazon Simple Notification Service cuando el CPU uso supera el 70 por ciento:

```
aws cloudwatch put-metric-alarm --alarm-name cpu-mon --alarm-description "Alarm when CPU exceeds 70 percent" --metric-name CPUUtilization --namespace AWS/EC2 --statistic Average --period 300 --threshold 70 --comparison-operator GreaterThanThreshold --dimensions "Name=InstanceId,Value=i-12345678" --evaluation-periods 2 --alarm-actions arn:aws:sns:us-east-1:111122223333:MyTopic --unit Percent
```

Este comando vuelve a la petición si se ejecuta correctamente. Si existe una alarma con el mismo nombre, la alarma nueva la sobrescribirá.

### Especificación de varias dimensiones

En el siguiente ejemplo, se ilustra cómo especificar varias dimensiones. Cada dimensión se especifica mediante un par nombre/valor, con una coma entre el nombre y el valor. Cuando existen varias dimensiones se separan con un espacio:

```
aws cloudwatch put-metric-alarm --alarm-name "Default_Test_Alarm3" --alarm-description "The default example alarm" --namespace "CW EXAMPLE METRICS" --metric-name Default_Test --statistic Average --period 60 --evaluation-periods 3 --threshold 50 --comparison-operator GreaterThanOrEqualToThreshold --dimensions Name=key1,Value=value1 Name=key2,Value=value2
```

- Para API obtener más información, consulte [PutMetricAlarm](#) la Referencia de AWS CLI comandos.



## put-metric-data

En el siguiente ejemplo de código, se muestra cómo usar `put-metric-data`.

### AWS CLI

Para publicar una métrica personalizada en Amazon CloudWatch

En el siguiente ejemplo, se utiliza el `put-metric-data` comando para publicar una métrica personalizada en Amazon CloudWatch:

```
aws cloudwatch put-metric-data --namespace "Usage Metrics" --metric-data file://metric.json
```

Los valores de la métrica en sí se almacenan en el JSON archivo, `metric.json`.

A continuación, se muestra el contenido de ese archivo:

```
[
  {
    "MetricName": "New Posts",
    "Timestamp": "Wednesday, June 12, 2013 8:28:20 PM",
    "Value": 0.50,
    "Unit": "Count"
  }
]
```

Para obtener más información, consulte [Publicar métricas personalizadas en la Guía para CloudWatch desarrolladores de Amazon](#).

### Cómo especificar varias dimensiones

En el siguiente ejemplo, se ilustra cómo especificar varias dimensiones. Cada dimensión se especifica con un par `Nombre=Valor`. Cuando existen varias dimensiones se separan con una coma:

```
aws cloudwatch put-metric-data --metric-name Buffers --
namespace MyNameSpace --unit Bytes --value 231434333 --
dimensions InstanceID=1-23456789,InstanceType=m1.small
```

- Para API obtener más información, consulte [PutMetricData](#) la Referencia de AWS CLI comandos.

## set-alarm-state

En el siguiente ejemplo de código, se muestra cómo usar `set-alarm-state`.

### AWS CLI

Para cambiar temporalmente el estado de una alarma

En el siguiente ejemplo, se utiliza el `set-alarm-state` comando para cambiar temporalmente el estado de una CloudWatch alarma de Amazon denominada «myalarm» y establecerla en ese ALARM estado con fines de prueba:

```
aws cloudwatch set-alarm-state --alarm-name "myalarm" --state-value ALARM --state-reason "testing purposes"
```

Este comando vuelve a la petición si se ejecuta correctamente.

- Para API obtener más información, consulte [SetAlarmState](#) la Referencia de AWS CLI comandos.

## CloudWatch Registra ejemplos usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso AWS Command Line Interface de CloudWatch registros.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

#### create-log-group

En el siguiente ejemplo de código, se muestra cómo usar `create-log-group`.

## AWS CLI

El siguiente comando crea un grupo de registro denominado `my-logs`:

```
aws logs create-log-group --log-group-name my-logs
```

- Para API obtener más información, consulte [CreateLogGroup](#) la Referencia de AWS CLI comandos.

### **create-log-stream**

En el siguiente ejemplo de código, se muestra cómo usar `create-log-stream`.

## AWS CLI

El siguiente comando crea un flujo de registro denominado `20150601` en el grupo de registro `my-logs`:

```
aws logs create-log-stream --log-group-name my-logs --log-stream-name 20150601
```

- Para API obtener más información, consulte [CreateLogStream](#) la Referencia de AWS CLI comandos.

### **delete-log-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-log-group`.

## AWS CLI

El siguiente comando elimina un grupo de registro denominado `my-logs`:

```
aws logs delete-log-group --log-group-name my-logs
```

- Para API obtener más información, consulte [DeleteLogGroup](#) la Referencia de AWS CLI comandos.

### **delete-log-stream**

En el siguiente ejemplo de código, se muestra cómo usar `delete-log-stream`.

## AWS CLI

El siguiente comando elimina un flujo de registro denominado `20150531` de un grupo de registros denominado `my-logs`:

```
aws logs delete-log-stream --log-group-name my-logs --log-stream-name 20150531
```

- Para API obtener más información, consulte [DeleteLogStream](#) la Referencia de AWS CLI comandos.

## `delete-retention-policy`

En el siguiente ejemplo de código, se muestra cómo usar `delete-retention-policy`.

## AWS CLI

El siguiente comando elimina la política de retención que se aplicó anteriormente a un grupo de registros denominado `my-logs`:

```
aws logs delete-retention-policy --log-group-name my-logs
```

- Para API obtener más información, consulte [DeleteRetentionPolicy](#) la Referencia de AWS CLI comandos.

## `describe-log-groups`

En el siguiente ejemplo de código, se muestra cómo usar `describe-log-groups`.

## AWS CLI

El siguiente comando describe un grupo de registro denominado `my-logs`:

```
aws logs describe-log-groups --log-group-name-prefix my-logs
```

Salida:

```
{
  "logGroups": [
```

```
{
  "storedBytes": 0,
  "metricFilterCount": 0,
  "creationTime": 1433189500783,
  "logGroupName": "my-logs",
  "retentionInDays": 5,
  "arn": "arn:aws:logs:us-west-2:0123456789012:log-group:my-logs:*"
}
]
```

- Para API obtener más información, consulte [DescribeLogGroups](#) la Referencia de AWS CLI comandos.

## describe-log-streams

En el siguiente ejemplo de código, se muestra cómo usar `describe-log-streams`.

### AWS CLI

El siguiente comando muestra todos los flujos de registro que comienzan con el prefijo `2015` del grupo `my-logs` de registros:

```
aws logs describe-log-streams --log-group-name my-logs --log-stream-name-prefix 2015
```

Salida:

```
{
  "logStreams": [
    {
      "creationTime": 1433189871774,
      "arn": "arn:aws:logs:us-west-2:0123456789012:log-group:my-logs:log-stream:20150531",
      "logStreamName": "20150531",
      "storedBytes": 0
    },
    {
      "creationTime": 1433189873898,
      "arn": "arn:aws:logs:us-west-2:0123456789012:log-group:my-logs:log-stream:20150601",
      "logStreamName": "20150601",
      "storedBytes": 0
    }
  ]
}
```

```
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeLogStreams](#) la Referencia de AWS CLI comandos.

## get-log-events

En el siguiente ejemplo de código, se muestra cómo usar `get-log-events`.

### AWS CLI

El siguiente comando recupera los eventos de registro de un flujo de registro denominado `20150601` en el grupo `my-logs` de registros:

```
aws logs get-log-events --log-group-name my-logs --log-stream-name 20150601
```

Salida:

```
{
  "nextForwardToken":
  "f/31961209122447488583055879464742346735121166569214640130",
  "events": [
    {
      "ingestionTime": 1433190494190,
      "timestamp": 1433190184356,
      "message": "Example Event 1"
    },
    {
      "ingestionTime": 1433190516679,
      "timestamp": 1433190184356,
      "message": "Example Event 1"
    },
    {
      "ingestionTime": 1433190494190,
      "timestamp": 1433190184358,
      "message": "Example Event 2"
    }
  ],
  "nextBackwardToken":
  "b/31961209122358285602261756944988674324553373268216709120"
```

```
}
```

- Para API obtener más información, consulte [GetLogEvents](#) la Referencia de AWS CLI comandos.

## put-log-events

En el siguiente ejemplo de código, se muestra cómo usar `put-log-events`.

### AWS CLI

El siguiente comando coloca los eventos de registro en un flujo de registro denominado `20150601` en el grupo de registros `my-logs`:

```
aws logs put-log-events --log-group-name my-logs --log-stream-name 20150601 --log-events file://events
```

Salida:

```
{
  "nextSequenceToken": "49542672486831074009579604567656788214806863282469607346"
}
```

El ejemplo anterior lee una JSON matriz de eventos de un archivo nombrado `events` en el directorio actual:

```
[
  {
    "timestamp": 1433190184356,
    "message": "Example Event 1"
  },
  {
    "timestamp": 1433190184358,
    "message": "Example Event 2"
  },
  {
    "timestamp": 1433190184360,
    "message": "Example Event 3"
  }
]
```

Cada llamada posterior requiere que el siguiente token de secuencia proporcionado por la llamada anterior se especifique con la opción de token de secuencia:

```
aws logs put-log-events --log-group-name my-logs --log-  
stream-name 20150601 --log-events file://events2 --sequence-  
token "49542672486831074009579604567656788214806863282469607346"
```

Salida:

```
{  
  "nextSequenceToken": "49542672486831074009579604567900991230369019956308219826"  
}
```

- Para API obtener más información, consulte [PutLogEvents](#) la Referencia de AWS CLI comandos.

## put-retention-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-retention-policy`.

AWS CLI

El siguiente comando agrega una política de retención de 5 días a un grupo de registros denominado `my-logs`:

```
aws logs put-retention-policy --log-group-name my-logs --retention-in-days 5
```

- Para API obtener más información, consulte [PutRetentionPolicy](#) la Referencia de AWS CLI comandos.

## CloudWatch Ejemplos de monitoreo de red usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante el uso de la AWS Command Line Interface supervisión CloudWatch de red.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.



Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **create-monitor**

En el siguiente ejemplo de código, se muestra cómo usar `create-monitor`.

#### AWS CLI

Ejemplo 1: Para crear un monitor de red con un período de agregación

En el siguiente `create-monitor` ejemplo, se crea un monitor `Example_NetworkMonitor` con un nombre `aggregationPeriod` establecido en 30 segundos. La inicial `state` del monitor se `INACTIVE` debe a que no tiene sondas asociadas. El estado cambia `ACTIVE` solo cuando se agregan sondas. Puede utilizar los comandos [update-monitor](#) o [create-probe](#) para añadir sondas a este monitor.

```
aws networkmonitor create-monitor \  
  --monitor-name Example_NetworkMonitor \  
  --aggregation-period 30
```

Salida:

```
{  
  "monitorArn": "arn:aws:networkmonitor:region:111122223333:monitor/  
Example_NetworkMonitor",  
  "monitorName": "Example_NetworkMonitor",  
  "state": "INACTIVE",  
  "aggregationPeriod": 30,  
  "tags": {}  
}
```

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

Ejemplo 2: Para crear un monitor de red con una sonda que también incluya etiquetas TCP

En el siguiente `create-monitor` ejemplo se crea un monitor denominado `Example_NetworkMonitor`. El comando también crea una sonda que utiliza el ICMP protocolo e incluye etiquetas. Como no `aggregationPeriod` se pasa en la solicitud, los 60 segundos se establecen como valores predeterminados. El `state` del monitor con la sonda será el mismo `PENDING` hasta que el monitor lo esté `ACTIVE`. Esto puede tardar varios minutos, momento en el que `state` pasará a ser `ACTIVE`, y podrás empezar a ver CloudWatch las métricas.

```
aws networkmonitor create-monitor \  
  --monitor-name Example_NetworkMonitor \  
  --probes sourceArn=arn:aws:ec2:region:111122223333:subnet/subnet-  
id,destination=10.0.0.100,destinationPort=80,protocol=TCP,packetSize=56,probeTags={Name=Prob  
 \  
  --tags Monitor=Monitor1
```

Salida:

```
{  
  "monitorArn": "arn:aws:networkmonitor:region111122223333:monitor/  
Example_NetworkMonitor",  
  "monitorName": "Example_NetworkMonitor",  
  "state": "PENDING",  
  "aggregationPeriod": 60,  
  "tags": {  
    "Monitor": "Monitor1"  
  }  
}
```

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

Ejemplo 3: Para crear un monitor de red con una sonda que también incluya etiquetas ICMP

El siguiente `create-monitor` ejemplo crea un monitor denominado `Example_NetworkMonitor` con un número `aggregationPeriod` de 30 segundos. El comando también crea una sonda que usa el ICMP protocolo e incluye etiquetas. Como no `aggregationPeriod` se pasa en la solicitud, los 60 segundos se establecen como valores predeterminados. El `state` del monitor con la sonda será el mismo `PENDING` hasta que el monitor lo esté `ACTIVE`. Esto puede tardar varios minutos, momento en el que `state` pasará a ser `ACTIVE`, y podrás empezar a ver CloudWatch las métricas.

```
aws networkmonitor create-monitor \  
  --monitor-name Example_NetworkMonitor \  
  --aggregation-period 30 \  
  --probes sourceArn=arn:aws:ec2:region111122223333:subnet/subnet-  
id,destination=10.0.0.100,protocol=ICMP,packetSize=56,probeTags={Name=Probe1} \  
  --tags Monitor=Monitor1
```

Salida:

```
{  
  "monitorArn": "arn:aws:networkmonitor:region:111122223333:monitor/  
Example_NetworkMonitor",  
  "monitorName": "Example_NetworkMonitor",  
  "state": "PENDING",  
  "aggregationPeriod": 30,  
  "tags": {  
    "Monitor": "Monitor1"  
  }  
}
```

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

- Para API obtener más información, consulte [CreateMonitor](#) la Referencia de AWS CLI comandos.

## create-probe

En el siguiente ejemplo de código, se muestra cómo usar create-probe.

### AWS CLI

Ejemplo 1: Para crear una sonda que la utilice TCP y añada a un monitor de red

En el siguiente create-probe ejemplo, se crea una sonda que utiliza la sonda TCP protocol y la añade a un monitor denominado Example\_NetworkMonitor. Una vez creada, la longitud state del monitor con la sonda será igual PENDING a la del monitor ACTIVE. Esto puede tardar varios minutos, momento en el que el estado cambiará a ACTIVE, y podrás empezar a ver CloudWatch las métricas.

```
aws networkmonitor create-probe \  

```

```
--monitor-name Example_NetworkMonitor \
--probe sourceArn=arn:aws:ec2:region:111122223333:subnet/subnet-
id,destination=10.0.0.100,destinationPort=80,protocol=TCP,packetSize=56,tags={Name=Probe1}
```

Salida:

```
{
  "probeId": "probe-12345",
  "probeArn": "arn:aws:networkmonitor:region:111122223333:probe/probe-12345",
  "destination": "10.0.0.100",
  "destinationPort": 80,
  "packetSize": 56,
  "addressFamily": "IPV4",
  "vpcId": "vpc-12345",
  "state": "PENDING",
  "createdAt": "2024-03-29T12:41:57.314000-04:00",
  "modifiedAt": "2024-03-29T12:41:57.314000-04:00",
  "tags": {
    "Name": "Probe1"
  }
}
```

Ejemplo 2: Para crear una sonda que utilice una sonda ICMP y añadirla a un monitor de red

En el siguiente `create-probe` ejemplo, se crea una sonda que utiliza la sonda ICMP protocol y la añade a un monitor denominado `Example_NetworkMonitor`. Una vez creada, la longitud state del monitor con la sonda será igual `PENDING` a la del monitor `ACTIVE`. Esto puede tardar varios minutos, momento en el que el estado cambiará a `ACTIVE`, y podrás empezar a ver CloudWatch las métricas.

```
aws networkmonitor create-probe \
--monitor-name Example_NetworkMonitor \
--probe sourceArn=arn:aws:ec2:region:012345678910:subnet/subnet-
id,destination=10.0.0.100,protocol=ICMP,packetSize=56,tags={Name=Probe1}
```

Salida:

```
{
  "probeId": "probe-12345",
  "probeArn": "arn:aws:networkmonitor:region:111122223333:probe/probe-12345",
  "destination": "10.0.0.100",
```

```
"packetSize": 56,  
"addressFamily": "IPv4",  
"vpcId": "vpc-12345",  
"state": "PENDING",  
"createdAt": "2024-03-29T12:44:02.452000-04:00",  
"modifiedAt": "2024-03-29T12:44:02.452000-04:00",  
"tags": {  
  "Name": "Probe1"  
}  
}
```

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

- Para API obtener más información, consulte [CreateProbe](#) la Referencia de AWS CLI comandos.

## delete-monitor

En el siguiente ejemplo de código, se muestra cómo usar `delete-monitor`.

### AWS CLI

Para eliminar un monitor

En el siguiente `delete-monitor` ejemplo, se elimina un monitor denominado `Example_NetworkMonitor`.

```
aws networkmonitor delete-monitor \  
  --monitor-name Example_NetworkMonitor
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

- Para API obtener más información, consulte [DeleteMonitor](#) la Referencia de AWS CLI comandos.

## delete-probe

En el siguiente ejemplo de código, se muestra cómo usar `delete-probe`.

## AWS CLI

Para eliminar una sonda

En el siguiente delete-probe ejemplo, se elimina una sonda con el ID probe-12345 de un monitor de red denominado Example\_NetworkMonitor.

```
aws networkmonitor delete-probe \  
  --monitor-name Example_NetworkMonitor \  
  --probe-id probe-12345
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

- Para API obtener más información, consulte [DeleteProbe](#) la Referencia de AWS CLI comandos.

## get-monitor

En el siguiente ejemplo de código, se muestra cómo usar get-monitor.

## AWS CLI

Para obtener información sobre el monitor

En el siguiente get-monitor ejemplo se obtiene información sobre un monitor denominado Example\_NetworkMonitor.

```
aws networkmonitor get-monitor \  
  --monitor-name Example_NetworkMonitor
```

Salida:

```
{  
  "monitorArn": "arn:aws:networkmonitor:region:012345678910:monitor/  
Example_NetworkMonitor",  
  "monitorName": "Example_NetworkMonitor",  
  "state": "ACTIVE",  
  "aggregationPeriod": 60,  
  "tags": {},
```

```
"probes": [],  
"createdAt": "2024-04-01T17:58:07.211000-04:00",  
"modifiedAt": "2024-04-01T17:58:07.211000-04:00"  
}
```

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

- Para API obtener más información, consulte [GetMonitor](#) la Referencia de AWS CLI comandos.

## get-probe

En el siguiente ejemplo de código, se muestra cómo usar `get-probe`.

### AWS CLI

Para ver los detalles de la sonda

El siguiente `get-probe` ejemplo devuelve detalles sobre una sonda con `probeID` `probe-12345` el nombre asociado a un monitor `Example_NetworkMonitor`.

```
aws networkmonitor get-probe \  
  --monitor-name Example_NetworkMonitor \  
  --probe-id probe-12345
```

Salida:

```
{  
  "probeId": "probe-12345",  
  "probeArn": "arn:aws:networkmonitor:region:012345678910:probe/probe-12345",  
  "sourceArn": "arn:aws:ec2:region:012345678910:subnet/subnet-12345",  
  "destination": "10.0.0.100",  
  "destinationPort": 80,  
  "protocol": "TCP",  
  "packetSize": 56,  
  "addressFamily": "IPV4",  
  "vpcId": "vpc-12345",  
  "state": "ACTIVE",  
  "createdAt": "2024-03-29T12:41:57.314000-04:00",  
  "modifiedAt": "2024-03-29T12:42:28.610000-04:00",  
  "tags": {  
    "Name": "Probe1"  
  }  
}
```

```
}  
}
```

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

- Para API obtener más información, consulte [GetProbela](#) Referencia de AWS CLI comandos.

## list-monitors

En el siguiente ejemplo de código, se muestra cómo usar `list-monitors`.

### AWS CLI

Ejemplo 1: Para enumerar todos los monitores (un solo monitor)

El siguiente `list-monitors` ejemplo devuelve una lista de un solo monitor. El monitor state es ACTIVE y tiene un tiempo aggregationPeriod de 60 segundos.

```
aws networkmonitor list-monitors
```

Salida:

```
{  
  "monitors": [  
    {  
      "monitorArn": "arn:aws:networkmonitor:region:012345678910:monitor/  
Example_NetworkMonitor",  
      "monitorName": "Example_NetworkMonitor",  
      "state": "ACTIVE",  
      "aggregationPeriod": 60,  
      "tags": {  
        "Monitor": "Monitor1"  
      }  
    }  
  ]  
}
```

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

Ejemplo 2: Para enumerar todos los monitores (varios monitores)



El siguiente `list-monitors` ejemplo devuelve una lista de tres monitores. El `state` de un monitor es el `ACTIVE` generador de CloudWatch métricas. Los estados de los otros dos monitores son `INACTIVE` y no generan CloudWatch métricas. Los tres monitores utilizan una duración `aggregationPeriod` de 60 segundos.

```
aws networkmonitor list-monitors
```

Salida:

```
{
  "monitors": [
    {
      "monitorArn": "arn:aws:networkmonitor:us-east-1:111122223333:monitor/
Example_NetworkMonitor",
      "monitorName": "Example_NetworkMonitor",
      "state": "INACTIVE",
      "aggregationPeriod": 60,
      "tags": {}
    },
    {
      "monitorArn": "arn:aws:networkmonitor:us-east-1:111122223333:monitor/
Example_NetworkMonitor2",
      "monitorName": "Example_NetworkMonitor2",
      "state": "ACTIVE",
      "aggregationPeriod": 60,
      "tags": {
        "Monitor": "Monitor1"
      }
    },
    {
      "monitorArn": "arn:aws:networkmonitor:us-east-1:111122223333:monitor/
TestNetworkMonitor_CLI",
      "monitorName": "TestNetworkMonitor_CLI",
      "state": "INACTIVE",
      "aggregationPeriod": 60,
      "tags": {}
    }
  ]
}
```

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

- Para API obtener más información, consulte [ListMonitors](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un recurso

El siguiente `list-tags-for-resource` ejemplo devuelve una lista de las etiquetas de un monitor denominado `Example_NetworkMonitor`.

```
aws networkmonitor list-tags-for-resource \  
  --resource-arn arn:aws:networkmonitor:region:012345678910:monitor/  
Example_NetworkMonitor
```

Salida:

```
{  
  "tags": {  
    "Environment": "Dev",  
    "Application": "PetStore"  
  }  
}
```

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar un recurso

En el siguiente `tag-resource` ejemplo, se etiqueta un monitor denominado `Example_NetworkMonitor` con `Environment=Dev` y `Application=PetStore` etiquetas.

```
aws networkmonitor tag-resource \  
  --resource-arn arn:aws:networkmonitor:region:012345678910:monitor/  
Example_NetworkMonitor \  
  --tags Environment=Dev,Application=PetStore
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar untag-resource.

### AWS CLI

Para quitar la etiqueta de un recurso

En el siguiente untag-resource ejemplo, se elimina un tag-keys parámetro con el par clave-valor Environment Application de su asociación con un monitor denominado Example\_NetworkMonitor

```
aws networkmonitor untag-resource \  
  --resource-arn arn:aws:networkmonitor:region:012345678910:monitor/  
Example_NetworkMonitor \  
  --tag-keys Environment Application
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-monitor

En el siguiente ejemplo de código, se muestra cómo usar update-monitor.

## AWS CLI

Para actualizar un monitor

En el siguiente `update-monitor` ejemplo, se cambia el tamaño `aggregationPeriod` de un monitor de 60 segundos a 30 segundos.

```
aws networkmonitor update-monitor \  
  --monitor-name Example_NetworkMonitor \  
  --aggregation-period 30
```

Salida:

```
{  
  "monitorArn": "arn:aws:networkmonitor:region:012345678910:monitor/  
Example_NetworkMonitor",  
  "monitorName": "Example_NetworkMonitor",  
  "state": "PENDING",  
  "aggregationPeriod": 30,  
  "tags": {  
    "Monitor": "Monitor1"  
  }  
}
```

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

- Para API obtener más información, consulte [UpdateMonitor](#) la Referencia de AWS CLI comandos.

## update-probe

En el siguiente ejemplo de código, se muestra cómo usar `update-probe`.

## AWS CLI

Para actualizar una sonda

El siguiente `update-probe` ejemplo actualiza la dirección `destination` IP original de una sonda y también actualiza la dirección `packetSize` de 60.

```
aws networkmonitor update-probe \  
  --monitor-name Example_NetworkMonitor \  
  --probe-id probe-12345 \  
  --destination 10.0.0.150 \  
  --packet-size 60
```

Salida:

```
{  
  "probeId": "probe-12345",  
  "probeArn": "arn:aws:networkmonitor:region:012345678910:probe/probe-12345",  
  "sourceArn": "arn:aws:ec2:region:012345678910:subnet/subnet-12345",  
  "destination": "10.0.0.150",  
  "destinationPort": 80,  
  "protocol": "TCP",  
  "packetSize": 60,  
  "addressFamily": "IPV4",  
  "vpcId": "vpc-12345",  
  "state": "PENDING",  
  "createdAt": "2024-03-29T12:41:57.314000-04:00",  
  "modifiedAt": "2024-03-29T13:52:23.115000-04:00",  
  "tags": {  
    "Name": "Probe1"  
  }  
}
```

Para obtener más información, consulte [Cómo funciona Amazon CloudWatch Network Monitor](#) en la Guía del CloudWatch usuario de Amazon.

- Para API obtener más información, consulte [UpdateProbe](#) la Referencia de AWS CLI comandos.

## CodeArtifact ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with CodeArtifact.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **associate-external-connection**

En el siguiente ejemplo de código, se muestra cómo usar `associate-external-connection`.

#### AWS CLI

Para añadir una conexión externa a un repositorio

En el siguiente `associate-external-connection` ejemplo, se agrega una conexión externa a `npmjs.com` a un repositorio denominado `test-repo`.

```
aws codeartifact associate-external-connection \  
  --repository test-repo \  
  --domain test-domain \  
  --external-connection public:npmjs
```

Salida:

```
{  
  "repository": {  
    "name": "test-repo",  
    "administratorAccount": "111122223333",  
    "domainName": "test-domain",  
    "domainOwner": "111122223333",  
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/  
test-repo",  
    "upstreams": [],  
    "externalConnections": [  
      {  
        "externalConnectionName": "public:npmjs",  
        "packageFormat": "npm",  
        "status": "AVAILABLE"  
      }  
    ]  
  }  
}
```

```
}  
}
```

Para obtener más información, consulte [Añadir una conexión externa](#) en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [AssociateExternalConnection](#) la Referencia de AWS CLI comandos.

## copy-package-versions

En el siguiente ejemplo de código, se muestra cómo usar `copy-package-versions`.

### AWS CLI

Para copiar versiones de paquetes de un repositorio a otro

A continuación, `copy-package-versions` se mueven las versiones 4.0.0 y 5.0.0 de un paquete denominado `test-package` de `my-repo` a `test-repo`.

```
aws codeartifact copy-package-versions \  
  --domain test-domain \  
  --source-repository my-repo \  
  --destination-repository test-repo \  
  --format npm \  
  --package test-package \  
  --versions '["4.0.0", "5.0.0"]'
```

Salida:

```
{  
  "format": "npm",  
  "package": "test-package",  
  "versions": [  
    {  
      "version": "5.0.0",  
      "revision": "REVISION-1-SAMPLE-6C81EFF7DA55CC",  
      "status": "Published"  
    },  
    {  
      "version": "4.0.0",  
      "revision": "REVISION-2-SAMPLE-55C752BEE772FC",
```

```

        "status": "Published"
      }
    ]
  }

```

[Para obtener más información, consulte Copiar paquetes entre repositorios en la Guía del usuario.AWS CodeArtifact](#)

- Para API obtener más información, consulte [CopyPackageVersions](#) la Referencia de AWS CLI comandos.

## create-domain

En el siguiente ejemplo de código, se muestra cómo usar `create-domain`.

### AWS CLI

Para crear un dominio

En el siguiente `create-domain` ejemplo, se crea un dominio denominado `test-domain`.

```

aws codeartifact create-domain \
  --domain test-domain

```

Salida:

```

{
  "domain": {
    "name": "test-domain",
    "owner": "111122223333",
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:domain/test-domain",
    "status": "Active",
    "createdTime": "2020-10-20T13:16:48.559000-04:00",
    "encryptionKey": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "repositoryCount": 0,
    "assetSizeBytes": 0
  }
}

```

Para obtener más información, consulte [Crear un dominio](#) en la Guía del AWS CodeArtifact usuario.



- Para API obtener más información, consulte [CreateDomain](#) la Referencia de AWS CLI comandos.

## create-repository

En el siguiente ejemplo de código, se muestra cómo usar `create-repository`.

### AWS CLI

Para crear un repositorio

El siguiente `create-repository` ejemplo crea un repositorio denominado `test-repo` dentro de un dominio denominado `test-domain`.

```
aws codeartifact create-repository \  
  --domain test-domain \  
  --domain-owner 111122223333 \  
  --repository test-repo \  
  --description "This is a test repository."
```

Salida:

```
{  
  "repository": {  
    "name": "test-repo",  
    "administratorAccount": "111122223333",  
    "domainName": "test-domain",  
    "domainOwner": "111122223333",  
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/  
test-repo",  
    "description": "This is a test repository.",  
    "upstreams": [],  
    "externalConnections": []  
  }  
}
```

Para obtener más información, consulte [Crear un dominio](#) en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [CreateRepository](#) la Referencia de AWS CLI comandos.

## delete-domain-permissions-policy

En el siguiente ejemplo de código, se muestra cómo usar `delete-domain-permissions-policy`.

### AWS CLI

Para eliminar el documento de política de permisos de un dominio

En el siguiente `delete-domain-permissions-policy` ejemplo, se elimina la política de permisos de un dominio denominado `test-domain`.

```
aws codeartifact delete-domain-permissions-policy \  
  --domain test-domain
```

Salida:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "BasicDomainPolicy",  
      "Action": [  
        "codeartifact:GetDomainPermissionsPolicy",  
        "codeartifact:ListRepositoriesInDomain",  
        "codeartifact:GetAuthorizationToken",  
        "codeartifact:CreateRepository"  
      ],  
      "Effect": "Allow",  
      "Resource": "*",  
      "Principal": {  
        "AWS": "arn:aws:iam::111122223333:root"  
      }  
    }  
  ]  
}
```

Para obtener más información, consulte [Eliminar una política de dominio](#) en la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [DeleteDomainPermissionsPolicy](#) la Referencia de AWS CLI comandos.

## delete-domain

En el siguiente ejemplo de código, se muestra cómo usar `delete-domain`.

### AWS CLI

Para eliminar un dominio

En el siguiente `delete-domain` ejemplo, se elimina un dominio denominado `test-domain`.

```
aws codeartifact delete-domain \  
  --domain test-domain
```

Salida:

```
{  
  "domain": {  
    "name": "test-domain",  
    "owner": "417498243647",  
    "arn": "arn:aws:codeartifact:us-west-2:417498243647:domain/test-domain",  
    "status": "Deleted",  
    "createdTime": "2020-10-20T13:16:48.559000-04:00",  
    "encryptionKey": "arn:aws:kms:us-west-2:417498243647:key/c9fe2447-0795-4fda-  
afbe-8464574ae162",  
    "repositoryCount": 0,  
    "assetSizeBytes": 0  
  }  
}
```

Para obtener más información, consulte [Eliminar un dominio](#) en la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [DeleteDomain](#) la Referencia de AWS CLI comandos.

## delete-package-versions

En el siguiente ejemplo de código, se muestra cómo usar `delete-package-versions`.

### AWS CLI

Para eliminar versiones de paquetes

En el siguiente `delete-package-versions` ejemplo, se elimina la versión 4.0.0 de un paquete denominado `test-package`.

```
aws codeartifact delete-package-versions \  
  --domain test-domain \  
  --repo test-repo \  
  --format npm \  
  --package test-package \  
  --versions 4.0.0
```

Salida:

```
{  
  "successfulVersions": {  
    "4.0.0": {  
      "revision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",  
      "status": "Deleted"  
    }  
  },  
  "failedVersions": {}  
}
```

Para obtener más información, consulte [Eliminar una versión de paquete](#) en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [DeletePackageVersions](#)la Referencia de AWS CLI comandos.

## **delete-repository-permissions-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-repository-permissions-policy`.

AWS CLI

Para eliminar una política de permisos de un repositorio

En el siguiente `delete-repository-permissions-policy` ejemplo, se elimina la política de permisos de un repositorio denominado `test-repo`.

```
aws codeartifact delete-repository-permissions-policy \  
  --domain test-domain \  
  --repo test-repo \  
  --policy test-policy
```

```
--domain test-domain \  
--repository test-repo
```

Salida:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Principal": {  
        "AWS": "arn:aws:iam::111122223333:root"  
      },  
      "Action": [  
        "codeartifact:DescribePackageVersion",  
        "codeartifact:DescribeRepository",  
        "codeartifact:GetPackageVersionReadme",  
        "codeartifact:GetRepositoryEndpoint",  
        "codeartifact:ListPackages",  
        "codeartifact:ListPackageVersions",  
        "codeartifact:ListPackageVersionAssets",  
        "codeartifact:ListPackageVersionDependencies",  
        "codeartifact:ReadFromRepository"  
      ],  
      "Resource": "*"   
    }  
  ]  
}
```

Para obtener más información, consulte [Eliminar una política](#) en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [DeleteRepositoryPermissionsPolicy](#) la Referencia de AWS CLI comandos.

## delete-repository

En el siguiente ejemplo de código, se muestra cómo usar delete-repository.

### AWS CLI

Para eliminar un repositorio

En el siguiente `delete-repository` ejemplo, se elimina un repositorio denominado `test-repo` en un dominio denominado `test-domain`.

```
aws codeartifact delete-repository \  
  --domain test-domain \  
  --repository test-repo
```

Salida:

```
{  
  "repository": {  
    "name": "test-repo",  
    "administratorAccount": "111122223333",  
    "domainName": "test-domain",  
    "domainOwner": "111122223333",  
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/  
test-repo",  
    "description": "This is a test repository",  
    "upstreams": [],  
    "externalConnections": []  
  }  
}
```

Para obtener más información, consulte [Eliminar un repositorio](#) en la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [DeleteRepository](#) la Referencia de AWS CLI comandos.

## describe-domain

En el siguiente ejemplo de código, se muestra cómo usar `describe-domain`.

AWS CLI

Para obtener información sobre un dominio

El siguiente `describe-domain` ejemplo devuelve un `DomainDescription` objeto para un dominio denominado `test-domain`.

```
aws codeartifact describe-domain \  

```

```
--domain test-domain
```

Salida:

```
{
  "domain": {
    "name": "test-domain",
    "owner": "111122223333",
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:domain/test-domain",
    "status": "Active",
    "createdTime": "2020-10-20T13:16:48.559000-04:00",
    "encryptionKey": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
    "repositoryCount": 2,
    "assetSizeBytes": 0,
    "s3BucketArn": "arn:aws:s3:::assets-111122223333-us-west-2"
  }
}
```

Para obtener más información, consulte [Descripción general del dominio](#) en la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [DescribeDomain](#) la Referencia de AWS CLI comandos.

## describe-repository

En el siguiente ejemplo de código, se muestra cómo usar `describe-repository`.

AWS CLI

Para obtener información sobre un repositorio

El siguiente `describe-repository` ejemplo devuelve un `RepositoryDescription` objeto para un repositorio denominado `test-repo`.

```
aws codeartifact describe-repository \
  --domain test-domain \
  --repository test-repo
```

Salida:

```
{
  "repository": {
    "name": "test-repo",
    "administratorAccount": "111122223333",
    "domainName": "test-domain",
    "domainOwner": "111122223333",
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/test-repo",
    "description": "This is a test repository.",
    "upstreams": [],
    "externalConnections": []
  }
}
```

Para obtener más información, consulte [Crear un dominio](#) en la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [DescribeRepository](#) la Referencia de AWS CLI comandos.

## disassociate-external-connection

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-external-connection`.

### AWS CLI

Para eliminar una conexión externa de un repositorio

En el siguiente `disassociate-external-connection` ejemplo, se elimina una conexión externa a `npmjs.com` de un repositorio denominado `test-repo`.

```
aws codeartifact disassociate-external-connection \
  --repository test-repo \
  --domain test-domain \
  --external-connection public:npmjs
```

Salida:

```
{
  "repository": {
```



```

    "name": "test-repo",
    "administratorAccount": "111122223333",
    "domainName": "test-domain",
    "domainOwner": "111122223333",
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/
test-repo",
    "upstreams": [],
    "externalConnections": []
  }
}

```

Para obtener más información, consulte [Eliminar una conexión externa](#) en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [DisassociateExternalConnection](#) la Referencia de AWS CLI comandos.

## dispose-package-versions

En el siguiente ejemplo de código, se muestra cómo usar `dispose-package-versions`.

### AWS CLI

Para eliminar los activos de una versión de paquete y establecer su estado en Eliminado

En el siguiente `dispose-package-versions` ejemplo, se eliminan los activos de la versión 4.0.0 del paquete de prueba y se establece su estado en Dispuesto.

```

aws codeartifact dispose-package-versions \
  --domain test-domain \
  --repo test-repo \
  --format npm \
  --package test-package \
  --versions 4.0.0

```

Salida:

```

{
  "successfulVersions": {
    "4.0.0": {
      "revision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",
      "status": "Disposed"
    }
  }
}

```

```
    }  
  },  
  "failedVersions": {}  
}
```

Para obtener más información, consulte [Trabajar con paquetes CodeArtifact en](#) la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [DisposePackageVersions](#)la Referencia de AWS CLI comandos.

## get-authorization-token

En el siguiente ejemplo de código, se muestra cómo usar `get-authorization-token`.

### AWS CLI

Para obtener un token de autorización

El siguiente `get-authorization-token` ejemplo recupera un token de CodeArtifact autorización.

```
aws codeartifact get-authorization-token \  
  --domain test-domain \  
  --query authorizationToken \  
  --output text
```

Salida:

```
This command will return the authorization token. You can store the output in an  
environment variable when calling the command.
```

Para obtener más información, consulte [Configurar pip sin el comando login](#) en la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [GetAuthorizationToken](#)la Referencia de AWS CLI comandos.

## get-domain-permissions-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-domain-permissions-policy`.

## AWS CLI

Para obtener el documento de política de permisos de un dominio

En el siguiente `get-domain-permissions-policy` ejemplo, se obtiene la política de permisos adjunta a un dominio denominado `test-domain`.

```
aws codeartifact get-domain-permissions-policy \  
  --domain test-domain
```

Salida:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "BasicDomainPolicy",  
      "Action": [  
        "codeartifact:GetDomainPermissionsPolicy",  
        "codeartifact:ListRepositoriesInDomain",  
        "codeartifact:GetAuthorizationToken",  
        "codeartifact:CreateRepository"  
      ],  
      "Effect": "Allow",  
      "Resource": "*",  
      "Principal": {  
        "AWS": "arn:aws:iam::111122223333:root"  
      }  
    }  
  ]  
}
```

Para obtener más información, consulte [Leer una política de dominio](#) en la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [GetDomainPermissionsPolicy](#) la Referencia de AWS CLI comandos.

## **get-package-version-asset**

En el siguiente ejemplo de código, se muestra cómo usar `get-package-version-asset`.

## AWS CLI

Para obtener un activo de una versión de paquete

El siguiente `get-package-version-asset` ejemplo recupera el `package.tgz` activo de la versión 4.0.0 de un paquete npm denominado `test-package`.

```
aws codeartifact get-package-version-asset \  
  --domain test-domain \  
  --repository test-repo \  
  --format npm \  
  --package test-package \  
  --package-version 4.0.0 \  
  --asset 'package.tgz' \  
  outfileName
```

Salida:

The output for this command will also store the raw asset in the file provided in place of `outfileName`.

```
{  
  "assetName": "package.tgz",  
  "packageVersion": "4.0.0",  
  "packageVersionRevision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs="  
}
```

Para obtener más información, consulte [Listar los activos de las versiones del paquete](#) en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [GetPackageVersionAsset](#)la Referencia de AWS CLI comandos.

## `get-package-version-readme`

En el siguiente ejemplo de código, se muestra cómo usar `get-package-version-readme`.

## AWS CLI

Para obtener el archivo `readme` de una versión del paquete

El siguiente `get-package-version-readme` ejemplo recupera el archivo `readme` de la versión 4.0.0 de un paquete npm denominado `test-package`.

```
aws codeartifact get-package-version-readme \  
  --domain test-domain \  
  --repo test-repo \  
  --format npm \  
  --package test-package \  
  --package-version 4.0.0
```

Salida:

```
{  
  "format": "npm",  
  "package": "test-package",  
  "version": "4.0.0",  
  "readme": "<div align=\"center\">\n  <a href=\"https://github.com/test-package/  
testpack\"> ... more content ... \n",  
  "versionRevision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs="  
}
```

Para obtener más información, consulte [Ver el archivo readme de la versión del paquete](#) en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [GetPackageVersionReadme](#) la Referencia de AWS CLI comandos.

## get-repository-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `get-repository-endpoint`.

AWS CLI

Para obtener el URL punto final de un repositorio

El siguiente `get-repository-endpoint` ejemplo devuelve el punto final npm del repositorio `test-repo`.

```
aws codeartifact get-repository-endpoint \  
  --domain test-domain \  
  --repository test-repo \  
  --format npm
```

```
--format npm
```

Salida:

```
{
  "repositoryEndpoint": "https://test-domain-111122223333.d.codeartifact.us-
west-2.amazonaws.com/npm/test-repo/"
}
```

Para obtener más información, consulte [Conectarse a un repositorio](#) en la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [GetRepositoryEndpoint](#) la Referencia de AWS CLI comandos.

## get-repository-permissions-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-repository-permissions-policy`.

AWS CLI

Para obtener el documento de política de permisos de un repositorio

En el siguiente `get-repository-permissions-policy` ejemplo, se adjunta la política de permisos a un repositorio denominado `test-repo`.

```
aws codeartifact get-repository-permissions-policy \
  --domain test-domain \
  --repository test-repo
```

Salida:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
      }
    }
  ],
}
```

```
    "Action": [  
      "codeartifact:DescribePackageVersion",  
      "codeartifact:DescribeRepository",  
      "codeartifact:GetPackageVersionReadme",  
      "codeartifact:GetRepositoryEndpoint",  
      "codeartifact:ListPackages",  
      "codeartifact:ListPackageVersions",  
      "codeartifact:ListPackageVersionAssets",  
      "codeartifact:ListPackageVersionDependencies",  
      "codeartifact:ReadFromRepository"  
    ],  
    "Resource": "*"    
  }  
]  
}
```

Para obtener más información, consulte [Leer una política](#) en la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [GetRepositoryPermissionsPolicy](#) la Referencia de AWS CLI comandos.

## list-domains

En el siguiente ejemplo de código, se muestra cómo usar `list-domains`.

### AWS CLI

Para enumerar dominios

En el siguiente `list-domains` ejemplo, se devuelve un resumen de todos los dominios que pertenecen a la AWS cuenta que realiza la llamada.

```
aws codeartifact list-domains
```

Salida:

```
{  
  "domains": [  
    {  
      "name": "my-domain",  
      "owner": "111122223333",
```

```

        "status": "Active",
        "encryptionKey": "arn:aws:kms:us-west-2:111122223333:key/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    },
    {
        "name": "test-domain",
        "owner": "111122223333",
        "status": "Active",
        "encryptionKey": "arn:aws:kms:us-west-2:111122223333:key/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
    }
]
}

```

Para obtener más información, consulte [Trabajar con dominios CodeArtifact en](#) la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [ListDomains](#) la Referencia de AWS CLI comandos.

## list-package-version-assets

En el siguiente ejemplo de código, se muestra cómo usar `list-package-version-assets`.

### AWS CLI

Para ver los activos de una versión de paquete

El siguiente `list-package-version-assets` ejemplo recupera los activos de la versión 4.0.0 de un paquete npm denominado `test-package`.

```

aws codeartifact list-package-version-assets \
  --domain test-domain \
  --repo test-repo \
  --format npm \
  --package test-package \
  --package-version 4.0.0

```

Salida:

```

{
  "format": "npm",
  "package": "test-package",

```



```

"version": "4.0.0",
"versionRevision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",
"assets": [
  {
    "name": "package.tgz",
    "size": 316680,
    "hashes": {
      "MD5": "60078ec6d9e76b89fb55c860832742b2",
      "SHA-1": "b44a9b6297bcb698f1c51a3545a2b3b368d59c52",
      "SHA-256":
"d2aa8c6afc3c8591765785a37d1c5acae482a8eb3ab9729ed28922692454f2e2",
      "SHA-512":
"3e585d15c8a594e20d7de57b362ea81754c011acb2641a19f1b72c8531ea39825896bab344ae616a0a5a824cb9
    }
  }
]
}

```

Para obtener más información, consulte [Listar los activos de las versiones del paquete](#) en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [ListPackageVersionAssets](#)la Referencia de AWS CLI comandos.

## list-package-version-dependencies

En el siguiente ejemplo de código, se muestra cómo usar `list-package-version-dependencies`.

### AWS CLI

Para ver las dependencias de una versión de paquete

El siguiente `list-package-version-dependencies` ejemplo recupera las dependencias de la versión 4.0.0 de un paquete npm denominado `test-package`.

```

aws codeartifact list-package-version-dependencies \
  --domain test-domain \
  --repo test-repo \
  --format npm \
  --package test-package \
  --package-version 4.0.0

```

**Salida:**

```
{
  "format": "npm",
  "package": "test-package",
  "version": "4.0.0",
  "versionRevision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",
  "dependencies": [
    {
      "namespace": "testns",
      "package": "testdep1",
      "dependencyType": "regular",
      "versionRequirement": "1.8.5"
    },
    {
      "namespace": "testns",
      "package": "testdep2",
      "dependencyType": "regular",
      "versionRequirement": "1.8.5"
    }
  ]
}
```

Para obtener más información, consulte [Ver y actualizar los detalles y las dependencias de la versión del paquete en la Guía del usuario](#).AWS CodeArtifact

- Para API obtener más información, consulte [ListPackageVersionDependencies](#) la Referencia de AWS CLI comandos.

**list-package-versions**

En el siguiente ejemplo de código, se muestra cómo usar `list-package-versions`.

**AWS CLI**

Para ver una lista de las versiones de un paquete

El siguiente `list-package-versions` ejemplo devuelve una lista de las versiones de un paquete denominado `kind-of`.

```
aws codeartifact list-package-versions \
  --package kind-of \
  --domain test-domain \
```

```
--repository test-repo \  
--format npm
```

Salida:

```
{  
  "defaultDisplayVersion": "1.0.1",  
  "format": "npm",  
  "package": "kind-of",  
  "versions": [  
    {  
      "version": "1.0.1",  
      "revision": "REVISION-SAMPLE-1-C7F4S5E9B772FC",  
      "status": "Published"  
    },  
    {  
      "version": "1.0.0",  
      "revision": "REVISION-SAMPLE-2-C752BEEF6D2CFC",  
      "status": "Published"  
    },  
    {  
      "version": "0.1.2",  
      "revision": "REVISION-SAMPLE-3-654S65A5C5E1FC",  
      "status": "Published"  
    },  
    {  
      "version": "0.1.1",  
      "revision": "REVISION-SAMPLE-1-C7F4S5E9B772FC",  
      "status": "Published"  
    },  
    {  
      "version": "0.1.0",  
      "revision": "REVISION-SAMPLE-4-AF669139B772FC",  
      "status": "Published"  
    }  
  ]  
}
```

Para obtener más información, consulte [Listar las versiones de los paquetes](#) en la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [ListPackageVersions](#) la Referencia de AWS CLI comandos.

## list-packages

En el siguiente ejemplo de código, se muestra cómo usar `list-packages`.

### AWS CLI

Para enumerar los paquetes de un repositorio

El siguiente `list-packages` ejemplo muestra una lista de los paquetes de un repositorio denominado `test-repo` en un dominio denominado `test-domain`.

```
aws codeartifact list-packages \  
  --domain test-domain \  
  --repository test-repo
```

Salida:

```
{  
  "packages": [  
    {  
      "format": "npm",  
      "package": "lodash"  
    },  
    {  
      "format": "python",  
      "package": "test-package"  
    }  
  ]  
}
```

Para obtener más información, consulte [Listar los nombres de los paquetes](#) en la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [ListPackages](#) la Referencia de AWS CLI comandos.

## list-repositories-in-domain

En el siguiente ejemplo de código, se muestra cómo usar `list-repositories-in-domain`.

### AWS CLI

Para enumerar los repositorios de un dominio

El siguiente `list-repositories-in-domain` ejemplo devuelve un resumen de todos los repositorios del dominio `test-domain`.

```
aws codeartifact list-repositories-in-domain \  
  --domain test-domain
```

Salida:

```
{  
  "repositories": [  
    {  
      "name": "test-repo",  
      "administratorAccount": "111122223333",  
      "domainName": "test-domain",  
      "domainOwner": "111122223333",  
      "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-  
domain/test-repo",  
      "description": "This is a test repository."  
    },  
    {  
      "name": "test-repo2",  
      "administratorAccount": "111122223333",  
      "domainName": "test-domain",  
      "domainOwner": "111122223333",  
      "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-  
domain/test-repo2",  
      "description": "This is a test repository."  
    }  
  ]  
}
```

Para obtener más información, consulte [Listar repositorios](#) en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [ListRepositoriesInDomain](#) la Referencia de AWS CLI comandos.

## **list-repositories**

En el siguiente ejemplo de código, se muestra cómo usar `list-repositories`.

## AWS CLI

Para enumerar los repositorios

El siguiente `list-repositories` ejemplo devuelve un resumen de todos los repositorios del dominio propiedad de la AWS cuenta que realiza la llamada.

```
aws codeartifact list-repositories
```

Salida:

```
{
  "repositories": [
    {
      "name": "npm-store",
      "administratorAccount": "111122223333",
      "domainName": "my-domain",
      "domainOwner": "111122223333",
      "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/my-domain/npm-store",
      "description": "Provides npm artifacts from npm, Inc."
    },
    {
      "name": "target-repo",
      "administratorAccount": "111122223333",
      "domainName": "my-domain",
      "domainOwner": "111122223333",
      "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/my-domain/target-repo",
      "description": "test target repo"
    },
    {
      "name": "test-repo2",
      "administratorAccount": "111122223333",
      "domainName": "test-domain",
      "domainOwner": "111122223333",
      "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/test-repo2",
      "description": "This is a test repository."
    }
  ]
}
```

Para obtener más información, consulta la sección [Listar repositorios](#) en la Guía del AWS CodeArtifact usuario.

- Para API obtener más información, consulte [ListRepositories](#) la Referencia de AWS CLI comandos.

## login

En el siguiente ejemplo de código, se muestra cómo usar login.

### AWS CLI

Para configurar la autenticación en su repositorio con el comando login

El siguiente login ejemplo configura el administrador de paquetes npm con un repositorio denominado test-repo en un dominio denominado test-domain.

```
aws codeartifact login \  
  --domain test-domain \  
  --repository test-repo \  
  --tool npm
```

Salida:

```
Successfully configured npm to use AWS CodeArtifact repository https://test-  
domain-111122223333.d.codeartifact.us-west-2.amazonaws.com/npm/test-repo/  
Login expires in 12 hours at 2020-11-12 01:53:16-05:00
```

Para obtener más información, consulte [Introducción a él en la Guía](#) del usuario. AWS CLI AWS CodeArtifact

- Para API obtener más información, consulte la referencia sobre cómo [iniciar sesión](#) en los AWS CLI comandos.

## put-domain-permissions-policy

En el siguiente ejemplo de código, se muestra cómo usar put-domain-permissions-policy.

### AWS CLI

Para adjuntar una política de permisos a un dominio

En el siguiente `put-domain-permissions-policy` ejemplo, se adjunta una política de permisos definida en el archivo `policy.json` a un dominio denominado `test-domain`.

```
aws codeartifact put-domain-permissions-policy \  
  --domain test-domain \  
  --policy-document file://PATH/T0/policy.json
```

Salida:

```
{  
  "policy": {  
    "resourceArn": "arn:aws:codeartifact:region-id:111122223333:domain/test-  
domain",  
    "document": "{ ...policy document content...}",  
    "revision": "MQ1yyTQRASRU3HB58gBtSDHXG7Q3hvxxxxxxxxx="
```

Para obtener más información, consulte [Establecer una política de dominio](#) en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [PutDomainPermissionsPolicy](#) la Referencia de AWS CLI comandos.

## **put-repository-permissions-policy**

En el siguiente ejemplo de código, se muestra cómo usar `put-repository-permissions-policy`.

AWS CLI

Para adjuntar una política de permisos a un repositorio

En el siguiente `put-repository-permissions-policy` ejemplo, se adjunta una política de permisos definida en el archivo `policy.json` a un repositorio denominado `test-repo`.

```
aws codeartifact put-repository-permissions-policy \  
  --domain test-domain \  
  --repository test-repo \  
  --policy-document file://PATH/T0/policy.json
```



Salida:

```
{
  "policy": {
    "resourceArn": "arn:aws:codeartifact:region-id:111122223333:repository/test-
domain/test-repo",
    "document": "{ ...policy document content...}",
    "revision": "MQ1yyTQRASRU3HB58gBtSDHXG7Q3hvxxxxxxxxx="
  }
}
```

Para obtener más información, consulte [Establecer una](#) política en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [PutRepositoryPermissionsPolicy](#) la Referencia de AWS CLI comandos.

## update-package-versions-status

En el siguiente ejemplo de código, se muestra cómo usar update-package-versions-status.

AWS CLI

Para actualizar el estado de la versión del paquete

En el siguiente update-package-versions-status ejemplo, se actualiza el estado de la versión 4.0.0 del paquete de prueba a Archivado.

```
aws codeartifact update-package-versions-status \
  --domain test-domain \
  --repo test-repo \
  --format npm \
  --package test-package \
  --versions 4.0.0 \
  --target-status Archived
```

Salida:

```
{
  "successfulVersions": {
    "4.0.0": {
      "revision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",
```

```

        "status": "Archived"
      }
    },
    "failedVersions": {}
  }

```

Para obtener más información, consulte [Actualizar el estado de la versión del paquete](#) en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [UpdatePackageVersionsStatus](#)la Referencia de AWS CLI comandos.

## update-repository

En el siguiente ejemplo de código, se muestra cómo usar update-repository.

### AWS CLI

Para actualizar un repositorio

En el siguiente update-repository ejemplo, se actualiza la descripción de un repositorio denominado test-repo en un dominio denominado test-domain a «esta es una descripción actualizada».

```

aws codeartifact update-repository \
  --domain test-domain \
  --repository test-repo \
  --description "this is an updated description"

```

Salida:

```

{
  "repository": {
    "name": "test-repo",
    "administratorAccount": "111122223333",
    "domainName": "test-domain",
    "domainOwner": "111122223333",
    "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/test-repo",
    "description": "this is an updated description",
    "upstreams": [],
    "externalConnections": []
  }
}

```

```
}  
}
```

Para obtener más información, consulta [Ver o modificar la configuración de un repositorio](#) en la Guía del usuario.AWS CodeArtifact

- Para API obtener más información, consulte [UpdateRepository](#) la Referencia de AWS CLI comandos.

## CodeBuild ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with CodeBuild.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **batch-delete-builds**

En el siguiente ejemplo de código, se muestra cómo usar batch-delete-builds.

AWS CLI

Para eliminar compilaciones AWS CodeBuild.

En el siguiente batch-delete-builds ejemplo, se eliminan las compilaciones incluidas CodeBuild con las especificadasIDs.

```
aws codebuild batch-delete-builds --ids my-build-project-one:a1b2c3d4-5678-9012-abcd-11111EXAMPLE my-build-project-two:a1b2c3d4-5678-9012-abcd-22222EXAMPLE
```

Salida:

```
{
  "buildsNotDeleted": [
    {
      "id": "arn:aws:codebuild:us-west-2:123456789012:build/my-build-project-
one:a1b2c3d4-5678-9012-abcd-11111EXAMPLE",
      "statusCode": "BUILD_IN_PROGRESS"
    }
  ],
  "buildsDeleted": [
    "arn:aws:codebuild:us-west-2:123456789012:build/my-build-project-
two:a1b2c3d4-5678-9012-abcd-22222EXAMPLE"
  ]
}
```

Para obtener más información, consulte [Eliminar compilaciones \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [BatchDeleteBuilds](#) la Referencia de AWS CLI comandos.

## batch-get-build-batches

En el siguiente ejemplo de código, se muestra cómo usar batch-get-build-batches.

### AWS CLI

Para ver los detalles de las compilaciones integradas AWS CodeBuild.

En el siguiente batch-get-build-batches ejemplo, se obtiene información sobre los lotes de compilación incluidos en CodeBuild el especificadoIDs.

```
aws codebuild batch-get-build-batches \
  --ids codebuild-demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE
```

Salida:

```
{
  "buildBatches": [
    {
      "id": "codebuild-demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE",
      "arn": "arn:aws:codebuild:us-west-2:123456789012:build-batch/codebuild-
demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE",

```

```
"startTime": "2020-11-03T21:52:20.775000+00:00",
"endTime": "2020-11-03T21:56:59.784000+00:00",
"currentPhase": "SUCCEEDED",
"buildBatchStatus": "SUCCEEDED",
"resolvedSourceVersion": "0a6546f68309560d08a310daac92314c4d378f6b",
"projectName": "codebuild-demo-project",
"phases": [
  {
    "phaseType": "SUBMITTED",
    "phaseStatus": "SUCCEEDED",
    "startTime": "2020-11-03T21:52:20.775000+00:00",
    "endTime": "2020-11-03T21:52:20.976000+00:00",
    "durationInSeconds": 0
  },
  {
    "phaseType": "DOWNLOAD_BATCHSPEC",
    "phaseStatus": "SUCCEEDED",
    "startTime": "2020-11-03T21:52:20.976000+00:00",
    "endTime": "2020-11-03T21:52:57.401000+00:00",
    "durationInSeconds": 36
  },
  {
    "phaseType": "IN_PROGRESS",
    "phaseStatus": "SUCCEEDED",
    "startTime": "2020-11-03T21:52:57.401000+00:00",
    "endTime": "2020-11-03T21:56:59.751000+00:00",
    "durationInSeconds": 242
  },
  {
    "phaseType": "COMBINE_ARTIFACTS",
    "phaseStatus": "SUCCEEDED",
    "startTime": "2020-11-03T21:56:59.751000+00:00",
    "endTime": "2020-11-03T21:56:59.784000+00:00",
    "durationInSeconds": 0
  },
  {
    "phaseType": "SUCCEEDED",
    "startTime": "2020-11-03T21:56:59.784000+00:00"
  }
],
"source": {
  "type": "GITHUB",
  "location": "https://github.com/my-repo/codebuild-demo-project.git",
  "gitCloneDepth": 1,
```

```
    "gitSubmodulesConfig": {
      "fetchSubmodules": false
    },
    "reportBuildStatus": false,
    "insecureSsl": false
  },
  "secondarySources": [],
  "secondarySourceVersions": [],
  "artifacts": {
    "location": ""
  },
  "secondaryArtifacts": [],
  "cache": {
    "type": "NO_CACHE"
  },
  "environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
    "computeType": "BUILD_GENERAL1_SMALL",
    "environmentVariables": [],
    "privilegedMode": false,
    "imagePullCredentialsType": "CODEBUILD"
  },
  "logConfig": {
    "cloudWatchLogs": {
      "status": "ENABLED"
    },
    "s3Logs": {
      "status": "DISABLED",
      "encryptionDisabled": false
    }
  },
  "buildTimeoutInMinutes": 60,
  "queuedTimeoutInMinutes": 480,
  "complete": true,
  "initiator": "Strohm",
  "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
  "buildBatchNumber": 6,
  "buildBatchConfig": {
    "serviceRole": "arn:aws:iam::123456789012:role/service-role/codebuild-demo-project",
    "restrictions": {
      "maximumBuildsAllowed": 100
    }
  },
}
```

```
        "timeoutInMins": 480
    },
    "buildGroups": [
        {
            "identifier": "DOWNLOAD_SOURCE",
            "ignoreFailure": false,
            "currentBuildSummary": {
                "arn": "arn:aws:codebuild:us-west-2:123456789012:build/
codebuild-demo-project:379737d8-bc35-48ec-97fd-776d27545315",
                "requestedOn": "2020-11-03T21:52:21.394000+00:00",
                "buildStatus": "SUCCEEDED",
                "primaryArtifact": {
                    "type": "no_artifacts",
                    "identifier": "DOWNLOAD_SOURCE"
                },
                "secondaryArtifacts": []
            }
        },
        {
            "identifier": "linux_small",
            "dependsOn": [],
            "ignoreFailure": false,
            "currentBuildSummary": {
                "arn": "arn:aws:codebuild:us-west-2:123456789012:build/
codebuild-demo-project:dd785171-ed84-4bb6-8ede-ceeb86e54bdb",
                "requestedOn": "2020-11-03T21:52:57.604000+00:00",
                "buildStatus": "SUCCEEDED",
                "primaryArtifact": {
                    "type": "no_artifacts",
                    "identifier": "linux_small"
                },
                "secondaryArtifacts": []
            }
        },
        {
            "identifier": "linux_medium",
            "dependsOn": [
                "linux_small"
            ],
            "ignoreFailure": false,
            "currentBuildSummary": {
                "arn": "arn:aws:codebuild:us-west-2:123456789012:build/
codebuild-demo-project:97cf7bd4-5313-4786-8243-4aef350a1267",
                "requestedOn": "2020-11-03T21:54:18.474000+00:00",
```

```

        "buildStatus": "SUCCEEDED",
        "primaryArtifact": {
            "type": "no_artifacts",
            "identifier": "linux_medium"
        },
        "secondaryArtifacts": []
    },
    {
        "identifier": "linux_large",
        "dependsOn": [
            "linux_medium"
        ],
        "ignoreFailure": false,
        "currentBuildSummary": {
            "arn": "arn:aws:codebuild:us-west-2:123456789012:build/
codebuild-demo-project:60a194cd-0d03-4337-9db1-d41476a17d27",
            "requestedOn": "2020-11-03T21:55:39.203000+00:00",
            "buildStatus": "SUCCEEDED",
            "primaryArtifact": {
                "type": "no_artifacts",
                "identifier": "linux_large"
            },
            "secondaryArtifacts": []
        }
    }
]
}
],
"buildBatchesNotFound": []
}

```

Para obtener más información, consulte Batch compilaciones en AWS CodeBuild < <https://docs.aws.amazon.com/codebuild/latest/userguide/batch-build.html>> \_\_ en la Guía del usuario.AWS CodeBuild

- Para obtener API más información, [BatchGetBuildBatches](#) consulte AWS CLI la Referencia de comandos.

## batch-get-builds

En el siguiente ejemplo de código, se muestra cómo usar batch-get-builds.



## AWS CLI

Para ver los detalles de las compilaciones integradas AWS CodeBuild.

En el siguiente batch-get-builds ejemplo, se obtiene información sobre las compilaciones CodeBuild incorporadas con lo especificado IDs.

```
aws codebuild batch-get-builds --ids codebuild-demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE codebuild-demo-project:815e755f-bade-4a7e-80f0-efe51EXAMPLE
```

Salida:

```
{
  "buildsNotFound": [],
  "builds": [
    {
      "artifacts": {
        "md5sum": "0e95edf915048a0c22efe6d139fff837",
        "location": "arn:aws:s3:::codepipeline-us-west-2-820783811474/CodeBuild-Python-Pip/BuildArtif/6DJsqQa",
        "encryptionDisabled": false,
        "sha256sum":
          "cfa0df33a090966a737f64ae4fe498969fdc842a0c9aec540bf93c37ac0d05a2"
      },
      "logs": {
        "cloudWatchLogs": {
          "status": "ENABLED"
        },
        "s3Logs": {
          "status": "DISABLED"
        },
        "streamName": "46472baf-8f6b-43c2-9255-b3b963af2732",
        "groupName": "/aws/codebuild/codebuild-demo-project",
        "deepLink": "https://console.aws.amazon.com/cloudwatch/home?region=us-west-2#logEvent:group=/aws/codebuild/codebuild-demo-project;stream=46472baf-8f6b-43c2-9255-b3b963af2732"
      },
      "timeoutInMinutes": 60,
      "environment": {
        "privilegedMode": false,
        "computeType": "BUILD_GENERAL1_MEDIUM",
        "image": "aws/codebuild/windows-base:1.0",
        "environmentVariables": [],

```

```
    "type": "WINDOWS_CONTAINER"
  },
  "projectName": "codebuild-demo-project",
  "buildComplete": true,
  "source": {
    "gitCloneDepth": 1,
    "insecureSsl": false,
    "type": "CODEPIPELINE"
  },
  "buildStatus": "SUCCEEDED",
  "secondaryArtifacts": [],
  "phases": [
    {
      "durationInSeconds": 0,
      "startTime": 1548717462.122,
      "phaseType": "SUBMITTED",
      "endTime": 1548717462.484,
      "phaseStatus": "SUCCEEDED"
    },
    {
      "durationInSeconds": 0,
      "startTime": 1548717462.484,
      "phaseType": "QUEUED",
      "endTime": 1548717462.775,
      "phaseStatus": "SUCCEEDED"
    },
    {
      "durationInSeconds": 34,
      "endTime": 1548717496.909,
      "contexts": [
        {
          "statusCode": "",
          "message": ""
        }
      ],
      "startTime": 1548717462.775,
      "phaseType": "PROVISIONING",
      "phaseStatus": "SUCCEEDED"
    },
    {
      "durationInSeconds": 15,
      "endTime": 1548717512.555,
      "contexts": [
        {
```

```
        "statusCode": "",
        "message": ""
    }
],
"startTime": 1548717496.909,
"phaseType": "DOWNLOAD_SOURCE",
"phaseStatus": "SUCCEEDED"
},
{
    "durationInSeconds": 0,
    "endTime": 1548717512.734,
    "contexts": [
        {
            "statusCode": "",
            "message": ""
        }
    ],
    "startTime": 1548717512.555,
    "phaseType": "INSTALL",
    "phaseStatus": "SUCCEEDED"
},
{
    "durationInSeconds": 0,
    "endTime": 1548717512.924,
    "contexts": [
        {
            "statusCode": "",
            "message": ""
        }
    ],
    "startTime": 1548717512.734,
    "phaseType": "PRE_BUILD",
    "phaseStatus": "SUCCEEDED"
},
{
    "durationInSeconds": 9,
    "endTime": 1548717522.254,
    "contexts": [
        {
            "statusCode": "",
            "message": ""
        }
    ],
    "startTime": 1548717512.924,
```

```
    "phaseType": "BUILD",
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 3,
    "endTime": 1548717525.498,
    "contexts": [
      {
        "statusCode": "",
        "message": ""
      }
    ],
    "startTime": 1548717522.254,
    "phaseType": "POST_BUILD",
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 9,
    "endTime": 1548717534.646,
    "contexts": [
      {
        "statusCode": "",
        "message": ""
      }
    ],
    "startTime": 1548717525.498,
    "phaseType": "UPLOAD_ARTIFACTS",
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 2,
    "endTime": 1548717536.846,
    "contexts": [
      {
        "statusCode": "",
        "message": ""
      }
    ],
    "startTime": 1548717534.646,
    "phaseType": "FINALIZING",
    "phaseStatus": "SUCCEEDED"
  },
  {
    "startTime": 1548717536.846,
```

```

        "phaseType": "COMPLETED"
      }
    ],
    "startTime": 1548717462.122,
    "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
    "initiator": "codepipeline/CodeBuild-Pipeline",
    "secondarySources": [],
    "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-
codebuild-service-role",
    "currentPhase": "COMPLETED",
    "id": "codebuild-demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE",
    "cache": {
      "type": "NO_CACHE"
    },
    "sourceVersion": "arn:aws:s3:::codepipeline-us-west-2-820783811474/
CodeBuild-Python-Pip/SourceArti/1TspnN3.zip",
    "endTime": 1548717536.846,
    "arn": "arn:aws:codebuild:us-west-2:123456789012:build/codebuild-demo-
project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE",
    "queuedTimeoutInMinutes": 480,
    "resolvedSourceVersion": "f2194c1757bbdcb0f8f229254a4b3c8b27d43e0b"
  },
  {
    "artifacts": {
      "md5sum": "",
      "overrideArtifactName": false,
      "location": "arn:aws:s3:::my-artifacts/codebuild-demo-project",
      "encryptionDisabled": false,
      "sha256sum": ""
    },
    "logs": {
      "cloudWatchLogs": {
        "status": "ENABLED"
      },
      "s3Logs": {
        "status": "DISABLED"
      },
      "streamName": "4dea3ca4-20ec-4898-b22a-a9eb9292775d",
      "groupName": "/aws/codebuild/codebuild-demo-project",
      "deepLink": "https://console.aws.amazon.com/cloudwatch/
home?region=us-west-2#logEvent:group=/aws/codebuild/codebuild-demo-
project;stream=4dea3ca4-20ec-4898-b22a-a9eb9292775d"
    },
    "timeoutInMinutes": 60,

```

```
"environment": {
  "privilegedMode": false,
  "computeType": "BUILD_GENERAL1_MEDIUM",
  "image": "aws/codebuild/windows-base:1.0",
  "environmentVariables": [],
  "type": "WINDOWS_CONTAINER"
},
"projectName": "codebuild-demo-project",
"buildComplete": true,
"source": {
  "gitCloneDepth": 1,
  "location": "https://github.com/my-repo/codebuild-demo-project.git",
  "insecureSsl": false,
  "reportBuildStatus": false,
  "type": "GITHUB"
},
"buildStatus": "SUCCEEDED",
"secondaryArtifacts": [],
"phases": [
  {
    "durationInSeconds": 0,
    "startTime": 1548716241.89,
    "phaseType": "SUBMITTED",
    "endTime": 1548716242.241,
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 0,
    "startTime": 1548716242.241,
    "phaseType": "QUEUED",
    "endTime": 1548716242.536,
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 33,
    "endTime": 1548716276.171,
    "contexts": [
      {
        "statusCode": "",
        "message": ""
      }
    ],
    "startTime": 1548716242.536,
    "phaseType": "PROVISIONING",
```

```
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 15,
    "endTime": 1548716291.809,
    "contexts": [
      {
        "statusCode": "",
        "message": ""
      }
    ],
    "startTime": 1548716276.171,
    "phaseType": "DOWNLOAD_SOURCE",
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 0,
    "endTime": 1548716291.993,
    "contexts": [
      {
        "statusCode": "",
        "message": ""
      }
    ],
    "startTime": 1548716291.809,
    "phaseType": "INSTALL",
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 0,
    "endTime": 1548716292.191,
    "contexts": [
      {
        "statusCode": "",
        "message": ""
      }
    ],
    "startTime": 1548716291.993,
    "phaseType": "PRE_BUILD",
    "phaseStatus": "SUCCEEDED"
  },
  {
    "durationInSeconds": 9,
    "endTime": 1548716301.622,
```

```
    "contexts": [  
      {  
        "statusCode": "",  
        "message": ""  
      }  
    ],  
    "startTime": 1548716292.191,  
    "phaseType": "BUILD",  
    "phaseStatus": "SUCCEEDED"  
  },  
  {  
    "durationInSeconds": 3,  
    "endTime": 1548716304.783,  
    "contexts": [  
      {  
        "statusCode": "",  
        "message": ""  
      }  
    ],  
    "startTime": 1548716301.622,  
    "phaseType": "POST_BUILD",  
    "phaseStatus": "SUCCEEDED"  
  },  
  {  
    "durationInSeconds": 8,  
    "endTime": 1548716313.775,  
    "contexts": [  
      {  
        "statusCode": "",  
        "message": ""  
      }  
    ],  
    "startTime": 1548716304.783,  
    "phaseType": "UPLOAD_ARTIFACTS",  
    "phaseStatus": "SUCCEEDED"  
  },  
  {  
    "durationInSeconds": 2,  
    "endTime": 1548716315.935,  
    "contexts": [  
      {  
        "statusCode": "",  
        "message": ""  
      }  
    ]  
  }  
]
```



```

        ],
        "startTime": 1548716313.775,
        "phaseType": "FINALIZING",
        "phaseStatus": "SUCCEEDED"
    },
    {
        "startTime": 1548716315.935,
        "phaseType": "COMPLETED"
    }
],
"startTime": 1548716241.89,
"secondarySourceVersions": [],
"initiator": "my-codebuild-project",
"arn": "arn:aws:codebuild:us-west-2:123456789012:build/codebuild-demo-
project:815e755f-bade-4a7e-80f0-efe51EXAMPLE",
"encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
"serviceRole": "arn:aws:iam::123456789012:role/service-role/my-
codebuild-service-role",
"currentPhase": "COMPLETED",
"id": "codebuild-demo-project:815e755f-bade-4a7e-80f0-efe51EXAMPLE",
"cache": {
    "type": "NO_CACHE"
},
"endTime": 1548716315.935,
"secondarySources": [],
"queuedTimeoutInMinutes": 480,
"resolvedSourceVersion": "f2194c1757bbdcb0f8f229254a4b3c8b27d43e0b"
}
]
}

```

Para obtener más información, consulte [Ver detalles de compilación \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [BatchGetBuilds](#) la Referencia de AWS CLI comandos.

## batch-get-projects

En el siguiente ejemplo de código, se muestra cómo usar batch-get-projects.

## AWS CLI

Para obtener una lista de los nombres de los proyectos de AWS CodeBuild construcción.

En el siguiente `batch-get-projects` ejemplo, se obtiene una lista de los proyectos de CodeBuild construcción especificados por su nombre.

```
aws codebuild batch-get-projects --names codebuild-demo-project codebuild-demo-project2 my-other-demo-project
```

En el siguiente resultado, la `projectsNotFound` matriz muestra los nombres de los proyectos de compilación que se especificaron pero que no se encontraron. La matriz `projects` muestra los detalles de cada proyecto de compilación donde se encontró información.

```
{
  "projectsNotFound": [],
  "projects": [
    {
      "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
      "name": "codebuild-demo-project2",
      "queuedTimeoutInMinutes": 480,
      "timeoutInMinutes": 60,
      "source": {
        "buildspec": "version: 0.2\n\n#env:\n #variables:\n # key:\n\n# key: \"value\"\n # key: \"value\"\n #parameter-store:\n # key: \"value\"\n\n# key: \"value\"\n\n#phases:\n #install:\n #commands:\n # - command\n # - command\n #pre_build:\n #commands:\n # - command\n # - command\n\n build:\n commands:\n # - command\n # - command\n #post_build:\n #commands:\n # - command\n # - command\n#artifacts:\n #files:\n # - location\n # - location\n #name: $(date +%Y-%m-%d)\n #discard-paths: yes\n #base-directory: location\n#cache:\n #paths:\n # - paths",
        "type": "NO_SOURCE",
        "insecureSsl": false,
        "gitCloneDepth": 1
      },
      "artifacts": {
        "type": "NO_ARTIFACTS"
      },
      "badge": {
        "badgeEnabled": false
      },
      "lastModified": 1540588091.108,
      "created": 1540588091.108,
    }
  ]
}
```

```
sample",
  "arn": "arn:aws:codebuild:us-west-2:123456789012:project/test-for-
  "secondarySources": [],
  "secondaryArtifacts": [],
  "cache": {
    "type": "NO_CACHE"
  },
  "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-test-
role",
  "environment": {
    "image": "aws/codebuild/java:openjdk-8",
    "privilegedMode": true,
    "type": "LINUX_CONTAINER",
    "computeType": "BUILD_GENERAL1_SMALL",
    "environmentVariables": []
  },
  "tags": []
},
{
  "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
  "name": "my-other-demo-project",
  "queuedTimeoutInMinutes": 480,
  "timeoutInMinutes": 60,
  "source": {
    "location": "https://github.com/iversonic/codedeploy-sample.git",
    "reportBuildStatus": false,
    "buildspec": "buildspec.yml",
    "insecureSsl": false,
    "gitCloneDepth": 1,
    "type": "GITHUB",
    "auth": {
      "type": "OAUTH"
    }
  },
  "artifacts": {
    "type": "NO_ARTIFACTS"
  },
  "badge": {
    "badgeEnabled": false
  },
  "lastModified": 1523401711.73,
  "created": 1523401711.73,
  "arn": "arn:aws:codebuild:us-west-2:123456789012:project/Project2",
  "cache": {
```

```

        "type": "NO_CACHE"
    },
    "serviceRole": "arn:aws:iam::123456789012:role/service-role/codebuild-
Project2-service-role",
    "environment": {
        "image": "aws/codebuild/nodejs:4.4.7",
        "privilegedMode": false,
        "type": "LINUX_CONTAINER",
        "computeType": "BUILD_GENERAL1_SMALL",
        "environmentVariables": []
    },
    "tags": []
}
]
}

```

Para obtener más información, consulte [Ver los detalles de un proyecto de compilación \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [BatchGetProjects](#) la Referencia de AWS CLI comandos.

## batch-get-report-groups

En el siguiente ejemplo de código, se muestra cómo usar batch-get-report-groups.

### AWS CLI

Para obtener información sobre uno o más grupos de informes en AWS CodeBuild.

En el siguiente batch-get-report-groups ejemplo, se recupera información sobre el grupo de informes con lo especificadoARN.

```

aws codebuild batch-get-report-groups \
  --report-group-arns arn:aws:codebuild:<region-ID>:<user-ID>:report-group/
<report-group-name>

```

Salida:

```

{
  "reportGroups": [
    {

```

```

        "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/<report-
group-name>",
        "name": "report-group-name",
        "type": "TEST",
        "exportConfig": {
            "exportConfigType": "NO_EXPORT"
        },
        "created": "2020-10-01T18:04:08.466000+00:00",
        "lastModified": "2020-10-01T18:04:08.466000+00:00",
        "tags": []
    }
],
"reportGroupsNotFound": []
}

```

Para obtener más información, consulte [Trabajar con grupos de informes](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [BatchGetReportGroups](#) la Referencia de AWS CLI comandos.

## batch-get-reports

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-reports`.

### AWS CLI

Para obtener información sobre uno o más informes en AWS CodeBuild.

En el siguiente `batch-get-reports` ejemplo, se recupera información sobre los informes con lo especificado ARNs.

```

aws codebuild batch-get-reports \
  --report-arns arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-
name>:<report 1 ID> arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-
name>:<report 2 ID>

```

Salida:

```

{
  "reports": [
    {

```

```
    "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-
name>:<report 1 ID>",
    "type": "TEST",
    "name": "<report-group-name>",
    "reportGroupArn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/
<report-group-name>",
    "executionId": "arn:aws:codebuild:<region-ID>:<user-ID>:build/test-
reports:<ID>",
    "status": "FAILED",
    "created": "2020-10-01T11:25:22.531000-07:00",
    "expired": "2020-10-31T11:25:22-07:00",
    "exportConfig": {
      "exportConfigType": "NO_EXPORT"
    },
    "truncated": false,
    "testSummary": {
      "total": 28,
      "statusCounts": {
        "ERROR": 5,
        "FAILED": 1,
        "SKIPPED": 4,
        "SUCCEEDED": 18,
        "UNKNOWN": 0
      },
      "durationInNanoSeconds": 94000000
    }
  },
  {
    "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-
name>:<report 2 ID>",
    "type": "TEST",
    "name": "<report-group-name>",
    "reportGroupArn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/
<report-group-name>",
    "executionId": "arn:aws:codebuild:<region-ID>:<user-ID>:build/test-
reports:<ID>",
    "status": "FAILED",
    "created": "2020-10-01T11:13:05.816000-07:00",
    "expired": "2020-10-31T11:13:05-07:00",
    "exportConfig": {
      "exportConfigType": "NO_EXPORT"
    },
    "truncated": false,
    "testSummary": {
```

```

        "total": 28,
        "statusCounts": {
            "ERROR": 5,
            "FAILED": 1,
            "SKIPPED": 4,
            "SUCCEEDED": 18,
            "UNKNOWN": 0
        },
        "durationInNanoSeconds": 94000000
    }
}
],
"reportsNotFound": []
}

```

Para obtener más información, consulte [Trabajar con informes](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [BatchGetReports](#) la Referencia de AWS CLI comandos.

## create-project

En el siguiente ejemplo de código, se muestra cómo usar `create-project`.

### AWS CLI

Ejemplo 1: Para crear un proyecto de AWS CodeBuild construcción

En el siguiente `create-project` ejemplo, se crea un proyecto de CodeBuild compilación con archivos fuente de un bucket de S3

```

aws codebuild create-project \
  --name "my-demo-project" \
  --source "{\"type\": \"S3\", \"location\": \"codebuild-us-west-2-123456789012-  
input-bucket/my-source.zip\"}" \
  --artifacts {"\"type\": \"S3\", \"location\": \"codebuild-us-west-2-123456789012-  
output-bucket\""} \
  --environment {"\"type\": \"LINUX_CONTAINER\", \"image\": \"aws/codebuild/  
standard:1.0\", \"computeType\": \"BUILD_GENERAL1_SMALL\"}" \
  --service-role "arn:aws:iam::123456789012:role/service-role/my-codebuild-  
service-role"

```

## Salida:

```
{
  "project": {
    "arn": "arn:aws:codebuild:us-west-2:123456789012:project/my-demo-project",
    "name": "my-cli-demo-project",
    "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
    "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-codebuild-
service-role",
    "lastModified": 1556839783.274,
    "badge": {
      "badgeEnabled": false
    },
    "queuedTimeoutInMinutes": 480,
    "environment": {
      "image": "aws/codebuild/standard:1.0",
      "computeType": "BUILD_GENERAL1_SMALL",
      "type": "LINUX_CONTAINER",
      "imagePullCredentialsType": "CODEBUILD",
      "privilegedMode": false,
      "environmentVariables": []
    },
    "artifacts": {
      "location": "codebuild-us-west-2-123456789012-output-bucket",
      "name": "my-cli-demo-project",
      "namespaceType": "NONE",
      "type": "S3",
      "packaging": "NONE",
      "encryptionDisabled": false
    },
    "source": {
      "type": "S3",
      "location": "codebuild-us-west-2-123456789012-input-bucket/my-
source.zip",
      "insecureSsl": false
    },
    "timeoutInMinutes": 60,
    "cache": {
      "type": "NO_CACHE"
    },
    "created": 1556839783.274
  }
}
```



Ejemplo 2: Para crear un proyecto de AWS CodeBuild compilación con un archivo JSON de entrada para los parámetros

En el siguiente `create-project` ejemplo, se CodeBuild crea un proyecto de construcción pasando todos los parámetros necesarios a un archivo JSON de entrada. Cree la plantilla del archivo de entrada ejecutando el comando solo con `--generate-cli-skeleton parameter`.

```
aws codebuild create-project --cli-input-json file://create-project.json
```

El JSON archivo de entrada `create-project.json` contiene el siguiente contenido:

```
{
  "name": "codebuild-demo-project",
  "source": {
    "type": "S3",
    "location": "codebuild-region-ID-account-ID-input-bucket/MessageUtil.zip"
  },
  "artifacts": {
    "type": "S3",
    "location": "codebuild-region-ID-account-ID-output-bucket"
  },
  "environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/standard:1.0",
    "computeType": "BUILD_GENERAL1_SMALL"
  },
  "serviceRole": "serviceIAMRole"
}
```

Salida:

```
{
  "project": {
    "name": "codebuild-demo-project",
    "serviceRole": "serviceIAMRole",
    "tags": [],
    "artifacts": {
      "packaging": "NONE",
      "type": "S3",
      "location": "codebuild-region-ID-account-ID-output-bucket",
      "name": "message-util.zip"
    },
  },
}
```

```

    "lastModified": 1472661575.244,
    "timeoutInMinutes": 60,
    "created": 1472661575.244,
    "environment": {
      "computeType": "BUILD_GENERAL1_SMALL",
      "image": "aws/codebuild/standard:1.0",
      "type": "LINUX_CONTAINER",
      "environmentVariables": []
    },
    "source": {
      "type": "S3",
      "location": "codebuild-region-ID-account-ID-input-bucket/
MessageUtil.zip"
    },
    "encryptionKey": "arn:aws:kms:region-ID:account-ID:alias/aws/s3",
    "arn": "arn:aws:codebuild:region-ID:account-ID:project/codebuild-demo-
project"
  }
}

```

Para obtener más información, consulte [Crear un proyecto de compilación \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [CreateProject](#) la Referencia de AWS CLI comandos.

## create-report-group

En el siguiente ejemplo de código, se muestra cómo usar `create-report-group`.

### AWS CLI

Para crear un grupo de informes en AWS CodeBuild.

En el siguiente `create-report-group` ejemplo, se crea un nuevo grupo de informes.

```

aws codebuild create-report-group \
  --cli-input-json file://create-report-group-source.json

```

Contenido de `create-report-group-source.json`:

```

{

```

```
"name": "cli-created-report-group",
"type": "TEST",
"exportConfig": {
  "exportConfigType": "S3",
  "s3Destination": {
    "bucket": "my-s3-bucket",
    "path": "",
    "packaging": "ZIP",
    "encryptionDisabled": true
  }
}
```

Salida:

```
{
  "reportGroup": {
    "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/cli-created-report-group",
    "name": "cli-created-report-group",
    "type": "TEST",
    "exportConfig": {
      "exportConfigType": "S3",
      "s3Destination": {
        "bucket": "my-s3-bucket",
        "path": "",
        "packaging": "ZIP",
        "encryptionDisabled": true
      }
    },
    "created": 1602020026.775,
    "lastModified": 1602020026.775
  }
}
```

Para obtener más información, consulte [Trabajar con grupos de informes](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [CreateReportGroup](#) la Referencia de AWS CLI comandos.

## create-webhook

En el siguiente ejemplo de código, se muestra cómo usar `create-webhook`.

### AWS CLI

Para crear filtros de webhook para un proyecto AWS CodeBuild

En el siguiente `create-webhook` ejemplo, se crea un webhook para un CodeBuild proyecto denominado `my-project` que tiene dos grupos de filtros. El primer grupo de filtros especifica las solicitudes de extracción que se crean, actualizan o vuelven a abrirse en ramificaciones con nombres de referencia de Git que coinciden con la expresión regular `^refs/heads/master$` y las referencias de encabezado que coinciden con `^refs/heads/myBranch$`. El segundo grupo de filtros especifica las solicitudes push en las ramas con nombres de referencia de Git que no coinciden con la expresión regular `^refs/heads/myBranch$`.

```
aws codebuild create-webhook \
  --project-name my-project \
  --filter-groups "[[{"type":"EVENT","pattern":"PULL_REQUEST_CREATED,
  PULL_REQUEST_UPDATED, PULL_REQUEST_REOPENED"}, {"type":"HEAD_REF","pattern
  \":"^refs/heads/myBranch$","excludeMatchedPattern":true}, {"type":"BASE_REF
  \","pattern":"^refs/heads/master$","excludeMatchedPattern":true}], [{"type":"
  EVENT","pattern":"PUSH"}, {"type":"HEAD_REF","pattern":"^refs/heads/
  myBranch$","excludeMatchedPattern":true}]]"
```

Salida:

```
{
  "webhook": {
    "payloadUrl": "https://codebuild.us-west-2.amazonaws.com/webhooks?
t=eyJlbnNyeXB0ZWREYXRhIjoiVVl5MGtoeGRwSzZFRXl2Wnh4bld1Z0tKZ291TVpQNEtFamQ3RD1DYWpRaGIreVFrdm
    "url": "https://api.github.com/repos/iversonic/codedeploy-sample/
hooks/105190656",
    "lastModifiedSecret": 1556311319.069,
    "filterGroups": [
      [
        {
          "type": "EVENT",
          "pattern": "PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED,
PULL_REQUEST_REOPENED",
          "excludeMatchedPattern": false
        }
      ]
    ]
  }
}
```

```
    {
      "type": "HEAD_REF",
      "pattern": "refs/heads/myBranch$",
      "excludeMatchedPattern": true
    },
    {
      "type": "BASE_REF",
      "pattern": "refs/heads/master$",
      "excludeMatchedPattern": true
    }
  ],
  [
    {
      "type": "EVENT",
      "pattern": "PUSH",
      "excludeMatchedPattern": false
    },
    {
      "type": "HEAD_REF",
      "pattern": "refs/heads/myBranch$",
      "excludeMatchedPattern": true
    }
  ]
]
}
```

Para obtener más información, consulte [Filtrar eventos de GitHub Webhook \(SDK\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [CreateWebhook](#) la Referencia de AWS CLI comandos.

## **delete-build-batch**

En el siguiente ejemplo de código, se muestra cómo usar `delete-build-batch`.

### AWS CLI

Para eliminar una versión integrada por lotes AWS CodeBuild.

En el siguiente `delete-build-batch` ejemplo, se elimina la compilación por lotes especificada.

```
aws codebuild delete-build-batch \  
  --id <project-name>:<batch-ID>
```

Salida:

```
{  
  "statusCode": "BATCH_DELETED",  
  "buildsDeleted": [  
    "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-name>:<build-ID>",  
    "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-name>:<build-ID>",  
    "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-name>:<build-ID>",  
    "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-name>:<build-ID>"  
  ],  
  "buildsNotDeleted": []  
}
```

Para obtener más información, consulte [Compilaciones por lotes AWS CodeBuild en la Guía del AWS CodeBuild usuario](#).

- Para API obtener más información, consulte [DeleteBuildBatch](#) la Referencia de AWS CLI comandos.

## delete-project

En el siguiente ejemplo de código, se muestra cómo usar delete-project.

AWS CLI

Para eliminar un proyecto de AWS CodeBuild compilación

En el siguiente delete-project ejemplo, se elimina el proyecto de CodeBuild compilación especificado.

```
aws codebuild delete-project --name my-project
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un proyecto de compilación \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [DeleteProject](#) la Referencia de AWS CLI comandos.

## delete-report-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-report-group`.

### AWS CLI

Para eliminar un informe, agrupe en AWS CodeBuild.

En el siguiente `delete-report-group` ejemplo, se elimina el grupo de informes con lo especificadoARN.

```
aws codebuild delete-report-group \  
  --arn arn:aws:codebuild:<region-ID>:<user-ID>:report-group/<report-group-name>
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con grupos de informes](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [DeleteReportGroup](#) la Referencia de AWS CLI comandos.

## delete-report

En el siguiente ejemplo de código, se muestra cómo usar `delete-report`.

### AWS CLI

Para eliminar un informe en AWS CodeBuild.

En el siguiente `delete-report` ejemplo, se elimina el informe especificado.

```
aws codebuild delete-report \  
  --arn arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-name>:<report-ID>
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con informes](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [DeleteReport](#) la Referencia de AWS CLI comandos.

## **delete-source-credentials**

En el siguiente ejemplo de código, se muestra cómo usar `delete-source-credentials`.

### AWS CLI

Para desconectarse de un proveedor de origen y eliminar sus tokens de acceso.

En el siguiente `delete-source-credentials` ejemplo, se desconecta de un proveedor de origen y se quitan sus tokens. Las credenciales ARN de origen utilizadas para conectarse al proveedor de origen determinan qué credenciales de origen.

```
aws codebuild delete-source-credentials --arn arn-of-your-credentials
```

Salida:

```
{
  "arn": "arn:aws:codebuild:your-region:your-account-id:token/your-server-type"
}
```

Para obtener más información, consulte [Connect Source Providers with Access Tokens \(CLI\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [DeleteSourceCredentials](#) la Referencia de AWS CLI comandos.

## **delete-webhook**

En el siguiente ejemplo de código, se muestra cómo usar `delete-webhook`.

### AWS CLI

Para eliminar un filtro de webhook de un proyecto AWS CodeBuild

En el siguiente `delete-webhook` ejemplo, se elimina un webhook del proyecto especificado. CodeBuild



```
aws codebuild delete-webhook --project-name my-project
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Dejar de ejecutar compilaciones automáticamente \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [DeleteWebhook](#) la Referencia de AWS CLI comandos.

## describe-code-coverages

En el siguiente ejemplo de código, se muestra cómo usar `describe-code-coverages`.

### AWS CLI

Para obtener información detallada sobre los resultados de las pruebas de cobertura de código, consulte AWS CodeBuild.

En el siguiente `describe-code-coverages` ejemplo, se obtiene información sobre los resultados de las pruebas de cobertura de código del informe especificado.

```
aws codebuild describe-code-coverages \  
  --report-arn arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-name>:<report-ID>
```

Salida:

```
{  
  "codeCoverages": [  
    {  
      "id": "20a0adcc-db13-4b66-804b-ecaf9f852855",  
      "reportARN": "arn:aws:codebuild:<region-ID>:972506530580:report/<report-group-name>:<report-ID>",  
      "filePath": "<source-file-1-path>",  
      "lineCoveragePercentage": 83.33,  
      "linesCovered": 5,  
      "linesMissed": 1,  
      "branchCoveragePercentage": 50.0,  
      "branchesCovered": 1,  
      "branchesMissed": 1,  
      "expired": "2020-11-20T21:22:45+00:00"    }  
  ]  
}
```

```

    },
    {
      "id": "0887162d-bf57-4cf1-a164-e432373d1a83",
      "reportARN": "arn:aws:codebuild:<region-ID>:972506530580:report/<report-
group-name>:<report-ID>",
      "filePath": "<source-file-2-path>",
      "lineCoveragePercentage": 90.9,
      "linesCovered": 10,
      "linesMissed": 1,
      "branchCoveragePercentage": 50.0,
      "branchesCovered": 1,
      "branchesMissed": 1,
      "expired": "2020-11-20T21:22:45+00:00"
    }
  ]
}

```

Para obtener más información, consulte los [informes de cobertura de código](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [DescribeCodeCoverages](#) la Referencia de AWS CLI comandos.

## describe-test-cases

En el siguiente ejemplo de código, se muestra cómo usar `describe-test-cases`.

### AWS CLI

Para obtener información detallada sobre los casos de prueba en AWS CodeBuild.

En el siguiente `describe-test-cases` ejemplo, se obtiene información sobre los casos de prueba del informe especificado.

```

aws codebuild describe-test-cases \
  --report-arn arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-
name>:<report-ID>

```

Salida:

```

{
  "testCases": [
    {

```

```

        "reportArn": "arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-
group-name>:<report-ID>",
        "testRawDataPath": "<test-report-path>",
        "prefix": "NUnit.Tests.Assemblies.MockTestFixture",
        "name": "NUnit.Tests.Assemblies.MockTestFixture.NotRunnableTest",
        "status": "ERROR",
        "durationInSeconds": 0,
        "message": "No arguments were provided\n",
        "expired": "2020-11-20T17:52:10+00:00"
    },
    {
        "reportArn": "arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-
group-name>:<report-ID>",
        "testRawDataPath": "<test-report-path>",
        "prefix": "NUnit.Tests.Assemblies.MockTestFixture",
        "name": "NUnit.Tests.Assemblies.MockTestFixture.TestWithException",
        "status": "ERROR",
        "durationInSeconds": 0,
        "message": "System.ApplicationException : Intentional Exception
\nat NUnit.Tests.Assemblies.MockTestFixture.MethodThrowsException()\nat
NUnit.Tests.Assemblies.MockTestFixture.TestWithException()\n\n",
        "expired": "2020-11-20T17:52:10+00:00"
    }
]
}

```

Para obtener más información, consulte [Trabajar con los informes de pruebas AWS CodeBuild en la Guía del AWS CodeBuild usuario](#).

- Para API obtener más información, consulte [DescribeTestCases](#) la Referencia de AWS CLI comandos.

## import-source-credentials

En el siguiente ejemplo de código, se muestra cómo usar `import-source-credentials`.

### AWS CLI

Conecte un AWS CodeBuild usuario a un proveedor de origen importando las credenciales del proveedor de origen.

En el siguiente `import-source-credentials` ejemplo, se importa un token para un repositorio de Bitbucket que usa `BASIC _ AUTH` como tipo de autenticación.

```
aws codebuild import-source-credentials --server-type BITBUCKET --auth-  
type BASIC_AUTH --token my-Bitbucket-password --username my-Bitbucket-username
```

Salida:

```
{  
  "arn": "arn:aws:codebuild:us-west-2:123456789012:token/bitbucket"  
}
```

Para obtener más información, consulte [Connect Source Providers with Access Tokens \(CLI\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [ImportSourceCredentials](#) la Referencia de AWS CLI comandos.

## **invalidate-project-cache**

En el siguiente ejemplo de código, se muestra cómo usar `invalidate-project-cache`.

AWS CLI

Para restablecer la memoria caché de un proyecto de AWS CodeBuild compilación.

En el siguiente `invalidate-project-cache` ejemplo, se restablece la caché del CodeBuild proyecto especificado.

```
aws codebuild invalidate-project-cache --project-name my-project
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Crear almacenamiento en caché CodeBuild en](#) la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [InvalidateProjectCache](#) la Referencia de AWS CLI comandos.

## **list-build-batches-for-project**

En el siguiente ejemplo de código, se muestra cómo usar `list-build-batches-for-project`.

## AWS CLI

Para enumerar las compilaciones por lotes para un proyecto de compilación específico en AWS CodeBuild.

En el siguiente `list-build-batches-for-project` ejemplo, se enumeran las compilaciones por CodeBuild lotes del proyecto especificado.

```
aws codebuild list-build-batches-for-project \
  --project-name "<project-name>"
```

Salida:

```
{
  "ids": [
    "<project-name>:<batch-ID>",
    "<project-name>:<batch-ID>"
  ]
}
```

Para obtener más información, consulte [Compilaciones por lotes AWS CodeBuild en](#) la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [ListBuildBatchesForProject](#) la Referencia de AWS CLI comandos.

## `list-build-batches`

En el siguiente ejemplo de código, se muestra cómo usar `list-build-batches`.

## AWS CLI

Para enumerar las compilaciones por lotes AWS CodeBuild.

En el siguiente `list-build-batches` ejemplo, se enumeran las compilaciones por CodeBuild lotes de la cuenta corriente.

```
aws codebuild list-build-batches
```

Salida:

```
{
```

```
"ids": [  
    "<project-name>:<batch-ID>",  
    "<project-name>:<batch-ID>"  
]  
}
```

Para obtener más información, consulte Batch compilaciones en AWS CodeBuild < <https://docs.aws.amazon.com/codebuild/latest/userguide/batch-build.html>> \_\_ en la Guía del usuario.AWS CodeBuild

- Para obtener API más información, [ListBuildBatches](#)consulte AWS CLI la Referencia de comandos.

## list-builds-for-project

En el siguiente ejemplo de código, se muestra cómo usar `list-builds-for-project`.

### AWS CLI

Para ver una lista de compilaciones de un proyecto de AWS CodeBuild compilación.

En el siguiente `list-builds-for-project` ejemplo, se muestra la compilación IDs en orden descendente para el proyecto de CodeBuild compilación especificado.

```
aws codebuild list-builds-for-project --project-name codebuild-demo-project --sort-order DESCENDING
```

Salida:

```
{  
  "ids": [  
    "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-11111example",  
    "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-22222example",  
    "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-33333example",  
    "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-44444example",  
    "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-55555example"  
  ]  
}
```

Para obtener más información, consulte [Ver una lista de compilaciones IDs para un proyecto de compilación \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario

- Para API obtener más información, consulte [ListBuildsForProject](#) la Referencia de AWS CLI comandos.

## list-builds

En el siguiente ejemplo de código, se muestra cómo usar `list-builds`.

### AWS CLI

Para obtener una lista de AWS CodeBuild compilacionesIDs.

En el siguiente `list-builds` ejemplo, se obtiene una lista de CodeBuild IDs ordenadas en orden ascendente.

```
aws codebuild list-builds --sort-order ASCENDING
```

El resultado incluye un `nextToken` valor que indica que hay más resultados disponibles.

```
{
  "nextToken": "4AEA6u7J...The full token has been omitted for
brevity...MzY20A==",
  "ids": [
    "codebuild-demo-project:815e755f-bade-4a7e-80f0-efe51EXAMPLE"
    "codebuild-demo-project:84a7f3d1-d40e-4956-b4cf-7a9d4EXAMPLE"
    ... The full list of build IDs has been omitted for brevity ...
    "codebuild-demo-project:931d0b72-bf6f-4040-a472-5c707EXAMPLE"
  ]
}
```

Vuelva a ejecutar este comando y proporcione el `nextToken` valor de la respuesta anterior como parámetro para obtener la siguiente parte de la salida. Repita el procedimiento hasta que no reciba ningún `nextToken` valor en la respuesta.

```
aws codebuild list-builds --sort-order ASCENDING --next-
token 4AEA6u7J...The full token has been omitted for brevity...MzY20A==
```

Siguiente parte del resultado:

```
{
  "ids": [
```

```

    "codebuild-demo-project:49015049-21cf-4b50-9708-df115EXAMPLE",
    "codebuild-demo-project:543e7206-68a3-46d6-a4da-759abEXAMPLE",
    ... The full list of build IDs has been omitted for brevity ...
    "codebuild-demo-project:c282f198-4582-4b38-bdc0-26f96EXAMPLE"
  ]
}

```

Para obtener más información, consulte [Ver una lista de compilaciones IDs \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario

- Para API obtener más información, consulte [ListBuilds](#) la Referencia de AWS CLI comandos.

## list-curated-environment-images

En el siguiente ejemplo de código, se muestra cómo usar `list-curated-environment-images`.

AWS CLI

Para obtener una lista de las imágenes de Docker gestionadas por ella AWS CodeBuild , puedes utilizarlas en tus compilaciones.

En el siguiente `list-curated-environment-images` ejemplo, se enumeran las imágenes de Docker gestionadas por las CodeBuild que se pueden utilizar en las compilaciones. :

```
aws codebuild list-curated-environment-images
```

Salida:

```

{
  "platforms": [
    {
      "platform": "AMAZON_LINUX",
      "languages": [
        {
          "language": "JAVA",
          "images": [
            {
              "description": "AWS ElasticBeanstalk - Java 7 Running on
Amazon Linux 64bit v2.1.3",
              "name": "aws/codebuild/eb-java-7-amazonlinux-64:2.1.3",
              "versions": [
                "aws/codebuild/eb-java-7-amazonlinux-64:2.1.3-1.0.0"
              ]
            }
          ]
        }
      ]
    }
  ]
}

```



```

        ]
      },
      {
        "description": "AWS ElasticBeanstalk - Java 8 Running on
Amazon Linux 64bit v2.1.3",
        "name": "aws/codebuild/eb-java-8-amazonlinux-64:2.1.3",
        "versions": [
          "aws/codebuild/eb-java-8-amazonlinux-64:2.1.3-1.0.0"
        ]
      },
      ... LIST TRUNCATED FOR BREVITY ...
    ]
  }
]
}

```

Para obtener más información, consulte [las imágenes de Docker proporcionadas por CodeBuild](#) en la Guía del AWS CodeBuild usuario

- Para API obtener más información, consulte [ListCuratedEnvironmentImages](#) la Referencia de AWS CLI comandos.

## list-projects

En el siguiente ejemplo de código, se muestra cómo usar `list-projects`.

### AWS CLI

Para obtener una lista de los nombres de los proyectos de AWS CodeBuild construcción.

En el siguiente `list-projects` ejemplo, se obtiene una lista de proyectos de CodeBuild construcción ordenados por nombre en orden ascendente.

```
aws codebuild list-projects --sort-by NAME --sort-order ASCENDING
```

El resultado incluye un `nextToken` valor que indica que hay más resultados disponibles.

```
{
  "nextToken": "Ci33ACF6...The full token has been omitted for brevity...U
+AkMx8=",
```

```

    "projects": [
      "codebuild-demo-project",
      "codebuild-demo-project2",
      ... The full list of build project names has been omitted for
brevity ...
      "codebuild-demo-project99"
    ]
  }

```

Vuelva a ejecutar este comando y proporcione el `nextToken` valor de la respuesta anterior como parámetro para obtener la siguiente parte de la salida. Repita el procedimiento hasta que no reciba ningún `nextToken` valor en la respuesta.

```

aws codebuild list-projects --sort-by NAME --sort-order ASCENDING --next-
token Ci33ACF6...The full token has been omitted for brevity...U+AkMx8=

{
  "projects": [
    "codebuild-demo-project100",
    "codebuild-demo-project101",

    ... The full list of build project names has been omitted for brevity ...
    "codebuild-demo-project122"
  ]
}

```

Para obtener más información, consulte [Ver una lista de nombres de proyectos de compilación \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [ListProjects](#) la Referencia de AWS CLI comandos.

## list-report-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-report-groups`.

### AWS CLI

Para obtener una lista del grupo de informes ARNs en el que se encuentra AWS CodeBuild.

En el siguiente `list-report-groups` ejemplo, se recupera el grupo de informes ARNs de la cuenta de la región.

```
aws codebuild list-report-groups
```

Salida:

```
{
  "reportGroups": [
    "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-1",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-2",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-3"
  ]
}
```

Para obtener más información, consulte [Trabajar con grupos de informes](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [ListReportGroups](#) la Referencia de AWS CLI comandos.

## list-reports-for-report-group

En el siguiente ejemplo de código, se muestra cómo usar `list-reports-for-report-group`.

AWS CLI

Para obtener una lista de los informes de un grupo de informes en AWS CodeBuild.

En el siguiente `list-report-for-report-groups` ejemplo, se recuperan los informes del grupo de informes especificado para la cuenta de la región.

```
aws codebuild list-reports-for-report-group \
  --report-group-arn arn:aws:codebuild:<region-ID>:<user-ID>:report-group/<report-  
group-name>
```

Salida:

```
{
  "reports": [
    "arn:aws:codebuild:<region-ID>:<user-ID>:report/report-1",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report/report-2",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report/report-3"
  ]
}
```

```
}
```

Para obtener más información, consulte [Trabajar con grupos de informes](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [ListReportsForReportGroup](#) la Referencia de AWS CLI comandos.

## **list-reports**

En el siguiente ejemplo de código, se muestra cómo usar `list-reports`.

### AWS CLI

Para obtener una lista de los informes de la cuenta corriente de AWS CodeBuild.

En el siguiente `list-reports` ejemplo, se recuperan los informes ARNs de la cuenta corriente.

```
aws codebuild list-reports
```

Salida:

```
{
  "reports": [
    "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-name>:<report ID>",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-name>:<report ID>",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-name>:<report ID>"
  ]
}
```

Para obtener más información, consulte [Trabajar con informes](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [ListReports](#) la Referencia de AWS CLI comandos.

## **list-shared-projects**

En el siguiente ejemplo de código, se muestra cómo usar `list-shared-projects`.

## AWS CLI

Para incluir el proyecto compartido en AWS CodeBuild.

En el siguiente `list-shared-projects` ejemplo, se enumeran los proyectos CodeBuild compartidos que están disponibles para la cuenta actual.

```
aws codebuild list-shared-projects
```

Salida:

```
{
  "projects": [
    "arn:aws:codebuild:<region-ID>:<account-ID>:project/<shared-project-name-1>",
    "arn:aws:codebuild:<region-ID>:<account-ID>:project/<shared-project-name-2>"
  ]
}
```

Para obtener más información, consulte [Trabajar con proyectos compartidos](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [ListSharedProjects](#) la Referencia de AWS CLI comandos.

## `list-shared-report-groups`

En el siguiente ejemplo de código, se muestra cómo usar `list-shared-report-groups`.

## AWS CLI

Para obtener una lista del grupo ARNs de informes compartido AWS CodeBuild.

En el siguiente `list-shared-report-groups` ejemplo, se recupera el grupo de informes ARNs de la cuenta de la región.

```
aws codebuild list-shared-report-groups
```

Salida:

```
{
  "reportGroups": [
```

```
    "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-1",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-2",
    "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-3"
  ]
}
```

Para obtener más información, consulte [Trabajar con grupos de informes](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [ListSharedReportGroups](#) la Referencia de AWS CLI comandos.

## list-source-credentials

En el siguiente ejemplo de código, se muestra cómo usar `list-source-credentials`.

### AWS CLI

Para ver una lista de `sourceCredentialsObjects`

En el siguiente `list-source-credentials` ejemplo, se muestran los tokens de una AWS cuenta conectada a una cuenta de Bitbucket y a una GitHub cuenta. Cada `sourceCredentialsInfos` objeto de la respuesta contiene información sobre las credenciales de la fuente conectada.

```
aws codebuild list-source-credentials
```

Salida:

```
{
  "sourceCredentialsInfos": [
    {
      "serverType": "BITBUCKET",
      "arn": "arn:aws:codebuild:us-west-2:123456789012:token/bitbucket",
      "authType": "BASIC_AUTH"
    },
    {
      "serverType": "GITHUB",
      "arn": "arn:aws:codebuild:us-west-2:123456789012:token/github",
      "authType": "OAUTH"
    }
  ]
}
```

```
}
```

Para obtener más información, consulte [Connect Source Providers with Access Tokens \(CLI\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [ListSourceCredentials](#) la Referencia de AWS CLI comandos.

## retry-build-batch

En el siguiente ejemplo de código, se muestra cómo usar `retry-build-batch`.

### AWS CLI

Para volver a intentar una compilación por lotes fallida. AWS CodeBuild

En el siguiente `retry-build-batch` ejemplo, se reinicia la creación por lotes especificada.

```
aws codebuild retry-build-batch \  
  --id <project-name>:<batch-ID>
```

Salida:

```
{  
  "buildBatch": {  
    "id": "<project-name>:<batch-ID>",  
    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build-batch/<project-  
name>:<batch-ID>",  
    "startTime": "2020-10-21T17:26:23.099000+00:00",  
    "currentPhase": "SUBMITTED",  
    "buildBatchStatus": "IN_PROGRESS",  
    "resolvedSourceVersion": "3a9e11cb419e8fff14b03883dc4e64f6155aaa7e",  
    "projectName": "<project-name>",  
    "phases": [  
      {  
        "phaseType": "SUBMITTED",  
        "phaseStatus": "SUCCEEDED",  
        "startTime": "2020-10-21T17:26:23.099000+00:00",  
        "endTime": "2020-10-21T17:26:23.457000+00:00",  
        "durationInSeconds": 0  
      },  
      {  
        "phaseType": "DOWNLOAD_BATCHSPEC",
```

```
    "phaseStatus": "SUCCEEDED",
    "startTime": "2020-10-21T17:26:23.457000+00:00",
    "endTime": "2020-10-21T17:26:54.902000+00:00",
    "durationInSeconds": 31
  },
  {
    "phaseType": "IN_PROGRESS",
    "phaseStatus": "CLIENT_ERROR",
    "startTime": "2020-10-21T17:26:54.902000+00:00",
    "endTime": "2020-10-21T17:28:16.060000+00:00",
    "durationInSeconds": 81
  },
  {
    "phaseType": "FAILED",
    "phaseStatus": "RETRY",
    "startTime": "2020-10-21T17:28:16.060000+00:00",
    "endTime": "2020-10-21T17:29:39.709000+00:00",
    "durationInSeconds": 83
  },
  {
    "phaseType": "SUBMITTED",
    "startTime": "2020-10-21T17:29:39.709000+00:00"
  }
],
"source": {
  "type": "GITHUB",
  "location": "https://github.com/strohm-a/<project-name>-graph.git",
  "gitCloneDepth": 1,
  "gitSubmodulesConfig": {
    "fetchSubmodules": false
  },
  "reportBuildStatus": false,
  "insecureSsl": false
},
"secondarySources": [],
"secondarySourceVersions": [],
"artifacts": {
  "location": ""
},
"secondaryArtifacts": [],
"cache": {
  "type": "NO_CACHE"
},
"environment": {
```



```
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
    "computeType": "BUILD_GENERAL1_SMALL",
    "environmentVariables": [],
    "privilegedMode": false,
    "imagePullCredentialsType": "CODEBUILD"
  },
  "logConfig": {
    "cloudWatchLogs": {
      "status": "ENABLED"
    },
    "s3Logs": {
      "status": "DISABLED",
      "encryptionDisabled": false
    }
  },
  "buildTimeoutInMinutes": 60,
  "queuedTimeoutInMinutes": 480,
  "complete": false,
  "initiator": "<username>",
  "encryptionKey": "arn:aws:kms:<region-ID>:<account-ID>:alias/aws/s3",
  "buildBatchNumber": 4,
  "buildBatchConfig": {
    "serviceRole": "arn:aws:iam::<account-ID>:role/service-role/<project-
name>",
    "restrictions": {
      "maximumBuildsAllowed": 100
    },
    "timeoutInMins": 480
  },
  "buildGroups": [
    {
      "identifier": "DOWNLOAD_SOURCE",
      "ignoreFailure": false,
      "currentBuildSummary": {
        "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
        "requestedOn": "2020-10-21T17:26:23.889000+00:00",
        "buildStatus": "SUCCEEDED",
        "primaryArtifact": {
          "type": "no_artifacts",
          "identifier": "DOWNLOAD_SOURCE"
        },
        "secondaryArtifacts": []
      }
    }
  ]
}
```

```
    }
  },
  {
    "identifier": "linux_small",
    "dependsOn": [],
    "ignoreFailure": false,
    "currentBuildSummary": {
      "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
      "requestedOn": "2020-10-21T17:26:55.115000+00:00",
      "buildStatus": "FAILED",
      "primaryArtifact": {
        "type": "no_artifacts",
        "identifier": "linux_small"
      },
      "secondaryArtifacts": []
    }
  },
  {
    "identifier": "linux_medium",
    "dependsOn": [
      "linux_small"
    ],
    "ignoreFailure": false,
    "currentBuildSummary": {
      "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
      "requestedOn": "2020-10-21T17:26:54.594000+00:00",
      "buildStatus": "STOPPED"
    }
  },
  {
    "identifier": "linux_large",
    "dependsOn": [
      "linux_medium"
    ],
    "ignoreFailure": false,
    "currentBuildSummary": {
      "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
      "requestedOn": "2020-10-21T17:26:54.701000+00:00",
      "buildStatus": "STOPPED"
    }
  }
}
```

```

    ]
  }
}

```

Para obtener más información, consulte [Compilaciones por lotes AWS CodeBuild en la Guía del AWS CodeBuild usuario](#).

- Para API obtener más información, consulte [RetryBuildBatch](#) la Referencia de AWS CLI comandos.

## retry-build

En el siguiente ejemplo de código, se muestra cómo usar `retry-build`.

### AWS CLI

Para volver a intentar una compilación fallida. AWS CodeBuild

En el siguiente `retry-build` ejemplo, se reinicia la compilación especificada.

```

aws codebuild retry-build \
  --id <project-name>:<build-ID>

```

Salida:

```

{
  "build": {
    "id": "<project-name>:<build-ID>",
    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-
name>:<build-ID>",
    "buildNumber": 9,
    "startTime": "2020-10-21T17:51:38.161000+00:00",
    "currentPhase": "QUEUED",
    "buildStatus": "IN_PROGRESS",
    "projectName": "<project-name>",
    "phases": [
      {
        "phaseType": "SUBMITTED",
        "phaseStatus": "SUCCEEDED",
        "startTime": "2020-10-21T17:51:38.161000+00:00",
        "endTime": "2020-10-21T17:51:38.210000+00:00",
        "durationInSeconds": 0
      }
    ],
  },
}

```

```
    {
      "phaseType": "QUEUED",
      "startTime": "2020-10-21T17:51:38.210000+00:00"
    }
  ],
  "source": {
    "type": "GITHUB",
    "location": "<GitHub-repo-URL>",
    "gitCloneDepth": 1,
    "gitSubmodulesConfig": {
      "fetchSubmodules": false
    },
    "reportBuildStatus": false,
    "insecureSsl": false
  },
  "secondarySources": [],
  "secondarySourceVersions": [],
  "artifacts": {
    "location": ""
  },
  "secondaryArtifacts": [],
  "cache": {
    "type": "NO_CACHE"
  },
  "environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
    "computeType": "BUILD_GENERAL1_SMALL",
    "environmentVariables": [],
    "privilegedMode": false,
    "imagePullCredentialsType": "CODEBUILD"
  },
  "serviceRole": "arn:aws:iam::<account-ID>:role/service-role/<service-role-name>",
  "logs": {
    "deepLink": "https://console.aws.amazon.com/cloudwatch/home?region=<region-ID>#logEvent:group=null;stream=null",
    "cloudWatchLogsArn": "arn:aws:logs:<region-ID>:<account-ID>:log-group:null:log-stream:null",
    "cloudWatchLogs": {
      "status": "ENABLED"
    },
    "s3Logs": {
      "status": "DISABLED",
```

```

        "encryptionDisabled": false
      }
    },
    "timeoutInMinutes": 60,
    "queuedTimeoutInMinutes": 480,
    "buildComplete": false,
    "initiator": "<username>",
    "encryptionKey": "arn:aws:kms:<region-ID>:<account-ID>:alias/aws/s3"
  }
}

```

Para obtener más información, consulte [Compilaciones por lotes AWS CodeBuild en](#) la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [RetryBuild](#) la Referencia de AWS CLI comandos.

## start-build-batch

En el siguiente ejemplo de código, se muestra cómo usar `start-build-batch`.

### AWS CLI

Para iniciar una compilación por lotes AWS CodeBuild.

En el siguiente `start-build-batch` ejemplo, se inicia una creación por lotes del proyecto especificado.

```

aws codebuild start-build-batch \
  --project-name <project-name>

```

Salida:

```

{
  "buildBatch": {
    "id": "<project-name>:<batch-ID>",
    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build-batch/<project-name>:<batch-ID>",
    "startTime": "2020-10-21T16:54:24.740000+00:00",
    "currentPhase": "SUBMITTED",
    "buildBatchStatus": "IN_PROGRESS",
    "projectName": "<project-name>",
    "source": {
      "type": "GITHUB",

```

```
    "location": "<GitHub-repo-URL>",
    "gitCloneDepth": 1,
    "gitSubmodulesConfig": {
      "fetchSubmodules": false
    },
    "reportBuildStatus": false,
    "insecureSsl": false
  },
  "secondarySources": [],
  "secondarySourceVersions": [],
  "artifacts": {
    "location": ""
  },
  "secondaryArtifacts": [],
  "cache": {
    "type": "NO_CACHE"
  },
  "environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
    "computeType": "BUILD_GENERAL1_SMALL",
    "environmentVariables": [],
    "privilegedMode": false,
    "imagePullCredentialsType": "CODEBUILD"
  },
  "logConfig": {
    "cloudWatchLogs": {
      "status": "ENABLED"
    },
    "s3Logs": {
      "status": "DISABLED",
      "encryptionDisabled": false
    }
  },
  "buildTimeoutInMinutes": 60,
  "queuedTimeoutInMinutes": 480,
  "complete": false,
  "initiator": "<username>",
  "encryptionKey": "arn:aws:kms:<region-ID>:<account-ID>:alias/aws/s3",
  "buildBatchNumber": 3,
  "buildBatchConfig": {
    "serviceRole": "arn:aws:iam::<account-ID>:role/service-role/<service-
role-name>",
    "restrictions": {
```

```

        "maximumBuildsAllowed": 100
      },
      "timeoutInMins": 480
    }
  }
}

```

Para obtener más información, consulte [Compilaciones por lotes AWS CodeBuild en](#) la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [StartBuildBatch](#) la Referencia de AWS CLI comandos.

## start-build

En el siguiente ejemplo de código, se muestra cómo usar `start-build`.

### AWS CLI

Para empezar a ejecutar una compilación de un proyecto de AWS CodeBuild compilación.

En el siguiente `start-build` ejemplo, se inicia una compilación para el CodeBuild proyecto especificado. La compilación anula tanto la configuración del proyecto para el número de minutos que se permite que la compilación esté en cola antes de que se agote el tiempo de espera como la configuración de los artefactos del proyecto.

```

aws codebuild start-build \
  --project-name "my-demo-project" \
  --queued-timeout-in-minutes-override 5 \
  --artifacts-override {"type": "S3","location": "arn:aws:s3:::artifacts-override","overrideArtifactName": true}

```

Salida:

```

{
  "build": {
    "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-codebuild-service-role",
    "buildStatus": "IN_PROGRESS",
    "buildComplete": false,
    "projectName": "my-demo-project",
    "timeoutInMinutes": 60,
  }
}

```

```
    "source": {
      "insecureSsl": false,
      "type": "S3",
      "location": "codebuild-us-west-2-123456789012-input-bucket/my-
source.zip"
    },
    "queuedTimeoutInMinutes": 5,
    "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
    "currentPhase": "QUEUED",
    "startTime": 1556905683.568,
    "environment": {
      "computeType": "BUILD_GENERAL1_MEDIUM",
      "environmentVariables": [],
      "type": "LINUX_CONTAINER",
      "privilegedMode": false,
      "image": "aws/codebuild/standard:1.0",
      "imagePullCredentialsType": "CODEBUILD"
    },
    "phases": [
      {
        "phaseStatus": "SUCCEEDED",
        "startTime": 1556905683.568,
        "phaseType": "SUBMITTED",
        "durationInSeconds": 0,
        "endTime": 1556905684.524
      },
      {
        "startTime": 1556905684.524,
        "phaseType": "QUEUED"
      }
    ],
    "logs": {
      "deepLink": "https://console.aws.amazon.com/cloudwatch/home?region=us-
west-2#logEvent:group=null;stream=null"
    },
    "artifacts": {
      "encryptionDisabled": false,
      "location": "arn:aws:s3:::artifacts-override/my-demo-project",
      "overrideArtifactName": true
    },
    "cache": {
      "type": "NO_CACHE"
    },
    "id": "my-demo-project::12345678-a1b2-c3d4-e5f6-11111EXAMPLE",
```



```

    "initiator": "my-aws-account-name",
    "arn": "arn:aws:codebuild:us-west-2:123456789012:build/my-demo-
project::12345678-a1b2-c3d4-e5f6-11111EXAMPLE"
  }
}

```

Para obtener más información, consulta [Run a Build \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [StartBuild](#) la Referencia de AWS CLI comandos.

## stop-build-batch

En el siguiente ejemplo de código, se muestra cómo usar stop-build-batch.

### AWS CLI

Para detener una compilación por lotes en AWS CodeBuild curso.

El siguiente stop-build-batch ejemplo detiene la creación por lotes especificada.

```

aws codebuild stop-build-batch \
  --id <project-name>:<batch-ID>

```

Salida:

```

{
  "buildBatch": {
    "id": "<project-name>:<batch-ID>",
    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build-batch/<project-
name>:<batch-ID>",
    "startTime": "2020-10-21T16:54:24.740000+00:00",
    "endTime": "2020-10-21T16:56:05.152000+00:00",
    "currentPhase": "STOPPED",
    "buildBatchStatus": "STOPPED",
    "resolvedSourceVersion": "aef7744ed069c51098e15c360f4102cd2cd1ad64",
    "projectName": "<project-name>",
    "phases": [
      {
        "phaseType": "SUBMITTED",
        "phaseStatus": "SUCCEEDED",
        "startTime": "2020-10-21T16:54:24.740000+00:00",
        "endTime": "2020-10-21T16:54:25.039000+00:00",

```

```
        "durationInSeconds": 0
      },
      {
        "phaseType": "DOWNLOAD_BATCHSPEC",
        "phaseStatus": "SUCCEEDED",
        "startTime": "2020-10-21T16:54:25.039000+00:00",
        "endTime": "2020-10-21T16:54:56.583000+00:00",
        "durationInSeconds": 31
      },
      {
        "phaseType": "IN_PROGRESS",
        "phaseStatus": "STOPPED",
        "startTime": "2020-10-21T16:54:56.583000+00:00",
        "endTime": "2020-10-21T16:56:05.152000+00:00",
        "durationInSeconds": 68
      },
      {
        "phaseType": "STOPPED",
        "startTime": "2020-10-21T16:56:05.152000+00:00"
      }
    ],
    "source": {
      "type": "GITHUB",
      "location": "<GitHub-repo-URL>",
      "gitCloneDepth": 1,
      "gitSubmodulesConfig": {
        "fetchSubmodules": false
      },
      "reportBuildStatus": false,
      "insecureSsl": false
    },
    "secondarySources": [],
    "secondarySourceVersions": [],
    "artifacts": {
      "location": ""
    },
    "secondaryArtifacts": [],
    "cache": {
      "type": "NO_CACHE"
    },
    "environment": {
      "type": "LINUX_CONTAINER",
      "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
      "computeType": "BUILD_GENERAL1_SMALL",
```

```
    "environmentVariables": [],
    "privilegedMode": false,
    "imagePullCredentialsType": "CODEBUILD"
  },
  "logConfig": {
    "cloudWatchLogs": {
      "status": "ENABLED"
    },
    "s3Logs": {
      "status": "DISABLED",
      "encryptionDisabled": false
    }
  },
  "buildTimeoutInMinutes": 60,
  "queuedTimeoutInMinutes": 480,
  "complete": true,
  "initiator": "Strohm",
  "encryptionKey": "arn:aws:kms:<region-ID>:<account-ID>:alias/aws/s3",
  "buildBatchNumber": 3,
  "buildBatchConfig": {
    "serviceRole": "arn:aws:iam::<account-ID>:role/service-role/<project-
name>",
    "restrictions": {
      "maximumBuildsAllowed": 100
    },
    "timeoutInMins": 480
  },
  "buildGroups": [
    {
      "identifier": "DOWNLOAD_SOURCE",
      "ignoreFailure": false,
      "currentBuildSummary": {
        "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
        "requestedOn": "2020-10-21T16:54:25.468000+00:00",
        "buildStatus": "SUCCEEDED",
        "primaryArtifact": {
          "type": "no_artifacts",
          "identifier": "DOWNLOAD_SOURCE"
        },
        "secondaryArtifacts": []
      }
    }
  ],
  {

```

```

        "identifier": "linux_small",
        "dependsOn": [],
        "ignoreFailure": false,
        "currentBuildSummary": {
          "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
          "requestedOn": "2020-10-21T16:54:56.833000+00:00",
          "buildStatus": "IN_PROGRESS"
        }
      },
      {
        "identifier": "linux_medium",
        "dependsOn": [
          "linux_small"
        ],
        "ignoreFailure": false,
        "currentBuildSummary": {
          "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
          "requestedOn": "2020-10-21T16:54:56.211000+00:00",
          "buildStatus": "PENDING"
        }
      },
      {
        "identifier": "linux_large",
        "dependsOn": [
          "linux_medium"
        ],
        "ignoreFailure": false,
        "currentBuildSummary": {
          "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
          "requestedOn": "2020-10-21T16:54:56.330000+00:00",
          "buildStatus": "PENDING"
        }
      }
    ]
  }
}

```

Para obtener más información, consulte [Compilaciones por lotes AWS CodeBuild en](#) la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [StopBuildBatch](#) la Referencia de AWS CLI comandos.

## stop-build

En el siguiente ejemplo de código, se muestra cómo usar stop-build.

### AWS CLI

Para detener la compilación de un proyecto de AWS CodeBuild compilación.

El siguiente stop-build ejemplo detiene la CodeBuild compilación especificada.

```
aws codebuild stop-build --id my-demo-project:12345678-a1b2-c3d4-e5f6-11111EXAMPLE
```

Salida:

```
{
  "build": {
    "startTime": 1556906956.318,
    "initiator": "my-aws-account-name",
    "projectName": "my-demo-project",
    "currentPhase": "COMPLETED",
    "cache": {
      "type": "NO_CACHE"
    },
    "source": {
      "insecureSsl": false,
      "location": "codebuild-us-west-2-123456789012-input-bucket/my-source.zip",
      "type": "S3"
    },
    "id": "my-demo-project:1a2b3c4d-5678-90ab-cdef-11111EXAMPLE",
    "endTime": 1556906974.781,
    "phases": [
      {
        "durationInSeconds": 0,
        "phaseType": "SUBMITTED",
        "endTime": 1556906956.935,
        "phaseStatus": "SUCCEEDED",
        "startTime": 1556906956.318
      },
      {

```

```
    "durationInSeconds": 1,
    "phaseType": "QUEUED",
    "endTime": 1556906958.272,
    "phaseStatus": "SUCCEEDED",
    "startTime": 1556906956.935
  },
  {
    "phaseType": "PROVISIONING",
    "phaseStatus": "SUCCEEDED",
    "durationInSeconds": 14,
    "contexts": [
      {
        "message": "",
        "statusCode": ""
      }
    ],
    "endTime": 1556906972.847,
    "startTime": 1556906958.272
  },
  {
    "phaseType": "DOWNLOAD_SOURCE",
    "phaseStatus": "SUCCEEDED",
    "durationInSeconds": 0,
    "contexts": [
      {
        "message": "",
        "statusCode": ""
      }
    ],
    "endTime": 1556906973.552,
    "startTime": 1556906972.847
  },
  {
    "phaseType": "INSTALL",
    "phaseStatus": "SUCCEEDED",
    "durationInSeconds": 0,
    "contexts": [
      {
        "message": "",
        "statusCode": ""
      }
    ],
    "endTime": 1556906973.75,
    "startTime": 1556906973.552
  }
]
```

```
    },
    {
      "phaseType": "PRE_BUILD",
      "phaseStatus": "SUCCEEDED",
      "durationInSeconds": 0,
      "contexts": [
        {
          "message": "",
          "statusCode": ""
        }
      ],
      "endTime": 1556906973.937,
      "startTime": 1556906973.75
    },
    {
      "durationInSeconds": 0,
      "phaseType": "BUILD",
      "endTime": 1556906974.781,
      "phaseStatus": "STOPPED",
      "startTime": 1556906973.937
    },
    {
      "phaseType": "COMPLETED",
      "startTime": 1556906974.781
    }
  ],
  "artifacts": {
    "location": "arn:aws:s3:::artifacts-override/my-demo-project",
    "encryptionDisabled": false,
    "overrideArtifactName": true
  },
  "buildComplete": true,
  "buildStatus": "STOPPED",
  "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
  "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-codebuild-
service-role",
  "queuedTimeoutInMinutes": 5,
  "timeoutInMinutes": 60,
  "environment": {
    "type": "LINUX_CONTAINER",
    "environmentVariables": [],
    "computeType": "BUILD_GENERAL1_MEDIUM",
    "privilegedMode": false,
    "image": "aws/codebuild/standard:1.0",
```

```

        "imagePullCredentialsType": "CODEBUILD"
    },
    "logs": {
        "streamName": "1a2b3c4d-5678-90ab-cdef-11111EXAMPLE",
        "deepLink": "https://console.aws.amazon.com/cloudwatch/home?region=us-west-2#logEvent:group=/aws/codebuild/my-demo-project;stream=1a2b3c4d-5678-90ab-cdef-11111EXAMPLE",
        "groupName": "/aws/codebuild/my-demo-project"
    },
    "arn": "arn:aws:codebuild:us-west-2:123456789012:build/my-demo-project:1a2b3c4d-5678-90ab-cdef-11111EXAMPLE"
}
}

```

Para obtener más información, consulte [Stop a Build \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [StopBuild](#) la Referencia de AWS CLI comandos.

## update-project

En el siguiente ejemplo de código, se muestra cómo usar `update-project`.

### AWS CLI

Para cambiar la configuración de un proyecto de AWS CodeBuild compilación.

El siguiente `update-project` ejemplo cambia la configuración del proyecto de CodeBuild construcción especificado denominado `my-demo-project`.

```

aws codebuild update-project --name "my-demo-project" \
  --description "This project is updated" \
  --source "{\"type\": \"S3\", \"location\": \"codebuild-us-west-2-123456789012-input-bucket/my-source-2.zip\"}" \
  --artifacts "{\"type\": \"S3\", \"location\": \"codebuild-us-west-2-123456789012-output-bucket-2\"}" \
  --environment "{\"type\": \"LINUX_CONTAINER\", \"image\": \"aws/codebuild/standard:1.0\", \"computeType\": \"BUILD_GENERAL1_MEDIUM\"}" \
  --service-role "arn:aws:iam::123456789012:role/service-role/my-codebuild-service-role"

```

El resultado muestra la configuración actualizada.



```
{
  "project": {
    "arn": "arn:aws:codebuild:us-west-2:123456789012:project/my-demo-project",
    "environment": {
      "privilegedMode": false,
      "environmentVariables": [],
      "type": "LINUX_CONTAINER",
      "image": "aws/codebuild/standard:1.0",
      "computeType": "BUILD_GENERAL1_MEDIUM",
      "imagePullCredentialsType": "CODEBUILD"
    },
    "queuedTimeoutInMinutes": 480,
    "description": "This project is updated",
    "artifacts": {
      "packaging": "NONE",
      "name": "my-demo-project",
      "type": "S3",
      "namespaceType": "NONE",
      "encryptionDisabled": false,
      "location": "codebuild-us-west-2-123456789012-output-bucket-2"
    },
    "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
    "badge": {
      "badgeEnabled": false
    },
    "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-codebuild-
service-role",
    "lastModified": 1556840545.967,
    "tags": [],
    "timeoutInMinutes": 60,
    "created": 1556839783.274,
    "name": "my-demo-project",
    "cache": {
      "type": "NO_CACHE"
    },
    "source": {
      "type": "S3",
      "insecureSsl": false,
      "location": "codebuild-us-west-2-123456789012-input-bucket/my-
source-2.zip"
    }
  }
}
```

Para obtener más información, consulte [Cambiar la configuración de un proyecto de compilación \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario

- Para API obtener más información, consulte [UpdateProject](#) la Referencia de AWS CLI comandos.

## update-report-group

En el siguiente ejemplo de código, se muestra cómo usar `update-report-group`.

### AWS CLI

Para actualizar un grupo de informes en AWS CodeBuild.

En el siguiente `update-report-group` ejemplo, se cambia el tipo de exportación del grupo de informes a «NO\_EXPORT».

```
aws codebuild update-report-group \
  --arn arn:aws:codebuild:<region-ID>:<user-ID>:report-group/cli-created-report-
group \
  --export-config="exportConfigType=NO_EXPORT"
```

Salida:

```
{
  "reportGroup": {
    "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/cli-created-
report-group",
    "name": "cli-created-report-group",
    "type": "TEST",
    "exportConfig": {
      "exportConfigType": "NO_EXPORT"
    },
    "created": 1602020686.009,
    "lastModified": 1602021033.454,
    "tags": []
  }
}
```

Para obtener más información, consulte [Trabajar con grupos de informes](#) en la Guía del AWS CodeBuild usuario.

- Para API obtener más información, consulte [UpdateReportGroup](#) la Referencia de AWS CLI comandos.

## update-webhook

En el siguiente ejemplo de código, se muestra cómo usar `update-webhook`.

### AWS CLI

Para actualizar el webhook de un proyecto AWS CodeBuild

En el siguiente `update-webhook` ejemplo, se actualiza un webhook para el CodeBuild proyecto especificado con dos grupos de filtros. El `--rotate-secret` parámetro especifica que se debe GitHub rotar la clave secreta del proyecto cada vez que un cambio de código desencadena una compilación. El primer grupo de filtros especifica las solicitudes de extracción que se crean, actualizan o vuelven a abrirse en ramificaciones con nombres de referencia de Git que coinciden con la expresión regular `^refs/heads/master$` y las referencias de encabezado que coinciden con `^refs/heads/myBranch$`. El segundo grupo de filtros especifica las solicitudes push en las ramas con nombres de referencia de Git que no coinciden con la expresión regular `^refs/heads/myBranch$`.

```
aws codebuild update-webhook \
  --project-name Project2 \
  --rotate-secret \
  --filter-groups "[[{"type":"EVENT","pattern":"PULL_REQUEST_CREATED,
  PULL_REQUEST_UPDATED, PULL_REQUEST_REOPENED"}, {"type":"HEAD_REF","pattern
  \":"^refs/heads/myBranch$","excludeMatchedPattern":true}, {"type":"BASE_REF
  \","pattern":"^refs/heads/master$","excludeMatchedPattern":true}], [{"type":"
  EVENT","pattern":"PUSH"}, {"type":"HEAD_REF","pattern":"^refs/heads/
  myBranch$","excludeMatchedPattern":true}]"]"
```

Salida:

```
{
  "webhook": {
    "filterGroups": [
      [
        {
          "pattern": "PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED,
          PULL_REQUEST_REOPENED",
          "type": "EVENT"
```

```
    },
    {
      "excludeMatchedPattern": true,
      "pattern": "refs/heads/myBranch$",
      "type": "HEAD_REF"
    },
    {
      "excludeMatchedPattern": true,
      "pattern": "refs/heads/master$",
      "type": "BASE_REF"
    }
  ],
  [
    {
      "pattern": "PUSH",
      "type": "EVENT"
    },
    {
      "excludeMatchedPattern": true,
      "pattern": "refs/heads/myBranch$",
      "type": "HEAD_REF"
    }
  ]
],
"lastModifiedSecret": 1556312220.133
}
}
```

Para obtener más información, consulte [Cambiar la configuración de un proyecto de compilación \(AWS CLI\)](#) en la Guía del AWS CodeBuild usuario

- Para API obtener más información, consulte [UpdateWebhook](#) la Referencia de AWS CLI comandos.

## CodeCommit ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with CodeCommit.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **associate-approval-rule-template-with-repository**

En el siguiente ejemplo de código, se muestra cómo usar `associate-approval-rule-template-with-repository`.

#### AWS CLI

Para asociar una plantilla de reglas de aprobación a un repositorio

El siguiente `associate-approval-rule-template-with-repository` ejemplo asocia la plantilla de regla de aprobación especificada a un repositorio denominado `MyDemoRepo`.

```
aws codecommit associate-approval-rule-template-with-repository \  
  --repository-name MyDemoRepo \  
  --approval-rule-template-name 2-approver-rule-for-main
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Asociar una plantilla de regla de aprobación a un repositorio](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [AssociateApprovalRuleTemplateWithRepository](#) la Referencia de AWS CLI comandos.

### **batch-associate-approval-rule-template-with-repositories**

En el siguiente ejemplo de código, se muestra cómo usar `batch-associate-approval-rule-template-with-repositories`.

#### AWS CLI

Para asociar una plantilla de reglas de aprobación a varios repositorios en una sola operación

El siguiente `batch-associate-approval-rule-template-with-repositories` ejemplo asocia la plantilla de reglas de aprobación especificada con los repositorios denominados `MyDemoRepo` y `MyOtherDemoRepo`.

Nota: Las plantillas de reglas de aprobación son específicas de la AWS región en la que se crean. Solo se pueden asociar a los repositorios de esa AWS región.

```
aws codecommit batch-associate-approval-rule-template-with-repositories \  
  --repository-names MyDemoRepo, MyOtherDemoRepo \  
  --approval-rule-template-name 2-approver-rule-for-main
```

Salida:

```
{  
  "associatedRepositoryNames": [  
    "MyDemoRepo",  
    "MyOtherDemoRepo"  
  ],  
  "errors": []  
}
```

Para obtener más información, consulte [Asociar una plantilla de regla de aprobación a un repositorio](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [BatchAssociateApprovalRuleTemplateWithRepositories](#) la Referencia de AWS CLI comandos.

## **batch-describe-merge-conflicts**

En el siguiente ejemplo de código, se muestra cómo usar `batch-describe-merge-conflicts`.

### AWS CLI

Para obtener información sobre los conflictos de fusión en todos los archivos o en un subconjunto de archivos en una combinación entre dos especificadores de confirmación

En el siguiente `batch-describe-merge-conflicts` ejemplo, se determinan los conflictos de fusión al fusionar una rama de origen nombrada `feature-randomization` con una rama de destino denominada `main` mediante la `THREE_WAY_MERGE` estrategia de un repositorio denominado `MyDemoRepo`.

```
aws codecommit batch-describe-merge-conflicts \
  --source-commit-specifier feature-randomizationfeature \
  --destination-commit-specifier main \
  --merge-option THREE_WAY_MERGE \
  --repository-name MyDemoRepo
```

Salida:

```
{
  "conflicts": [
    {
      "conflictMetadata": {
        "filePath": "readme.md",
        "fileSizes": {
          "source": 139,
          "destination": 230,
          "base": 85
        },
        "fileModes": {
          "source": "NORMAL",
          "destination": "NORMAL",
          "base": "NORMAL"
        },
        "objectTypes": {
          "source": "FILE",
          "destination": "FILE",
          "base": "FILE"
        },
        "numberOfConflicts": 1,
        "isBinaryFile": {
          "source": false,
          "destination": false,
          "base": false
        },
        "contentConflict": true,
        "fileModeConflict": false,
        "objectTypeConflict": false,
        "mergeOperations": {
          "source": "M",
          "destination": "M"
        }
      },
      "mergeHunks": [
```

```

        {
            "isConflict": true,
            "source": {
                "startLine": 0,
                "endLine": 3,
                "hunkContent": "VGhpcyBpEXAMPLE=="
            },
            "destination": {
                "startLine": 0,
                "endLine": 1,
                "hunkContent": "VXNlIHRoEXAMPLE="
            }
        }
    ]
}
],
"errors": [],
"destinationCommitId": "86958e0aEXAMPLE",
"sourceCommitId": "6ccd57fdEXAMPLE",
"baseCommitId": "767b6958EXAMPLE"
}

```

Para obtener más información, consulta la sección [Resolución de conflictos en una solicitud de extracción](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [BatchDescribeMergeConflicts](#) la Referencia de AWS CLI comandos.

## **batch-disassociate-approval-rule-template-from-repositories**

En el siguiente ejemplo de código, se muestra cómo usar `batch-disassociate-approval-rule-template-from-repositories`.

### AWS CLI

Para desasociar una plantilla de reglas de aprobación de varios repositorios en una sola operación

El siguiente `batch-disassociate-approval-rule-template-from-repositories` ejemplo desasocia la plantilla de reglas de aprobación especificada de los repositorios denominados `MyDemoRepo` `MyOtherDemoRepo`



```
aws codecommit batch-disassociate-approval-rule-template-from-repositories \  
  --repository-names MyDemoRepo, MyOtherDemoRepo \  
  --approval-rule-template-name 1-approval-rule-for-all pull requests
```

Salida:

```
{  
  "disassociatedRepositoryNames": [  
    "MyDemoRepo",  
    "MyOtherDemoRepo"  
  ],  
  "errors": []  
}
```

Para obtener más información, consulte [Desasociar una plantilla de regla de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [BatchDisassociateApprovalRuleTemplateFromRepositories](#) la Referencia de AWS CLI comandos.

## batch-get-commits

En el siguiente ejemplo de código, se muestra cómo usar batch-get-commits.

AWS CLI

Para ver información sobre varias confirmaciones

En el siguiente batch-get-commits ejemplo, se muestran detalles sobre las confirmaciones especificadas.

```
aws codecommit batch-get-commits \  
  --repository-name MyDemoRepo \  
  --commit-ids 317f8570EXAMPLE 4c925148EXAMPLE
```

Salida:

```
{  
  "commits": [  
    {
```

```

    "additionalData": "",
    "committer": {
      "date": "1508280564 -0800",
      "name": "Mary Major",
      "email": "mary_major@example.com"
    },
    "author": {
      "date": "1508280564 -0800",
      "name": "Mary Major",
      "email": "mary_major@example.com"
    },
    "commitId": "317f8570EXAMPLE",
    "treeId": "1f330709EXAMPLE",
    "parents": [
      "6e147360EXAMPLE"
    ],
    "message": "Change variable name and add new response element"
  },
  {
    "additionalData": "",
    "committer": {
      "date": "1508280542 -0800",
      "name": "Li Juan",
      "email": "li_juan@example.com"
    },
    "author": {
      "date": "1508280542 -0800",
      "name": "Li Juan",
      "email": "li_juan@example.com"
    },
    "commitId": "4c925148EXAMPLE",
    "treeId": "1f330709EXAMPLE",
    "parents": [
      "317f8570EXAMPLE"
    ],
    "message": "Added new class"
  }
}

```

Para obtener más información, consulte [Ver los detalles de las confirmaciones](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [BatchGetCommits](#) la Referencia de AWS CLI comandos.

## batch-get-repositories

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-repositories`.

### AWS CLI

Para ver detalles sobre varios repositorios

En este ejemplo, se muestran detalles sobre varios AWS CodeCommit repositorios.

```
aws codecommit batch-get-repositories \  
  --repository-names MyDemoRepo MyOtherDemoRepo
```

Salida:

```
{  
  "repositoriesNotFound": [],  
  "repositories": [  
    {  
      "creationDate": 1429203623.625,  
      "defaultBranch": "main",  
      "repositoryName": "MyDemoRepo",  
      "cloneUrlSsh": "ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/  
MyDemoRepo",  
      "lastModifiedDate": 1430783812.0869999,  
      "repositoryDescription": "My demonstration repository",  
      "cloneUrlHttp": "https://codecommit.us-east-2.amazonaws.com/v1/repos/  
MyDemoRepo",  
      "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",  
      "Arn": "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo"  
      "accountId": "111111111111"  
    },  
    {  
      "creationDate": 1429203623.627,  
      "defaultBranch": "main",  
      "repositoryName": "MyOtherDemoRepo",  
      "cloneUrlSsh": "ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/  
MyOtherDemoRepo",  
      "lastModifiedDate": 1430783812.0889999,  
      "repositoryDescription": "My other demonstration repository",  
      "cloneUrlHttp": "https://codecommit.us-east-2.amazonaws.com/v1/repos/  
MyOtherDemoRepo",  
      "repositoryId": "cfc29ac4-b0cb-44dc-9990-f6f51EXAMPLE",  
      "Arn": "arn:aws:codecommit:us-east-2:111111111111:MyOtherDemoRepo"  ]  
}
```

```

        "accountId": "111111111111"
      }
    ],
    "repositoriesNotFound": []
  }

```

- Para API obtener más información, consulte [BatchGetRepositories](#) la Referencia de AWS CLI comandos.

## create-approval-rule-template

En el siguiente ejemplo de código, se muestra cómo usar `create-approval-rule-template`.

### AWS CLI

Para crear una plantilla de reglas de aprobación

En el siguiente `create-approval-rule-template` ejemplo, se crea una plantilla de reglas de aprobación `2-approver-rule-for-main`. The template requires two users who assume the role of `CodeCommitReview` con el nombre de aprobar cualquier solicitud de extracción antes de que se pueda fusionar con la main sucursal.

```

aws codecommit create-approval-rule-template \
  --approval-rule-template-name 2-approver-rule-for-main \
  --approval-rule-template-description "Requires two developers from the team to approve the pull request if the destination branch is main" \
  --approval-rule-template-content "{\"Version\": \"2018-11-08\",
  \"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
  \": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":
  [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"

```

Salida:

```

{
  "approvalRuleTemplate": {
    "approvalRuleTemplateName": "2-approver-rule-for-main",
    "creationDate": 1571356106.936,
    "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",
  \"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
  \": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":
  [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",

```

```
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
    "approvalRuleTemplateDescription": "Requires two developers from the team to
approve the pull request if the destination branch is main",
    "lastModifiedDate": 1571356106.936,
    "ruleContentSha256": "4711b576EXAMPLE"
  }
}
```

Para obtener más información, consulte [Crear una plantilla de regla de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [CreateApprovalRuleTemplate](#) la Referencia de AWS CLI comandos.

## create-branch

En el siguiente ejemplo de código, se muestra cómo usar `create-branch`.

### AWS CLI

Para crear una rama

En este ejemplo, se crea una rama en un AWS CodeCommit repositorio. Este comando devuelve resultados solo si hay errores.

Comando:

```
aws codecommit create-branch --repository-name MyDemoRepo --branch-name MyNewBranch
--commit-id 317f8570EXAMPLE
```

Salida:

```
None.
```

- Para API obtener más información, consulte [CreateBranch](#) la Referencia de AWS CLI comandos.

## create-commit

En el siguiente ejemplo de código, se muestra cómo usar `create-commit`.

## AWS CLI

Para crear una confirmación

En el siguiente `create-commit` ejemplo, se muestra cómo crear una confirmación inicial para un repositorio que agrega un `readme.md` archivo a un repositorio denominado `MyDemoRepo` en la `main` rama.

```
aws codecommit create-commit \  
  --repository-name MyDemoRepo \  
  --branch-name main \  
  --put-files "filePath=readme.md,fileContent='Welcome to our team repository.'"
```

Salida:

```
{  
  "filesAdded": [  
    {  
      "blobId": "5e1c309d-EXAMPLE",  
      "absolutePath": "readme.md",  
      "fileMode": "NORMAL"  
    }  
  ],  
  "commitId": "4df8b524-EXAMPLE",  
  "treeId": "55b57003-EXAMPLE",  
  "filesDeleted": [],  
  "filesUpdated": []  
}
```

Para obtener más información, consulta [Cómo crear una confirmación AWS CodeCommit en la Guía del AWS CodeCommit usuario](#).

- Para API obtener más información, consulte [CreateCommit](#) la Referencia de AWS CLI comandos.

### **create-pull-request-approval-rule**

En el siguiente ejemplo de código, se muestra cómo usar `create-pull-request-approval-rule`.

## AWS CLI

Para crear una regla de aprobación para una solicitud de extracción

El siguiente `create-pull-request-approval-rule` ejemplo crea una regla de aprobación con el nombre `Require two approved approvers` de la solicitud de extracción especificada. La regla especifica que se requieren dos aprobaciones de un grupo de aprobaciones. El grupo incluye a todos los usuarios que acceden CodeCommit asumiendo el rol de miembros CodeCommitReview de la 123456789012 AWS cuenta. También incluye un IAM usuario o un usuario federado cuyo nombre `Nikhil_Jayashankar` proviene de la misma AWS cuenta.

```
aws codecommit create-pull-request-approval-rule \
  --approval-rule-name "Require two approved approvers" \
  --approval-rule-content "{\"Version\": \"2018-11-08\", \"Statements\":
  [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers
  \": [\"CodeCommitApprovers:123456789012:Nikhil_Jayashankar\",
  \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"
```

Salida:

```
{
  "approvalRule": {
    "approvalRuleName": "Require two approved approvers",
    "lastModifiedDate": 1570752871.932,
    "ruleContentSha256": "7c44e6ebEXAMPLE",
    "creationDate": 1570752871.932,
    "approvalRuleId": "aac33506-EXAMPLE",
    "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"Statements\":
    [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers
    \": [\"CodeCommitApprovers:123456789012:Nikhil_Jayashankar\",
    \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major"
  }
}
```

Para obtener más información, consulte [Crear una regla de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [CreatePullRequestApprovalRule](#) la Referencia de AWS CLI comandos.

## create-pull-request

En el siguiente ejemplo de código, se muestra cómo usar `create-pull-request`.

### AWS CLI

Para crear una solicitud de extracción

En el siguiente `create-pull-request` ejemplo, se crea una solicitud de extracción denominada «Analizador de dificultades de pronunciación» con la descripción «Revisa estos cambios antes del martes», que se dirige a la rama fuente «jane-branch» y se fusiona con la rama predeterminada, «main», en un repositorio llamado «». AWS CodeCommit MyDemoRepo

```
aws codecommit create-pull-request \  
  --title "My Pull Request" \  
  --description "Please review these changes by Tuesday" \  
  --client-request-token 123Example \  
  --targets repositoryName=MyDemoRepo,sourceReference=MyNewBranch
```

Salida:

```
{  
  "pullRequest": {  
    "approvalRules": [  
      {  
        "approvalRuleContent": "{\"Version\": \"2018-11-08\",  
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type  
\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":  
[\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",  
        "approvalRuleId": "dd8b17fe-EXAMPLE",  
        "approvalRuleName": "2-approver-rule-for-main",  
        "creationDate": 1571356106.936,  
        "lastModifiedDate": 571356106.936,  
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",  
        "originApprovalRuleTemplate": {  
          "approvalRuleTemplateId": "dd3d22fe-EXAMPLE",  
          "approvalRuleTemplateName": "2-approver-rule-for-main"  
        },  
        "ruleContentSha256": "4711b576EXAMPLE"  
      }  
    ],  
    "authorArn": "arn:aws:iam::111111111111:user/Jane_Doe",  
    "description": "Please review these changes by Tuesday",
```



```

    "title": "Pronunciation difficulty analyzer",
    "pullRequestTargets": [
      {
        "destinationCommit": "5d036259EXAMPLE",
        "destinationReference": "refs/heads/main",
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "317f8570EXAMPLE",
        "sourceReference": "refs/heads/jane-branch",
        "mergeMetadata": {
          "isMerged": false
        }
      }
    ],
    "lastActivityDate": 1508962823.285,
    "pullRequestId": "42",
    "clientRequestToken": "123Example",
    "pullRequestStatus": "OPEN",
    "creationDate": 1508962823.285
  }
}

```

- Para obtener más información, consulta la sección Referencia de comandos. API [CreatePullRequest](#) AWS CLI

## create-repository

En el siguiente ejemplo de código, se muestra cómo usar `create-repository`.

AWS CLI

Para crear un repositorio

En este ejemplo, se crea un repositorio y se asocia a la AWS cuenta del usuario.

Comando:

```
aws codecommit create-repository --repository-name MyDemoRepo --repository-
description "My demonstration repository"
```

Salida:

```
{
```

```
"repositoryMetadata": {
  "repositoryName": "MyDemoRepo",
  "cloneUrlSsh": "ssh://git-codecommit.us-east-1.amazonaws.com/v1/
repos/MyDemoRepo",
  "lastModifiedDate": 1444766838.027,
  "repositoryDescription": "My demonstration repository",
  "cloneUrlHttp": "https://git-codecommit.us-east-1.amazonaws.com/v1/
repos/MyDemoRepo",
  "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
  "Arn": "arn:aws:codecommit:us-
east-1:111111111111EXAMPLE:MyDemoRepo",
  "accountId": "111111111111"
}
}
```

- Para API obtener más información, consulte [CreateRepository](#) la Referencia de AWS CLI comandos.

## create-unreferenced-merge-commit

En el siguiente ejemplo de código, se muestra cómo usar `create-unreferenced-merge-commit`.

### AWS CLI

Para crear una confirmación sin referencia que represente el resultado de la fusión de dos especificadores de confirmación

En el siguiente `create-unreferenced-merge-commit` ejemplo, se crea una confirmación que representa los resultados de una fusión entre una rama de origen denominada `bugfix-1234` y una rama de destino denominada `main` mediante la MERGE estrategia `THREE_WAY_MERGE` en un repositorio denominado. `MyDemoRepo`

```
aws codecommit create-unreferenced-merge-commit \
  --source-commit-specifier bugfix-1234 \
  --destination-commit-specifier main \
  --merge-option THREE_WAY_MERGE \
  --repository-name MyDemoRepo \
  --name "Maria Garcia" \
  --email "maria_garcia@example.com" \
  --commit-message "Testing the results of this merge."
```

Salida:

```
{
  "commitId": "4f178133EXAMPLE",
  "treeId": "389765daEXAMPLE"
}
```

Para obtener más información, consulta la [sección Resolución de conflictos en una solicitud de extracción](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [CreateUnreferencedMergeCommit](#) la Referencia de AWS CLI comandos.

## credential-helper

En el siguiente ejemplo de código, se muestra cómo usar `credential-helper`.

### AWS CLI

Para configurar el asistente de credenciales incluido en el AWS CLI AWS CodeCommit

La `credential-helper` utilidad no está diseñada para ser llamada directamente desde AWS CLI. En su lugar, está diseñado para usarse como un parámetro con el `git config` comando para configurar su computadora local. Permite a Git usar HTTPS una versión firmada criptográficamente de tus credenciales de IAM usuario o rol de EC2 instancia de Amazon siempre que Git necesite autenticarse AWS para interactuar con CodeCommit los repositorios.

```
git config --global credential.helper '!aws codecommit credential-helper $@'
git config --global credential.UseHttpPath true
```

Salida:

```
[credential]
  helper = !aws codecommit credential-helper $@
  UseHttpPath = true
```

Para obtener más información, consulta [Cómo configurar el AWS CodeCommit uso de otros métodos](#) en la Guía del AWS CodeCommit usuario. Revise el contenido detenidamente y, a continuación, siga los procedimientos de uno de los siguientes temas: [Para HTTPS](#)

conexiones en Linux, macOS o Unix o Para HTTPS conexiones en Windows de la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [CredentialHelper](#) la Referencia de AWS CLI comandos.

## **delete-approval-rule-template**

En el siguiente ejemplo de código, se muestra cómo usar `delete-approval-rule-template`.

### AWS CLI

Para eliminar una plantilla de reglas de aprobación

En el siguiente `delete-approval-rule-template` ejemplo, se elimina la plantilla de regla de aprobación especificada.

```
aws codecommit delete-approval-rule-template \
  --approval-rule-template-name 1-approver-for-all-pull-requests
```

Salida:

```
{
  "approvalRuleTemplateId": "41de97b7-EXAMPLE"
}
```

Para obtener más información, consulte [Eliminar una plantilla de regla de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [DeleteApprovalRuleTemplate](#) la Referencia de AWS CLI comandos.

## **delete-branch**

En el siguiente ejemplo de código, se muestra cómo usar `delete-branch`.

### AWS CLI

Para eliminar una rama

En este ejemplo, se muestra cómo eliminar una rama de un AWS CodeCommit repositorio.

Comando:

```
aws codecommit delete-branch --repository-name MyDemoRepo --branch-name MyNewBranch
```

Salida:

```
{
  "branch": {
    "commitId": "317f8570EXAMPLE",
    "branchName": "MyNewBranch"
  }
}
```

- Para API obtener más información, consulte [DeleteBranch](#) la Referencia de AWS CLI comandos.

## delete-comment-content

En el siguiente ejemplo de código, se muestra cómo usar `delete-comment-content`.

AWS CLI

Para eliminar el contenido de un comentario

Solo puede eliminar el contenido de un comentario que ha creado usted mismo. En este ejemplo se muestra cómo eliminar el contenido de un comentario con el identificador generado por el sistema de. `ff30b348EXAMPLEb9aa670f`

```
aws codecommit delete-comment-content \
  --comment-id ff30b348EXAMPLEb9aa670f
```

Salida:

```
{
  "comment": {
    "creationDate": 1508369768.142,
    "deleted": true,
    "lastModifiedDate": 1508369842.278,
    "clientRequestToken": "123Example",
    "commentId": "ff30b348EXAMPLEb9aa670f",
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
  }
}
```

```
    "callerReactions": [],
    "reactionCounts":
      {
        "CLAP" : 1
      }
  }
}
```

- Para API obtener más información, consulte la Referencia [DeleteCommentContent](#) de AWS CLI comandos.

## delete-file

En el siguiente ejemplo de código, se muestra cómo usar `delete-file`.

### AWS CLI

Para eliminar un archivo

En el siguiente `delete-file` ejemplo, se muestra cómo eliminar un archivo nombrado `README.md` de una rama denominada `main` con el identificador de confirmación más reciente `c5709475EXAMPLE` de un repositorio denominado `MyDemoRepo`.

```
aws codecommit delete-file \
  --repository-name MyDemoRepo \
  --branch-name main \
  --file-path README.md \
  --parent-commit-id c5709475EXAMPLE
```

Salida:

```
{
  "blobId": "559b44fEXAMPLE",
  "commitId": "353cf655EXAMPLE",
  "filePath": "README.md",
  "treeId": "6bc824cEXAMPLE"
}
```

Para obtener más información, consulta [Cómo editar o eliminar un archivo AWS CodeCommit en la guía de AWS CodeCommit API](#) referencia.

- Para API obtener más información, consulte [DeleteFile](#) la Referencia de AWS CLI comandos.

## delete-pull-request-approval-rule

En el siguiente ejemplo de código, se muestra cómo usar `delete-pull-request-approval-rule`.

### AWS CLI

Para eliminar una regla de aprobación para una solicitud de extracción

En el siguiente `delete-pull-request-approval-rule` ejemplo, se elimina la regla de aprobación denominada `My Approval Rule` para la solicitud de extracción especificada.

```
aws codecommit delete-pull-request-approval-rule \
  --approval-rule-name "My Approval Rule" \
  --pull-request-id 15
```

Salida:

```
{
  "approvalRuleId": "077d8e8a8-EXAMPLE"
}
```

Para obtener más información, consulte [Editar o eliminar una regla de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [DeletePullRequestApprovalRule](#) la Referencia de AWS CLI comandos.

## delete-repository

En el siguiente ejemplo de código, se muestra cómo usar `delete-repository`.

### AWS CLI

Para eliminar un repositorio

En este ejemplo se muestra cómo eliminar un AWS CodeCommit repositorio.

Comando:

```
aws codecommit delete-repository --repository-name MyDemoRepo
```

Salida:

```
{
  "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE"
}
```

- Para API obtener más información, consulte [DeleteRepository](#) la Referencia de AWS CLI comandos.

## describe-merge-conflicts

En el siguiente ejemplo de código, se muestra cómo usar `describe-merge-conflicts`.

AWS CLI

Para obtener información detallada sobre los conflictos de fusión

El siguiente `describe-merge-conflicts` ejemplo determina los conflictos de fusión de un archivo nombrado `readme.md` en la rama de origen y la rama de destino especificadas mediante la MERGE estrategia `THREE_WAY __`.

```
aws codecommit describe-merge-conflicts \
  --source-commit-specifier feature-randomizationfeature \
  --destination-commit-specifier main \
  --merge-option THREE_WAY_MERGE \
  --file-path readme.md \
  --repository-name MyDemoRepo
```

Salida:

```
{
  "conflictMetadata": {
    "filePath": "readme.md",
    "fileSizes": {
      "source": 139,
      "destination": 230,
      "base": 85
    },
    "fileModes": {
      "source": "NORMAL",
      "destination": "NORMAL",
```



```
    "base": "NORMAL"
  },
  "objectTypes": {
    "source": "FILE",
    "destination": "FILE",
    "base": "FILE"
  },
  "numberOfConflicts": 1,
  "isBinaryFile": {
    "source": false,
    "destination": false,
    "base": false
  },
  "contentConflict": true,
  "fileModeConflict": false,
  "objectTypeConflict": false,
  "mergeOperations": {
    "source": "M",
    "destination": "M"
  }
},
"mergeHunks": [
  {
    "isConflict": true,
    "source": {
      "startLine": 0,
      "endLine": 3,
      "hunkContent": "VGhpcyBpEXAMPLE="
    },
    "destination": {
      "startLine": 0,
      "endLine": 1,
      "hunkContent": "VXNlIHRoEXAMPLE="
    }
  }
],
"destinationCommitId": "86958e0aEXAMPLE",
"sourceCommitId": "6ccd57fdEXAMPLE",
"baseCommitId": "767b69580EXAMPLE"
}
```

Para obtener más información, consulta la [sección Resolución de conflictos en una solicitud de extracción](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [DescribeMergeConflicts](#) la Referencia de AWS CLI comandos.

## describe-pull-request-events

En el siguiente ejemplo de código, se muestra cómo usar `describe-pull-request-events`.

### AWS CLI

Para ver los eventos de una solicitud de cambios

En el siguiente `describe-pull-request-events` ejemplo, se recuperan los eventos de una solicitud de extracción con el identificador «8».

```
aws codecommit describe-pull-request-events --pull-request-id 8
```

Salida:

```
{
  "pullRequestEvents": [
    {
      "pullRequestId": "8",
      "pullRequestEventType": "PULL_REQUEST_CREATED",
      "eventDate": 1510341779.53,
      "actor": "arn:aws:iam::111111111111:user/Zhang_Wei"
    },
    {
      "pullRequestStatusChangedEventMetadata": {
        "pullRequestStatus": "CLOSED"
      },
      "pullRequestId": "8",
      "pullRequestEventType": "PULL_REQUEST_STATUS_CHANGED",
      "eventDate": 1510341930.72,
      "actor": "arn:aws:iam::111111111111:user/Jane_Doe"
    }
  ]
}
```

- Para API obtener más información, consulta la Referencia de [DescribePullRequestEvents](#) comandos AWS CLI .

## disassociate-approval-rule-template-from-repository

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-approval-rule-template-from-repository`.

### AWS CLI

Para desasociar una plantilla de reglas de aprobación de un repositorio

El siguiente `disassociate-approval-rule-template-from-repository` ejemplo desasocia la plantilla de regla de aprobación especificada de un repositorio denominado.

MyDemoRepo

```
aws codecommit disassociate-approval-rule-template-from-repository \  
  --repository-name MyDemoRepo \  
  --approval-rule-template-name 1-approver-rule-for-all-pull-requests
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Desasociar una plantilla de regla de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [DisassociateApprovalRuleTemplateFromRepository](#) la Referencia de AWS CLI comandos.

## evaluate-pull-request-approval-rules

En el siguiente ejemplo de código, se muestra cómo usar `evaluate-pull-request-approval-rules`.

### AWS CLI

Para evaluar si una solicitud de extracción cumple con todas sus reglas de aprobación

El siguiente `evaluate-pull-request-approval-rules` ejemplo evalúa el estado de las reglas de aprobación de la solicitud de extracción especificada. En este ejemplo, no se ha cumplido una regla de aprobación para la solicitud de extracción, por lo que el resultado del comando muestra un `approved` valor de `false`.

```
aws codecommit evaluate-pull-request-approval-rules \  
  --pull-request-id 27 \  
  --approval-rule-template-name 1-approver-rule-for-all-pull-requests
```

```
--revision-id 9f29d167EXAMPLE
```

Salida:

```
{
  "evaluation": {
    "approved": false,
    "approvalRulesNotSatisfied": [
      "Require two approved approvers"
    ],
    "overridden": false,
    "approvalRulesSatisfied": []
  }
}
```

Para obtener más información, consulte [Combinar una solicitud de extracción](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [EvaluatePullRequestApprovalRules](#) la Referencia de AWS CLI comandos.

## get-approval-rule-template

En el siguiente ejemplo de código, se muestra cómo usar `get-approval-rule-template`.

AWS CLI

Para obtener el contenido de una plantilla de reglas de aprobación

En el siguiente `get-approval-rule-template` ejemplo se obtiene el contenido de una plantilla de reglas de aprobación denominada `1-approver-rule-for-all-pull-requests`.

```
aws codecommit get-approval-rule-template \
  --approval-rule-template-name 1-approver-rule-for-all-pull-requests
```

Salida:

```
{
  "approvalRuleTemplate": {
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"Statements\": [
    {\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers\": [
    \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",

```

```

    "ruleContentSha256": "621181bbEXAMPLE",
    "lastModifiedDate": 1571356106.936,
    "creationDate": 1571356106.936,
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan",
    "approvalRuleTemplateId": "a29abb15-EXAMPLE",
    "approvalRuleTemplateDescription": "All pull requests must be approved by
one developer on the team."
  }
}

```

Para obtener más información, consulte [Gestión de plantillas de reglas de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [GetApprovalRuleTemplate](#) la Referencia de AWS CLI comandos.

## get-blob

En el siguiente ejemplo de código, se muestra cómo usar `get-blob`.

### AWS CLI

Para ver información sobre un objeto blob de Git

El siguiente `get-blob` ejemplo recupera información sobre un blob de Git con el identificador «2eb4af3bEXAMPLE» en un repositorio denominado «». AWS CodeCommit MyDemoRepo

```
aws codecommit get-blob --repository-name MyDemoRepo --blob-id 2eb4af3bEXAMPLE
```

Salida:

```
{
  "content": "QSBcaw5hcnkgTGFyToEXAMPLE="
}
```

- Para API obtener más información, consulte la Referencia de comandos. [GetBlob](#) AWS CLI

## get-branch

En el siguiente ejemplo de código, se muestra cómo usar `get-branch`.

## AWS CLI

Para obtener información sobre una sucursal

En este ejemplo se obtiene información sobre una rama de un AWS CodeCommit repositorio.

Comando:

```
aws codecommit get-branch --repository-name MyDemoRepo --branch-name MyNewBranch
```

Salida:

```
{
  "BranchInfo": {
    "commitID": "317f8570EXAMPLE",
    "branchName": "MyNewBranch"
  }
}
```

- Para API obtener más información, consulte [GetBranch](#) la Referencia de AWS CLI comandos.

## get-comment-reactions

En el siguiente ejemplo de código, se muestra cómo usar `get-comment-reactions`.

### AWS CLI

Para ver las reacciones de los emojis a un comentario

En el siguiente `get-comment-reactions` ejemplo, se enumeran todas las reacciones de los emojis a un comentario con el identificador `abcd1234EXAMPLEb5678efgh`. Si la fuente de tu shell admite la visualización de la versión 1.0 de Emoji, en `emoji` la salida del emoji se mostrará.

```
aws codecommit get-comment-reactions \
  --comment-id abcd1234EXAMPLEb5678efgh
```

Salida:

```
{
```

```
"reactionsForComment": {
  [
    {
      "reaction": {
        "emoji": "??",
        "shortCode": "thumbsup",
        "unicode": "U+1F44D"
      },
      "users": [
        "arn:aws:iam::123456789012:user/Li_Juan",
        "arn:aws:iam::123456789012:user/Mary_Major",
        "arn:aws:iam::123456789012:user/Jorge_Souza"
      ]
    },
    {
      "reaction": {
        "emoji": "??",
        "shortCode": "thumbsdown",
        "unicode": "U+1F44E"
      },
      "users": [
        "arn:aws:iam::123456789012:user/Nikhil_Jayashankar"
      ]
    },
    {
      "reaction": {
        "emoji": "??",
        "shortCode": "confused",
        "unicode": "U+1F615"
      },
      "users": [
        "arn:aws:iam::123456789012:user/Saanvi_Sarkar"
      ]
    }
  ]
}
```

Para obtener más información, consulta [la sección Comentar una confirmación AWS CodeCommit en](#) la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [GetCommentReactions](#) la Referencia de AWS CLI comandos.

## get-comment

En el siguiente ejemplo de código, se muestra cómo usar `get-comment`.

### AWS CLI

Para ver los detalles de un comentario

En este ejemplo se muestra cómo ver los detalles de un comentario con el identificador de comentario generado por el sistema de. `ff30b348EXAMPLEb9aa670f`

```
aws codecommit get-comment \  
--comment-id ff30b348EXAMPLEb9aa670f
```

Salida:

```
{  
  "comment": {  
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
    "clientRequestToken": "123Example",  
    "commentId": "ff30b348EXAMPLEb9aa670f",  
    "content": "Whoops - I meant to add this comment to the line, but I don't  
see how to delete it.",  
    "creationDate": 1508369768.142,  
    "deleted": false,  
    "commentId": "",  
    "lastModifiedDate": 1508369842.278,  
    "callerReactions": [],  
    "reactionCounts":  
    {  
      "SMILE" : 6,  
      "THUMBSUP" : 1  
    }  
  }  
}
```

- Para API obtener más información, consulte la Referencia [GetComment](#) de AWS CLI comandos.



## get-comments-for-compared-commit

En el siguiente ejemplo de código, se muestra cómo usar `get-comments-for-compared-commit`.

### AWS CLI

Para ver los comentarios de una confirmación

En este ejemplo, se muestra cómo ver los comentarios realizados en la comparación entre dos confirmaciones en un repositorio denominado `MyDemoRepo`.

```
aws codecommit get-comments-for-compared-commit \
  --repository-name MyDemoRepo \
  --before-commit-ID 6e147360EXAMPLE \
  --after-commit-id 317f8570EXAMPLE
```

Salida:

```
{
  "commentsForComparedCommitData": [
    {
      "afterBlobId": "1f330709EXAMPLE",
      "afterCommitId": "317f8570EXAMPLE",
      "beforeBlobId": "80906a4cEXAMPLE",
      "beforeCommitId": "6e147360EXAMPLE",
      "comments": [
        {
          "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
          "clientRequestToken": "123Example",
          "commentId": "ff30b348EXAMPLEb9aa670f",
          "content": "Whoops - I meant to add this comment to the line,
not the file, but I don't see how to delete it.",
          "creationDate": 1508369768.142,
          "deleted": false,
          "CommentId": "123abc-EXAMPLE",
          "lastModifiedDate": 1508369842.278,
          "callerReactions": [],
          "reactionCounts":
            {
              "SMILE" : 6,
              "THUMBSUP" : 1
            }
        }
      ]
    }
  ]
}
```

```

    },
    {
      "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
      "clientRequestToken": "123Example",
      "commentId": "553b509bEXAMPLE56198325",
      "content": "Can you add a test case for this?",
      "creationDate": 1508369612.240,
      "deleted": false,
      "commentId": "456def-EXAMPLE",
      "lastModifiedDate": 1508369612.240,
      "callerReactions": [],
      "reactionCounts":
        {
          "THUMBSUP" : 2
        }
    }
  ],
  "location": {
    "filePath": "cl_sample.js",
    "filePosition": 1232,
    "relativeFileVersion": "after"
  },
  "repositoryName": "MyDemoRepo"
}
],
"nextToken": "exampleToken"
}

```

- Para API obtener más información, consulte [GetCommentsForComparedCommit](#) la Referencia de AWS CLI comandos.

## get-comments-for-pull-request

En el siguiente ejemplo de código, se muestra cómo usar `get-comments-for-pull-request`.

### AWS CLI

Para ver los comentarios de una solicitud de cambios

En este ejemplo, se muestra cómo ver los comentarios de una solicitud de incorporación de datos en un repositorio denominado `MyDemoRepo`.

```
aws codecommit get-comments-for-pull-request \
```

```

--repository-name MyDemoRepo \
--before-commit-ID 317f8570EXAMPLE \
--after-commit-id 5d036259EXAMPLE

```

Salida:

```

{
  "commentsForPullRequestData": [
    {
      "afterBlobId": "1f330709EXAMPLE",
      "afterCommitId": "5d036259EXAMPLE",
      "beforeBlobId": "80906a4cEXAMPLE",
      "beforeCommitId": "317f8570EXAMPLE",
      "comments": [
        {
          "authorArn": "arn:aws:iam::111111111111:user/Saanvi_Sarkar",
          "clientRequestToken": "",
          "commentId": "abcd1234EXAMPLEb5678efgh",
          "content": "These don't appear to be used anywhere. Can we
remove them?",
          "creationDate": 1508369622.123,
          "deleted": false,
          "lastModifiedDate": 1508369622.123,
          "callerReactions": [],
          "reactionCounts":
            {
              "THUMBSUP" : 6,
              "CONFUSED" : 1
            }
        },
        {
          "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
          "clientRequestToken": "",
          "commentId": "442b498bEXAMPLE5756813",
          "content": "Good catch. I'll remove them.",
          "creationDate": 1508369829.104,
          "deleted": false,
          "lastModifiedDate": 150836912.273,
          "callerReactions": ["THUMBSUP"]
          "reactionCounts":
            {
              "THUMBSUP" : 14
            }
        }
      ]
    }
  ]
}

```

```

    }
  ],
  "location": {
    "filePath": "ahs_count.py",
    "filePosition": 367,
    "relativeFileVersion": "AFTER"
  },
  "repositoryName": "MyDemoRepo",
  "pullRequestId": "42"
}
],
"nextToken": "exampleToken"
}

```

- Para API obtener más información, consulte [GetCommentsForPullRequest](#) la Referencia de AWS CLI comandos.

## get-commit

En el siguiente ejemplo de código, se muestra cómo usar `get-commit`.

### AWS CLI

Para ver información sobre una confirmación en un repositorio

En este ejemplo, se muestran detalles sobre una confirmación con el identificador generado por el sistema de '7e9fd3091thisisanexample1' en un repositorio denominado ". AWS CodeCommit MyDemoRepo

Comando:

```
aws codecommit get-commit --repository-name MyDemoRepo --commit-id 7e9fd3091thisisanexample1thisisanexample1
```

Salida:

```
{
  "commit": {
    "additionalData": "",
    "committer": {
      "date": "1484167798 -0800",

```

```

    "name": "Mary Major",
    "email": "mary_major@example.com"
  },
  "author": {
    "date": "1484167798 -0800",
    "name": "Mary Major",
    "email": "mary_major@example.com"
  },
  "treeId": "347a3408thisisanexampletreeidexample",
  "parents": [
    "7aa87a031thisisanexamplethisisanexample1"
  ],
  "message": "Fix incorrect variable name"
}
}

```

- API Para obtener más [GetCommit](#) información AWS CLI , consulta la Referencia de comandos.

## get-differences

En el siguiente ejemplo de código, se muestra cómo usar `get-differences`.

### AWS CLI

Para obtener información sobre las diferencias de un especificador de confirmación en un repositorio

En este ejemplo, se muestra información de metadatos sobre los cambios entre dos especificadores de confirmación (rama HEAD, etiqueta u otras referencias totalmente cualificadas, como la confirmación IDs) en una carpeta del AWS CodeCommit repositorio cuyo nombre ha cambiado. MyDemoRepo El ejemplo incluye varias opciones que no son obligatorias, como `--before-commit-specifier`, `--before-path` y `--afterpath`, para ilustrar con más detalle cómo puedes usar estas opciones para limitar los resultados. La respuesta incluye los permisos del modo de archivo.

Comando:

```

aws codecommit get-differences --repository-name MyDemoRepo --before-
commit-specifier 955bba12thisisanexamplethisisanexample --after-commit-
specifier 14a95463thisisanexamplethisisanexample --before-path tmp/example-folder --
after-path tmp/renamed-folder

```

Salida:

```
{
  "differences": [
    {
      "afterBlob": {
        "path": "blob.txt",
        "blobId": "2eb4af3b1thisisanexamplethisisanexample1",
        "mode": "100644"
      },
      "changeType": "M",
      "beforeBlob": {
        "path": "blob.txt",
        "blobId": "bf7fcf281thisisanexamplethisisanexample1",
        "mode": "100644"
      }
    }
  ]
}
```

- Para API obtener más información, consulte [GetDifferences](#) la Referencia de AWS CLI comandos.

## get-file

En el siguiente ejemplo de código, se muestra cómo usar `get-file`.

### AWS CLI

Para obtener el contenido codificado en base 64 de un archivo en un repositorio AWS CodeCommit

El siguiente `get-file` ejemplo muestra cómo obtener el contenido codificado en base 64 de un archivo denominado `README.md` desde una rama denominada `main` en un repositorio denominado `MyDemoRepo`

```
aws codecommit get-file \
  --repository-name MyDemoRepo \
  --commit-specifier main \
  --file-path README.md
```

Salida:

```
{
  "blobId": "559b44fEXAMPLE",
  "commitId": "c5709475EXAMPLE",
  "fileContent": "IyBQaHVzEXAMPLE",
  "filePath": "README.md",
  "fileMode": "NORMAL",
  "fileSize": 1563
}
```

Para obtener más información, consulte la [GetFile](#) guía de AWS CodeCommit API referencia.

- Para API obtener más información, consulte [GetFile](#) la Referencia de AWS CLI comandos.

## get-folder

En el siguiente ejemplo de código, se muestra cómo usar `get-folder`.

### AWS CLI

Para obtener el contenido de una carpeta de un AWS CodeCommit repositorio

El siguiente `get-folder` ejemplo muestra cómo obtener el contenido de una carpeta de nivel superior de un repositorio denominado `MyDemoRepo`.

```
aws codecommit get-folder --repository-name MyDemoRepo --folder-path ""
```

Salida:

```
{
  "commitId": "c5709475EXAMPLE",
  "files": [
    {
      "absolutePath": ".gitignore",
      "blobId": "74094e8bEXAMPLE",
      "fileMode": "NORMAL",
      "relativePath": ".gitignore"
    },
    {
      "absolutePath": "Gemfile",
      "blobId": "9ceb72f6EXAMPLE",
      "fileMode": "NORMAL",

```

```

        "relativePath":"Gemfile"
    },
    {
        "absolutePath":"Gemfile.lock",
        "blobId":"795c4a2aEXAMPLE",
        "fileMode":"NORMAL",
        "relativePath":"Gemfile.lock"
    },
    {
        "absolutePath":"LICENSE.txt",
        "blobId":"0c7932c8EXAMPLE",
        "fileMode":"NORMAL",
        "relativePath":"LICENSE.txt"
    },
    {
        "absolutePath":"README.md",
        "blobId":"559b44feEXAMPLE",
        "fileMode":"NORMAL",
        "relativePath":"README.md"
    }
],
"folderPath":"",
"subFolders":[
    {
        "absolutePath":"public",
        "relativePath":"public",
        "treeId":"d5e92ae3aEXAMPLE"
    },
    {
        "absolutePath":"tmp",
        "relativePath":"tmp",
        "treeId":"d564d0bcEXAMPLE"
    }
],
"subModules":[],
"symbolicLinks":[],
"treeId":"7b3c4dadEXAMPLE"
}

```

Para obtener más información, consulte [GetFolder](#) la guía de AWS CodeCommit API referencia.

- Para API obtener más información, consulte [GetFolder](#) la Referencia de AWS CLI comandos.



## get-merge-commit

En el siguiente ejemplo de código, se muestra cómo usar `get-merge-commit`.

### AWS CLI

Para obtener información detallada sobre una confirmación de fusión

En el siguiente `get-merge-commit` ejemplo, se muestran detalles sobre una confirmación de fusión para la rama de origen nombrada `bugfix-bug1234` con una rama de destino denominada `main` mediante la MERGE estrategia `THREE_WAY` en un repositorio denominado `MyDemoRepo`.

```
aws codecommit get-merge-commit \
  --source-commit-specifier bugfix-bug1234 \
  --destination-commit-specifier main \
  --merge-option THREE_WAY_MERGE \
  --repository-name MyDemoRepo
```

Salida:

```
{
  "sourceCommitId": "c5709475EXAMPLE",
  "destinationCommitId": "317f8570EXAMPLE",
  "baseCommitId": "fb12a539EXAMPLE",
  "mergeCommitId": "ffc4d608eEXAMPLE"
}
```

Para obtener más información, consulta [Ver los detalles de la confirmación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [GetMergeCommit](#) la Referencia de AWS CLI comandos.

## get-merge-conflicts

En el siguiente ejemplo de código, se muestra cómo usar `get-merge-conflicts`.

### AWS CLI

Para ver si hay algún conflicto de fusión en una solicitud de extracción

En el siguiente `get-merge-conflicts` ejemplo, se muestra si hay algún conflicto de fusión entre la punta de una rama de origen denominada `feature-randomizationfeature` y una rama de destino denominada «`main`» en un repositorio denominado `MyDemoRepo`.

```
aws codecommit get-merge-conflicts \  
  --repository-name MyDemoRepo \  
  --source-commit-specifier feature-randomizationfeature \  
  --destination-commit-specifier main \  
  --merge-option THREE_WAY_MERGE
```

Salida:

```
{  
  "mergeable": false,  
  "destinationCommitId": "86958e0aEXAMPLE",  
  "sourceCommitId": "6ccd57fdEXAMPLE",  
  "baseCommitId": "767b6958EXAMPLE",  
  "conflictMetadataList": [  
    {  
      "filePath": "readme.md",  
      "fileSizes": {  
        "source": 139,  
        "destination": 230,  
        "base": 85  
      },  
      "fileModes": {  
        "source": "NORMAL",  
        "destination": "NORMAL",  
        "base": "NORMAL"  
      },  
      "objectTypes": {  
        "source": "FILE",  
        "destination": "FILE",  
        "base": "FILE"  
      },  
      "numberOfConflicts": 1,  
      "isBinaryFile": {  
        "source": false,  
        "destination": false,  
        "base": false  
      },  
      "contentConflict": true,  
      "fileModeConflict": false,  
    }  
  ]  
}
```

```

        "objectTypeConflict": false,
        "mergeOperations": {
            "source": "M",
            "destination": "M"
        }
    }
]
}

```

- Para API obtener más información, consulte [GetMergeConflicts](#) la Referencia de AWS CLI comandos.

## get-merge-options

En el siguiente ejemplo de código, se muestra cómo usar `get-merge-options`.

### AWS CLI

Para obtener información sobre las opciones de combinación disponibles para combinar dos ramas específicas

El siguiente `get-merge-options` ejemplo determina las opciones de combinación disponibles para fusionar una rama de origen nombrada `bugfix-bug1234` con una rama de destino nombrada `main` en un repositorio denominado `MyDemoRepo`

```

aws codecommit get-merge-options \
  --source-commit-specifier bugfix-bug1234 \
  --destination-commit-specifier main \
  --repository-name MyDemoRepo

```

Salida:

```

{
  "mergeOptions": [
    "FAST_FORWARD_MERGE",
    "SQUASH_MERGE",
    "THREE_WAY_MERGE"
  ],
  "sourceCommitId": "18059494EXAMPLE",
  "destinationCommitId": "ffd3311dEXAMPLE",
  "baseCommitId": "ffd3311dEXAMPLE"
}

```

```
}
```

Para obtener más información, consulta la sección [Resolución de conflictos en una solicitud de extracción](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [GetMergeOptions](#) la Referencia de AWS CLI comandos.

## get-pull-request-approval-states

En el siguiente ejemplo de código, se muestra cómo usar `get-pull-request-approval-states`.

### AWS CLI

Para ver las aprobaciones de una solicitud de extracción

El siguiente `get-pull-request-approval-states` ejemplo devuelve las aprobaciones de la solicitud de extracción especificada.

```
aws codecommit get-pull-request-approval-states \  
  --pull-request-id 8 \  
  --revision-id 9f29d167EXAMPLE
```

Salida:

```
{  
  "approvals": [  
    {  
      "userArn": "arn:aws:iam::123456789012:user/Mary_Major",  
      "approvalState": "APPROVE"  
    }  
  ]  
}
```

Para obtener más información, consulta [Ver las solicitudes de extracción](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [GetPullRequestApprovalStates](#) la Referencia de AWS CLI comandos.

## get-pull-request-override-state

En el siguiente ejemplo de código, se muestra cómo usar `get-pull-request-override-state`.

### AWS CLI

Para obtener información sobre el estado de anulación de una solicitud de extracción

El siguiente `get-pull-request-override-state` ejemplo devuelve el estado de anulación de la solicitud de extracción especificada. En este ejemplo, una usuaria llamada Mary Major anuló las reglas de aprobación de la solicitud de extracción, por lo que el resultado devuelve un valor de `true` :

```
aws codecommit get-pull-request-override-state \  
  --pull-request-id 34 \  
  --revision-id 9f29d167EXAMPLE
```

Salida:

```
{  
  "overridden": true,  
  "overrider": "arn:aws:iam::123456789012:user/Mary_Major"  
}
```

Para obtener más información, consulte [Anular las reglas de aprobación en una solicitud de extracción](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [GetPullRequestOverrideState](#) la Referencia de AWS CLI comandos.

## get-pull-request

En el siguiente ejemplo de código, se muestra cómo usar `get-pull-request`.

### AWS CLI

Para ver los detalles de una solicitud de extracción

En este ejemplo se muestra cómo ver la información sobre una solicitud de extracción con el ID de27.

```
aws codecommit get-pull-request \  
--pull-request-id 27
```

Salida:

```
{  
  "pullRequest": {  
    "approvalRules": [  
      {  
        "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"Statements\":  
[{\n\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":  
[\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]]\"",  
        "approvalRuleId": "dd8b17fe-EXAMPLE",  
        "approvalRuleName": "2-approver-rule-for-main",  
        "creationDate": 1571356106.936,  
        "lastModifiedDate": 571356106.936,  
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",  
        "ruleContentSha256": "4711b576EXAMPLE"  
      }  
    ],  
    "lastActivityDate": 1562619583.565,  
    "pullRequestTargets": [  
      {  
        "sourceCommit": "ca45e279EXAMPLE",  
        "sourceReference": "refs/heads/bugfix-1234",  
        "mergeBase": "a99f5ddbEXAMPLE",  
        "destinationReference": "refs/heads/main",  
        "mergeMetadata": {  
          "isMerged": false  
        },  
        "destinationCommit": "2abfc6beEXAMPLE",  
        "repositoryName": "MyDemoRepo"  
      }  
    ],  
    "revisionId": "e47def21EXAMPLE",  
    "title": "Quick fix for bug 1234",  
    "authorArn": "arn:aws:iam::123456789012:user/Nikhil_Jayashankar",  
    "clientRequestToken": "d8d7612e-EXAMPLE",  
    "creationDate": 1562619583.565,  
    "pullRequestId": "27",  
    "pullRequestStatus": "OPEN"  
  }  
}
```

```
}
```

- Para API obtener más información, consulte [GetPullRequest](#) la Referencia de AWS CLI comandos.

## get-repository-triggers

En el siguiente ejemplo de código, se muestra cómo usar `get-repository-triggers`.

### AWS CLI

Para obtener información sobre los activadores de un repositorio

En este ejemplo, se muestran detalles sobre los desencadenadores configurados para un AWS CodeCommit repositorio denominado `MyDemoRepo`.

```
aws codecommit get-repository-triggers \  
  --repository-name MyDemoRepo
```

Salida:

```
{  
  "configurationId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",  
  "triggers": [  
    {  
      "destinationArn": "arn:aws:sns:us-  
east-1:111111111111:MyCodeCommitTopic",  
      "branches": [  
        "main",  
        "preprod"  
      ],  
      "name": "MyFirstTrigger",  
      "customData": "",  
      "events": [  
        "all"  
      ]  
    },  
    {  
      "destinationArn": "arn:aws:lambda:us-  
east-1:111111111111:function:MyCodeCommitPythonFunction",  
      "branches": [],
```

```

        "name": "MySecondTrigger",
        "customData": "EXAMPLE",
        "events": [
            "all"
        ]
    }
]
}

```

- Para API obtener más información, consulte [GetRepositoryTriggers](#) la Referencia de AWS CLI comandos.

## get-repository

En el siguiente ejemplo de código, se muestra cómo usar `get-repository`.

### AWS CLI

Para obtener información sobre un repositorio

En este ejemplo se muestran detalles sobre un AWS CodeCommit repositorio.

```

aws codecommit get-repository \
  --repository-name MyDemoRepo

```

Salida:

```

{
  "repositoryMetadata": {
    "creationDate": 1429203623.625,
    "defaultBranch": "main",
    "repositoryName": "MyDemoRepo",
    "cloneUrlSsh": "ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos/v1/
repos/MyDemoRepo",
    "lastModifiedDate": 1430783812.0869999,
    "repositoryDescription": "My demonstration repository",
    "cloneUrlHttp": "https://codecommit.us-east-1.amazonaws.com/v1/repos/
MyDemoRepo",
    "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
    "Arn": "arn:aws:codecommit:us-east-1:80398EXAMPLE:MyDemoRepo",
    "accountId": "111111111111"
  }
}

```



```
}
```

- Para API obtener más información, consulte [GetRepository](#) la Referencia de AWS CLI comandos.

## **list-approval-rule-templates**

En el siguiente ejemplo de código, se muestra cómo usar `list-approval-rule-templates`.

### AWS CLI

Para enumerar todas las plantillas de reglas de aprobación de una AWS región

En el siguiente `list-approval-rule-templates` ejemplo, se enumeran todas las plantillas de reglas de aprobación de la región especificada. Si no se especifica ninguna AWS región como parámetro, el comando devuelve las plantillas de reglas de aprobación para la región especificada en el AWS CLI perfil utilizado para ejecutar el comando.

```
aws codecommit list-approval-rule-templates \  
  --region us-east-2
```

Salida:

```
{  
  "approvalRuleTemplateName": [  
    "2-approver-rule-for-main",  
    "1-approver-rule-for-all-pull-requests"  
  ]  
}
```

Para obtener más información, consulte [Gestión de plantillas de reglas de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [ListApprovalRuleTemplates](#) la Referencia de AWS CLI comandos.

## **list-associated-approval-rule-templates-for-repository**

En el siguiente ejemplo de código, se muestra cómo usar `list-associated-approval-rule-templates-for-repository`.

## AWS CLI

Para ver una lista de todas las plantillas asociadas a un repositorio

En el siguiente `list-associated-approval-rule-templates-for-repository` ejemplo, se enumeran todas las plantillas de reglas de aprobación asociadas a un repositorio denominado `MyDemoRepo`.

```
aws codecommit list-associated-approval-rule-templates-for-repository \  
  --repository-name MyDemoRepo
```

Salida:

```
{  
  "approvalRuleTemplateNames": [  
    "2-approver-rule-for-main",  
    "1-approver-rule-for-all-pull-requests"  
  ]  
}
```

Para obtener más información, consulte [Gestión de plantillas de reglas de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [ListAssociatedApprovalRuleTemplatesForRepository](#) la Referencia de AWS CLI comandos.

## list-branches

En el siguiente ejemplo de código, se muestra cómo usar `list-branches`.

### AWS CLI

Para ver una lista de nombres de sucursales

En este ejemplo se muestran todos los nombres de las ramas de un AWS CodeCommit repositorio.

```
aws codecommit list-branches \  
  --repository-name MyDemoRepo
```

Salida:

```
{
  "branches": [
    "MyNewBranch",
    "main"
  ]
}
```

- Para API obtener más información, consulte [ListBranches](#) la Referencia de AWS CLI comandos.

## list-pull-requests

En el siguiente ejemplo de código, se muestra cómo usar `list-pull-requests`.

AWS CLI

Para ver una lista de solicitudes de incorporación de cambios en un repositorio

En este ejemplo, se muestra cómo enumerar las solicitudes de extracción creadas por un IAM usuario con el nombre ARN 'arn:aws:iam: :1111:user/li\_juan' y el estado " en un repositorio llamado ": CLOSED AWS CodeCommit MyDemoRepo

```
aws codecommit list-pull-requests --author-arn arn:aws:iam::111111111111:user/Li_Juan --pull-request-status CLOSED --repository-name MyDemoRepo
```

Salida:

```
{
  "nextToken": "",
  "pullRequestIds": ["2", "12", "16", "22", "23", "35", "30", "39", "47"]
}
```

- API [ListPullRequests](#) Para AWS CLI obtener más información, consulte la Referencia de comandos.

## list-repositories-for-approval-rule-template

En el siguiente ejemplo de código, se muestra cómo usar `list-repositories-for-approval-rule-template`.

## AWS CLI

Para ver una lista de todos los repositorios asociados a una plantilla

En el siguiente `list-repositories-for-approval-rule-template` ejemplo, se enumeran todos los repositorios asociados a la plantilla de reglas de aprobación especificada.

```
aws codecommit list-repositories-for-approval-rule-template \  
--approval-rule-template-name 2-approver-rule-for-main
```

Salida:

```
{  
  "repositoryNames": [  
    "MyDemoRepo",  
    "MyClonedRepo"  
  ]  
}
```

Para obtener más información, consulte [Gestión de plantillas de reglas de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [ListRepositoriesForApprovalRuleTemplate](#) la Referencia de AWS CLI comandos.

## **list-repositories**

En el siguiente ejemplo de código, se muestra cómo usar `list-repositories`.

## AWS CLI

Para ver una lista de repositorios

En este ejemplo, se enumeran todos los AWS CodeCommit repositorios asociados a la cuenta del AWS usuario.

Comando:

```
aws codecommit list-repositories
```

Salida:

```
{
  "repositories": [
    {
      "repositoryName": "MyDemoRepo"
      "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
    },
    {
      "repositoryName": "MyOtherDemoRepo"
      "repositoryId": "cfc29ac4-b0cb-44dc-9990-f6f51EXAMPLE"
    }
  ]
}
```

- Para API obtener más información, consulte [ListRepositories](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para ver las AWS etiquetas de un repositorio

En el siguiente `list-tags-for-resource` ejemplo, se muestran las claves y los valores de las etiquetas del repositorio especificado.

```
aws codecommit list-tags-for-resource \
  --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo
```

Salida:

```
{
  "tags": {
    "Status": "Secret",
    "Team": "Saanvi"
  }
}
```

Para obtener más información, consulte [Ver las etiquetas de un repositorio](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## merge-branches-by-fast-forward

En el siguiente ejemplo de código, se muestra cómo usar `merge-branches-by-fast-forward`.

### AWS CLI

Para fusionar dos ramas mediante la estrategia de fusión rápida

En el siguiente `merge-branches-by-fast-forward` ejemplo, se fusiona la rama de origen especificada con la rama de destino especificada en un repositorio denominado `MyDemoRepo`

```
aws codecommit merge-branches-by-fast-forward \  
  --source-commit-specifier bugfix-bug1234 \  
  --destination-commit-specifier bugfix-bug1233 \  
  --repository-name MyDemoRepo
```

Salida:

```
{  
  "commitId": "4f178133EXAMPLE",  
  "treeId": "389765daEXAMPLE"  
}
```

Para obtener más información, consulte [Comparar y combinar ramas](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [MergeBranchesByFastForward](#) la Referencia de AWS CLI comandos.

## merge-branches-by-squash

En el siguiente ejemplo de código, se muestra cómo usar `merge-branches-by-squash`.

### AWS CLI

Para fusionar dos ramas utilizando la estrategia de fusión por aplastamiento

El siguiente `merge-branches-by-squash` ejemplo fusiona la rama de origen especificada con la rama de destino especificada en un repositorio denominado `MyDemoRepo`.

```
aws codecommit merge-branches-by-squash \  
  --source-commit-specifier bugfix-bug1234 \  
  --destination-commit-specifier bugfix-bug1233 \  
  --author-name "Maria Garcia" \  
  --email "maria_garcia@example.com" \  
  --commit-message "Merging two fix branches to prepare for a general patch." \  
  --repository-name MyDemoRepo
```

Salida:

```
{  
  "commitId": "4f178133EXAMPLE",  
  "treeId": "389765daEXAMPLE"  
}
```

Para obtener más información, consulte [Comparar y combinar ramas](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [MergeBranchesBySquash](#) la Referencia de AWS CLI comandos.

## merge-branches-by-three-way

En el siguiente ejemplo de código, se muestra cómo usar merge-branches-by-three-way.

### AWS CLI

Para fusionar dos ramas utilizando la estrategia de fusión a tres bandas

En el siguiente merge-branches-by-three-way ejemplo, se fusiona la rama de origen especificada con la rama de destino especificada en un repositorio denominado. MyDemoRepo

```
aws codecommit merge-branches-by-three-way \  
  --source-commit-specifier main \  
  --destination-commit-specifier bugfix-bug1234 \  
  --author-name "Jorge Souza" --email "jorge_souza@example.com" \  
  --commit-message "Merging changes from main to bugfix branch before additional testing." \  
  --repository-name MyDemoRepo
```

Salida:

```
{
  "commitId": "4f178133EXAMPLE",
  "treeId": "389765daEXAMPLE"
}
```

Para obtener más información, consulte [Comparar y combinar ramas](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [MergeBranchesByThreeWay](#) la Referencia de AWS CLI comandos.

## merge-pull-request-by-fast-forward

En el siguiente ejemplo de código, se muestra cómo usar `merge-pull-request-by-fast-forward`.

### AWS CLI

Para fusionar y cerrar una solicitud de extracción

En este ejemplo, se muestra cómo fusionar y cerrar una solicitud de extracción con el identificador «47» y el identificador de confirmación de origen «99132ab0EXAMPLE» en un repositorio denominado. MyDemoRepo

```
aws codecommit merge-pull-request-by-fast-forward \
  --pull-request-id 47 \
  --source-commit-id 99132ab0EXAMPLE \
  --repository-name MyDemoRepo
```

Salida:

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"Statements\": [
          {\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers\": [
            \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "I want one approver for this pull request",
        "creationDate": 1571356106.936,

```



```

        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "ruleContentSha256": "4711b576EXAMPLE"
    }
],
"authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
"clientRequestToken": "",
"creationDate": 1508530823.142,
"description": "Review the latest changes and updates to the global
variables",
"lastActivityDate": 1508887223.155,
"pullRequestId": "47",
"pullRequestStatus": "CLOSED",
"pullRequestTargets": [
    {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
            "isMerged": true,
            "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
        },
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
    }
],
"title": "Consolidation of global variables"
}
}

```

Para obtener más información, consulta [Fusionar una solicitud de extracción](#) en la Guía del usuario.AWS CodeCommit

- Para API obtener más información, consulte [MergePullRequestByFastForward](#) la Referencia de AWS CLI comandos.

## merge-pull-request-by-squash

En el siguiente ejemplo de código, se muestra cómo usar merge-pull-request-by-squash.

### AWS CLI

Para fusionar una solicitud de extracción mediante la estrategia de fusión automática

El siguiente `merge-pull-request-by-squash` ejemplo fusiona y cierra la solicitud de extracción especificada mediante la estrategia de resolución de conflictos de `ACCEPT_SOURCE` en un repositorio denominado `MyDemoRepo`.

```
aws codecommit merge-pull-request-by-squash \
  --pull-request-id 47 \
  --source-commit-id 99132ab0EXAMPLE \
  --repository-name MyDemoRepo \
  --conflict-detail-level LINE_LEVEL \
  --conflict-resolution-strategy ACCEPT_SOURCE \
  --name "Jorge Souza" --email "jorge_souza@example.com" \
  --commit-message "Merging pull request 47 by squash and accepting source in
merge conflicts"
```

Salida:

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\",
DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":
[\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approver-rule-for-main",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "originApprovalRuleTemplate": {
          "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
          "approvalRuleTemplateName": "2-approver-rule-for-main"
        },
        "ruleContentSha256": "4711b576EXAMPLE"
      }
    ],
    "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
    "clientRequestToken": "",
    "creationDate": 1508530823.142,
    "description": "Review the latest changes and updates to the global
variables",
    "lastActivityDate": 1508887223.155,
    "pullRequestId": "47",
```

```

    "pullRequestStatus": "CLOSED",
    "pullRequestTargets": [
      {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
          "isMerged": true,
          "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
        },
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
      }
    ],
    "title": "Consolidation of global variables"
  }
}

```

Para obtener más información, consulta [Fusionar una solicitud de extracción](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [MergePullRequestBySquash](#) la Referencia de AWS CLI comandos.

## merge-pull-request-by-three-way

En el siguiente ejemplo de código, se muestra cómo usar `merge-pull-request-by-three-way`.

### AWS CLI

Para fusionar una solicitud de extracción mediante la estrategia de fusión a tres bandas

En el siguiente `merge-pull-request-by-three-way` ejemplo, se fusiona y cierra la solicitud de extracción especificada mediante las opciones predeterminadas de detalle y estrategia de resolución de conflictos en un repositorio denominado. `MyDemoRepo`

```

aws codecommit merge-pull-request-by-three-way \
  --pull-request-id 47 \
  --source-commit-id 99132ab0EXAMPLE \
  --repository-name MyDemoRepo \
  --name "Maria Garcia" \
  --email "maria_garcia@example.com" \

```

```
--commit-message "Merging pull request 47 by three-way with default options"
```

Salida:

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\",
        \"DestinationReferences\": [\"refs/heads/main\"], \"Statements\": [{\"Type
        \": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":
        [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approver-rule-for-main",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "originApprovalRuleTemplate": {
          "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
          "approvalRuleTemplateName": "2-approver-rule-for-main"
        },
        "ruleContentSha256": "4711b576EXAMPLE"
      }
    ],
    "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
    "clientRequestToken": "",
    "creationDate": 1508530823.142,
    "description": "Review the latest changes and updates to the global
    variables",
    "lastActivityDate": 1508887223.155,
    "pullRequestId": "47",
    "pullRequestStatus": "CLOSED",
    "pullRequestTargets": [
      {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
          "isMerged": true,
          "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
        },
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
      }
    ]
  }
}
```

```

    }
  ],
  "title": "Consolidation of global variables"
}
}

```

Para obtener más información, consulta [Fusionar una solicitud de extracción](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [MergePullRequestByThreeWay](#) la Referencia de AWS CLI comandos.

## override-pull-request-approval-rules

En el siguiente ejemplo de código, se muestra cómo usar `override-pull-request-approval-rules`.

### AWS CLI

Para anular los requisitos de las reglas de aprobación en una solicitud de extracción

El siguiente `override-pull-request-approval-rules` ejemplo anula las reglas de aprobación de la solicitud de extracción especificada. En su lugar, para revocar una anulación, establece el `--override-status` valor del parámetro en. `REVOKE`

```

aws codecommit override-pull-request-approval-rules \
  --pull-request-id 34 \
  --revision-id 927df8d8EXAMPLE \
  --override-status OVERRIDE

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Anular las reglas de aprobación en una solicitud de extracción](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [OverridePullRequestApprovalRules](#) la Referencia de AWS CLI comandos.

## post-comment-for-compared-commit

En el siguiente ejemplo de código, se muestra cómo usar `post-comment-for-compared-commit`.

## AWS CLI

Para crear un comentario en una confirmación

En este ejemplo se muestra cómo añadir el comentario "Can you add a test case for this?" sobre el cambio al `cl_sample.js` archivo en la comparación entre dos confirmaciones de un repositorio denominado `MyDemoRepo`.

```
aws codecommit post-comment-for-compared-commit \  
  --repository-name MyDemoRepo \  
  --before-commit-id 317f8570EXAMPLE \  
  --after-commit-id 5d036259EXAMPLE \  
  --client-request-token 123Example \  
  --content "Can you add a test case for this?" \  
  --location filePath=cl_sample.js,filePosition=1232,relativeFileVersion=AFTER
```

Salida:

```
{  
  "afterBlobId": "1f330709EXAMPLE",  
  "afterCommitId": "317f8570EXAMPLE",  
  "beforeBlobId": "80906a4cEXAMPLE",  
  "beforeCommitId": "6e147360EXAMPLE",  
  "comment": {  
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
    "clientRequestToken": "",  
    "commentId": "553b509bEXAMPLE56198325",  
    "content": "Can you add a test case for this?",  
    "creationDate": 1508369612.203,  
    "deleted": false,  
    "commentId": "abc123-EXAMPLE",  
    "lastModifiedDate": 1508369612.203,  
    "callerReactions": [],  
    "reactionCounts": []  
  },  
  "location": {  
    "filePath": "cl_sample.js",  
    "filePosition": 1232,  
    "relativeFileVersion": "AFTER"  
  },  
  "repositoryName": "MyDemoRepo"  
}
```

```
}

```

- Para API obtener más información, consulte [PostCommentForComparedCommit](#) la Referencia de AWS CLI comandos.

## post-comment-for-pull-request

En el siguiente ejemplo de código, se muestra cómo usar `post-comment-for-pull-request`.

### AWS CLI

Para añadir un comentario a una solicitud de cambios

En el siguiente `post-comment-for-pull-request` ejemplo, se añade el comentario: «Parece que no se utilizan en ningún sitio. ¿Se podrían eliminar?» al cambiar el `ahs_count.py` fichero en una solicitud de cambios con el identificador 47 de un repositorio denominado `MyDemoRepo`.

```
aws codecommit post-comment-for-pull-request \
  --pull-request-id "47" \
  --repository-name MyDemoRepo \
  --before-commit-id 317f8570EXAMPLE \
  --after-commit-id 5d036259EXAMPLE \
  --client-request-token 123Example \
  --content "These don't appear to be used anywhere. Can we remove them?" \
  --location filePath=ahs_count.py,filePosition=367,relativeFileVersion=AFTER
```

Salida:

```
{
  "afterBlobId": "1f330709EXAMPLE",
  "afterCommitId": "5d036259EXAMPLE",
  "beforeBlobId": "80906a4cEXAMPLE",
  "beforeCommitId": "317f8570EXAMPLE",
  "comment": {
    "authorArn": "arn:aws:iam::111111111111:user/Saanvi_Sarkar",
    "clientRequestToken": "123Example",
    "commentId": "abcd1234EXAMPLEb5678efgh",
    "content": "These don't appear to be used anywhere. Can we remove
them?",
    "creationDate": 1508369622.123,
    "deleted": false,
    "CommentId": ""
```

```

        "lastModifiedDate": 1508369622.123,
        "callerReactions": [],
        "reactionCounts": []
    },
    "location": {
        "filePath": "ahs_count.py",
        "filePosition": 367,
        "relativeFileVersion": "AFTER"
    },
    "repositoryName": "MyDemoRepo",
    "pullRequestId": "47"
}

```

- Para API obtener más información, consulte [PostCommentForPullRequest](#) la Referencia de AWS CLI comandos.

## post-comment-reply

En el siguiente ejemplo de código, se muestra cómo usar `post-comment-reply`.

### AWS CLI

Para responder a un comentario en una confirmación o en una solicitud de cambios

En este ejemplo se muestra cómo añadir la respuesta "Good catch. I'll remove them." al comentario con el identificador generado por el sistema de. `abcd1234EXAMPLEb5678efgh`

```

aws codecommit post-comment-reply \
  --in-reply-to abcd1234EXAMPLEb5678efgh \
  --content "Good catch. I'll remove them." \
  --client-request-token 123Example

```

Salida:

```

{
  "comment": {
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
    "clientRequestToken": "123Example",
    "commentId": "442b498bEXAMPLE5756813",
    "content": "Good catch. I'll remove them.",
    "creationDate": 1508369829.136,
    "deleted": false,

```



```
    "CommentId": "abcd1234EXAMPLEb5678efgh",
    "lastModifiedDate": 150836912.221,
    "callerReactions": [],
    "reactionCounts": []
  }
}
```

- Para API obtener más información, consulte la Referencia [PostCommentReply](#) de AWS CLI comandos.

## put-comment-reaction

En el siguiente ejemplo de código, se muestra cómo usar `put-comment-reaction`.

### AWS CLI

Para responder a un comentario de una confirmación con un emoji

En el siguiente `put-comment-reaction` ejemplo, se responde a un comentario con el identificador de `abcd1234EXAMPLEb5678efgh` y un valor de reacción de un emoji igual a `:thumbsup:`.

```
aws codecommit put-comment-reaction \
  --comment-id abcd1234EXAMPLEb5678efgh \
  --reaction-value :thumbsup:
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Comentar una confirmación AWS CodeCommit en la Guía del AWS CodeCommit usuario](#).

- Para API obtener más información, consulte [PutCommentReaction](#) la Referencia de AWS CLI comandos.

## put-file

En el siguiente ejemplo de código, se muestra cómo usar `put-file`.

### AWS CLI

Para añadir un archivo a un repositorio

En el siguiente `put-file` ejemplo, se agrega un archivo llamado «`ExampleSolution.py`» a un repositorio llamado «`MyDemoRepo`», a una rama llamada «`feature-randomizationfeature`» cuya confirmación más reciente tiene el identificador «`4c925148`». EXAMPLE

```
aws codecommit put-file \  
  --repository-name MyDemoRepo \  
  --branch-name feature-randomizationfeature \  
  --file-content file://MyDirectory/ExampleSolution.py \  
  --file-path /solutions/ExampleSolution.py \  
  --parent-commit-id 4c925148EXAMPLE \  
  --name "Maria Garcia" \  
  --email "maria_garcia@example.com" \  
  --commit-message "I added a third randomization routine."
```

Salida:

```
{  
  "blobId": "2eb4af3bEXAMPLE",  
  "commitId": "317f8570EXAMPLE",  
  "treeId": "347a3408EXAMPLE"  
}
```

- Para AWS CLI obtener más información, consulta la Referencia de comandos. API [PutFile](#)

## **put-repository-triggers**

En el siguiente ejemplo de código, se muestra cómo usar `put-repository-triggers`.

AWS CLI

Para añadir o actualizar un disparador en un repositorio

En este ejemplo, se muestra cómo actualizar los activadores denominados `MyFirstTrigger` " y `MySecondTrigger` " mediante un JSON archivo ya creado (en este caso denominado `MyTriggers.json`) que contiene la estructura de todos los activadores de un repositorio denominado. `MyDemoRepo` Para obtener información sobre cómo obtener los JSON activadores existentes, consulta el comando. `get-repository-triggers`

```
aws codecommit put-repository-triggers \  
  --repository-name MyDemoRepo file://MyTriggers.json
```

## Contenidos de MyTriggers.json:

```
{
  "repositoryName": "MyDemoRepo",
  "triggers": [
    {
      "destinationArn": "arn:aws:sns:us-
east-1:80398EXAMPLE:MyCodeCommitTopic",
      "branches": [
        "main",
        "preprod"
      ],
      "name": "MyFirstTrigger",
      "customData": "",
      "events": [
        "all"
      ]
    },
    {
      "destinationArn": "arn:aws:lambda:us-
east-1:111111111111:function:MyCodeCommitPythonFunction",
      "branches": [],
      "name": "MySecondTrigger",
      "customData": "EXAMPLE",
      "events": [
        "all"
      ]
    }
  ]
}
```

## Salida:

```
{
  "configurationId": "6fa51cd8-35c1-EXAMPLE"
}
```

- Para API obtener más información, consulte [PutRepositoryTriggers](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir AWS etiquetas a un repositorio existente

El siguiente `tag-resource` ejemplo etiqueta el repositorio especificado con dos etiquetas.

```
aws codecommit tag-resource \  
  --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo \  
  --tags Status=Secret,Team=Saanvi
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir una etiqueta a un repositorio](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## test-repository-triggers

En el siguiente ejemplo de código, se muestra cómo usar `test-repository-triggers`.

### AWS CLI

Para probar los activadores en un repositorio

En este ejemplo se muestra cómo probar un disparador denominado "MyFirstTrigger" en un AWS CodeCommit repositorio denominado MyDemoRepo. En este ejemplo, los eventos del repositorio activan notificaciones desde un tema de Amazon Simple Notification Service (AmazonSNS).

Comando:

```
aws codecommit test-repository-triggers --repository-name MyDemoRepo \  
  --triggers name=MyFirstTrigger,destinationArn=arn:aws:sns:us-east-1:111111111111:MyCodeCommitTopic,branches=mainline,preprod,events=all
```

Salida:

```
{
  "successfulExecutions": [
    "MyFirstTrigger"
  ],
  "failedExecutions": []
}
```

- Para API obtener más información, consulte [TestRepositoryTriggers](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar AWS etiquetas de un repositorio

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta con la clave especificada del repositorio denominado `MyDemoRepo`.

```
aws codecommit untag-resource \
  --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo \
  --tag-keys Status
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar una etiqueta de un repositorio](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-approval-rule-template-content

En el siguiente ejemplo de código, se muestra cómo usar `update-approval-rule-template-content`.

### AWS CLI

Para actualizar el contenido de una plantilla de reglas de aprobación

El siguiente `update-approval-rule-template-content` ejemplo cambia el contenido de la plantilla de reglas de aprobación especificada para redefinir el grupo de aprobación para los usuarios que asumen la función de `CodeCommitReview`.

```
aws codecommit update-approval-rule-template-content \
  --approval-rule-template-name 1-approver-rule \
  --new-rule-content "{\"Version\": \"2018-11-08\", \"DestinationReferences\": [\"refs/heads/main\"], \"Statements\": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"
```

Salida:

```
{
  "approvalRuleTemplate": {
    "creationDate": 1571352720.773,
    "approvalRuleTemplateDescription": "Requires 1 approval for all pull requests from the CodeCommitReview pool",
    "lastModifiedDate": 1571358728.41,
    "approvalRuleTemplateId": "41de97b7-EXAMPLE",
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"Statements\": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "ruleContentSha256": "2f6c21a5EXAMPLE",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan"
  }
}
```

Para obtener más información, consulte [Gestión de plantillas de reglas de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [UpdateApprovalRuleTemplateContent](#) la Referencia de AWS CLI comandos.

## update-approval-rule-template-description

En el siguiente ejemplo de código, se muestra cómo usar `update-approval-rule-template-description`.

## AWS CLI

Para actualizar la descripción de una plantilla de reglas de aprobación

En el siguiente `update-approval-rule-template-description` ejemplo, se cambia la descripción de la plantilla de regla de aprobación especificada a `Requires 1 approval for all pull requests from the CodeCommitReview pool.`:

```
aws codecommit update-approval-rule-template-description \  
  --approval-rule-template-name 1-approver-rule-for-all-pull-requests \  
  --approval-rule-template-description "Requires 1 approval for all pull requests  
from the CodeCommitReview pool"
```

Salida:

```
{  
  "approvalRuleTemplate": {  
    "creationDate": 1571352720.773,  
    "approvalRuleTemplateDescription": "Requires 1 approval for all pull requests  
from the CodeCommitReview pool",  
    "lastModifiedDate": 1571358728.41,  
    "approvalRuleTemplateId": "41de97b7-EXAMPLE",  
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"Statements\":  
[{\n\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers\":  
[\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",  
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",  
    "ruleContentSha256": "2f6c21a5EXAMPLE",  
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan"  
  }  
}
```

Para obtener más información, consulte [Gestión de plantillas de reglas de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [UpdateApprovalRuleTemplateDescription](#) la Referencia de AWS CLI comandos.

### **update-approval-rule-template-name**

En el siguiente ejemplo de código, se muestra cómo usar `update-approval-rule-template-name`.

## AWS CLI

Para actualizar el nombre de una plantilla de reglas de aprobación

En el siguiente `update-approval-rule-template-name` ejemplo, se cambia el nombre de una plantilla de reglas de aprobación `1-approver-rule` a `1-approver-rule-for-all-pull-requests`.

```
aws codecommit update-approval-rule-template-name \
  --old-approval-rule-template-name 1-approver-rule \
  --new-approval-rule-template-name 1-approver-rule-for-all-pull-requests
```

Salida:

```
{
  "approvalRuleTemplate": {
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "lastModifiedDate": 1571358241.619,
    "approvalRuleTemplateId": "41de97b7-EXAMPLE",
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"Statements\": [
    [\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers\": [
    [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]]]]}",
    "creationDate": 1571352720.773,
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
    "approvalRuleTemplateDescription": "All pull requests must be approved by one
    developer on the team.",
    "ruleContentSha256": "2f6c21a5cEXAMPLE"
  }
}
```

Para obtener más información, consulte [Gestión de plantillas de reglas de aprobación](#) en la Guía del usuario.AWS CodeCommit

- Para API obtener más información, consulte [UpdateApprovalRuleTemplateName](#) la Referencia de AWS CLI comandos.

## update-comment

En el siguiente ejemplo de código, se muestra cómo usar `update-comment`.



## AWS CLI

Para actualizar un comentario en una confirmación

En este ejemplo se muestra cómo añadir el contenido "Fixed as requested. I'll update the pull request." a un comentario con un identificador de 442b498bEXAMPLE5756813.

```
aws codecommit update-comment \  
  --comment-id 442b498bEXAMPLE5756813 \  
  --content "Fixed as requested. I'll update the pull request."
```

Salida:

```
{  
  "comment": {  
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
    "clientRequestToken": "",  
    "commentId": "442b498bEXAMPLE5756813",  
    "content": "Fixed as requested. I'll update the pull request.",  
    "creationDate": 1508369929.783,  
    "deleted": false,  
    "lastModifiedDate": 1508369929.287,  
    "callerReactions": [],  
    "reactionCounts":  
      {  
        "THUMBSUP" : 2  
      }  
  }  
}
```

- Para API obtener más información, consulte [UpdateComment](#) la Referencia de AWS CLI comandos.

## update-default-branch

En el siguiente ejemplo de código, se muestra cómo usar `update-default-branch`.

## AWS CLI

Para cambiar la rama predeterminada de un repositorio

En este ejemplo se cambia la rama predeterminada de un AWS CodeCommit repositorio. Este comando devuelve resultados solo si hay errores.

Comando:

```
aws codecommit update-default-branch --repository-name MyDemoRepo --default-branch-name MyNewBranch
```

Salida:

```
None.
```

- Para API obtener más información, consulte [UpdateDefaultBranch](#) la Referencia de AWS CLI comandos.

## update-pull-request-approval-rule-content

En el siguiente ejemplo de código, se muestra cómo usar `update-pull-request-approval-rule-content`.

AWS CLI

Para editar una regla de aprobación para una solicitud de extracción

En el siguiente `update-pull-request-approval-rule-content` ejemplo, se actualiza la regla de aprobación especificada para requerir la aprobación de un usuario de un grupo de aprobaciones que incluye a todos los IAM usuarios de la 123456789012 AWS cuenta.

```
aws codecommit update-pull-request-approval-rule-content \  
  --pull-request-id 27 \  
  --approval-rule-name "Require two approved approvers" \  
  --approval-rule-content "{Version: 2018-11-08, Statements: [{Type: \  
  \"Approvers\", NumberOfApprovalsNeeded: 1, ApprovalPoolMembers: \  
  [\"CodeCommitApprovers:123456789012:user/*\"]}]}"
```

Salida:

```
{  
  "approvalRule": {
```

```

    "approvalRuleContent": "{Version: 2018-11-08, Statements:
  [{Type: \"Approvers\", NumberOfApprovalsNeeded: 1, ApprovalPoolMembers:
  [\"CodeCommitApprovers:123456789012:user/*\"]}]}",
    "approvalRuleId": "aac33506-EXAMPLE",
    "originApprovalRuleTemplate": {},
    "creationDate": 1570752871.932,
    "lastModifiedDate": 1570754058.333,
    "approvalRuleName": "Require two approved approvers",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
    "ruleContentSha256": "cd93921cEXAMPLE",
  }
}

```

Para obtener más información, consulte [Editar o eliminar una regla de aprobación](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [UpdatePullRequestApprovalRuleContent](#) la Referencia de AWS CLI comandos.

## update-pull-request-approval-state

En el siguiente ejemplo de código, se muestra cómo usar `update-pull-request-approval-state`.

### AWS CLI

Para aprobar o revocar la aprobación de una solicitud de extracción

En el siguiente `update-pull-request-approval-state` ejemplo, se aprueba una solicitud de extracción con el identificador de 27 y el identificador de revisión de. 9f29d167EXAMPLE Si en vez de eso quieres revocar la aprobación, establece el valor del `--approval-state` parámetro en. REVOKE

```

aws codecommit update-pull-request-approval-state \
  --pull-request-id 27 \
  --revision-id 9f29d167EXAMPLE \
  --approval-state "APPROVE"

```

Este comando no genera ninguna salida.

Para obtener más información, consulta Cómo [revisar una solicitud de retirada](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [UpdatePullRequestApprovalState](#) la Referencia de AWS CLI comandos.

## update-pull-request-description

En el siguiente ejemplo de código, se muestra cómo usar `update-pull-request-description`.

AWS CLI

Para cambiar la descripción de una solicitud de extracción

En este ejemplo se muestra cómo cambiar la descripción de una solicitud de extracción con el ID de 47.

```
aws codecommit update-pull-request-description \  
  --pull-request-id 47 \  
  --description "Updated the pull request to remove unused global variable."
```

Salida:

```
{  
  "pullRequest": {  
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
    "clientRequestToken": "",  
    "creationDate": 1508530823.155,  
    "description": "Updated the pull request to remove unused global variable.",  
    "lastActivityDate": 1508372423.204,  
    "pullRequestId": "47",  
    "pullRequestStatus": "OPEN",  
    "pullRequestTargets": [  
      {  
        "destinationCommit": "9f31c968EXAMPLE",  
        "destinationReference": "refs/heads/main",  
        "mergeMetadata": {  
          "isMerged": false,  
        },  
        "repositoryName": "MyDemoRepo",  
        "sourceCommit": "99132ab0EXAMPLE",  
        "sourceReference": "refs/heads/variables-branch"  
      }  
    ],  
    "title": "Consolidation of global variables"  
  }  
}
```

```
}
}
```

- Para API obtener más información, consulte [UpdatePullRequestDescription](#) la Referencia de AWS CLI comandos.

## update-pull-request-status

En el siguiente ejemplo de código, se muestra cómo usar `update-pull-request-status`.

### AWS CLI

Para cambiar el estado de una solicitud de extracción

En este ejemplo se muestra cómo cambiar el estado de una solicitud de extracción con el ID de 42 a un estado de CLOSED en un AWS CodeCommit repositorio denominado MyDemoRepo.

```
aws codecommit update-pull-request-status \
  --pull-request-id 42 \
  --pull-request-status CLOSED
```

Salida:

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"Statements\": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approvers-needed-for-this-change",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "ruleContentSha256": "4711b576EXAMPLE"
      }
    ],
    "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
    "clientRequestToken": "",
    "creationDate": 1508530823.165,
    "description": "Updated the pull request to remove unused global variable.",
```

```

    "lastActivityDate": 1508372423.12,
    "pullRequestId": "47",
    "pullRequestStatus": "CLOSED",
    "pullRequestTargets": [
      {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
          "isMerged": false,
        },
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
      }
    ],
    "title": "Consolidation of global variables"
  }
}

```

- Para API obtener más información, consulte [UpdatePullRequestStatus](#) la Referencia de AWS CLI comandos.

## update-pull-request-title

En el siguiente ejemplo de código, se muestra cómo usar `update-pull-request-title`.

### AWS CLI

Para cambiar el título de una solicitud de cambios

En este ejemplo se muestra cómo cambiar el título de una solicitud de extracción con el identificador de 47.

```

aws codecommit update-pull-request-title \
  --pull-request-id 47 \
  --title "Consolidation of global variables - updated review"

```

Salida:

```

{
  "pullRequest": {
    "approvalRules": [

```

```

    {
      "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\\\"DestinationReferences\": [\\\"refs/heads/main\\\"],\\\"Statements\": [{\\\"Type
\\\": \\\"Approvers\\\",\\\"NumberOfApprovalsNeeded\": 2,\\\"ApprovalPoolMembers\":
[\\\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\\\"]}]}",
      "approvalRuleId": "dd8b17fe-EXAMPLE",
      "approvalRuleName": "2-approver-rule-for-main",
      "creationDate": 1571356106.936,
      "lastModifiedDate": 571356106.936,
      "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
      "originApprovalRuleTemplate": {
        "approvalRuleTemplateId": "dd8b26gr-EXAMPLE",
        "approvalRuleTemplateName": "2-approver-rule-for-main"
      },
      "ruleContentSha256": "4711b576EXAMPLE"
    },
    {
      "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
      "clientRequestToken": "",
      "creationDate": 1508530823.12,
      "description": "Review the latest changes and updates to the global
variables. I have updated this request with some changes, including removing some
unused variables.",
      "lastActivityDate": 1508372657.188,
      "pullRequestId": "47",
      "pullRequestStatus": "OPEN",
      "pullRequestTargets": [
        {
          "destinationCommit": "9f31c968EXAMPLE",
          "destinationReference": "refs/heads/main",
          "mergeMetadata": {
            "isMerged": false,
          },
          "repositoryName": "MyDemoRepo",
          "sourceCommit": "99132ab0EXAMPLE",
          "sourceReference": "refs/heads/variables-branch"
        }
      ],
      "title": "Consolidation of global variables - updated review"
    }
  ]
}

```

- Para API obtener más información, consulte [UpdatePullRequestTitle](#) la Referencia de AWS CLI comandos.

## update-repository-description

En el siguiente ejemplo de código, se muestra cómo usar `update-repository-description`.

### AWS CLI

Para cambiar la descripción de un repositorio

En este ejemplo se cambia la descripción de un AWS CodeCommit repositorio. Este comando devuelve resultados solo si hay errores.

Comando:

```
aws codecommit update-repository-description --repository-name MyDemoRepo --  
repository-description "This description was changed"
```

Salida:

```
None.
```

- Para API obtener más información, consulte [UpdateRepositoryDescription](#) la Referencia de AWS CLI comandos.

## update-repository-name

En el siguiente ejemplo de código, se muestra cómo usar `update-repository-name`.

### AWS CLI

Para cambiar el nombre de un repositorio

En este ejemplo se cambia el nombre de un AWS CodeCommit repositorio. Este comando devuelve resultados solo si hay errores. Al cambiar el nombre del AWS CodeCommit repositorio, se cambiará el SSH y HTTPS URLs que los usuarios necesitan para conectarse al repositorio. Para que los usuarios puedan conectarse a este repositorio, tendrán que actualizar la configuración de la conexión. Además, dado que el repositorio ARN cambiará, cambiar el nombre del repositorio invalidará cualquier política de IAM usuario que dependa del ARN repositorio.



Comando:

```
aws codecommit update-repository-name --old-name MyDemoRepo --new-name MyRenamedDemoRepo
```

Salida:

```
None.
```

- Para API obtener más información, consulte [UpdateRepositoryName](#) la Referencia de AWS CLI comandos.

## CodeDeploy ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with CodeDeploy.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **add-tags-to-on-premises-instances**

En el siguiente ejemplo de código, se muestra cómo usar `add-tags-to-on-premises-instances`.

AWS CLI

Para añadir etiquetas a las instancias locales

En el siguiente `add-tags-to-on-premises-instances` ejemplo, se asocia AWS CodeDeploy la misma etiqueta de instancia local a dos instancias locales. No registra las instancias locales con. AWS CodeDeploy

```
aws deploy add-tags-to-on-premises-instances \  
  --instance-names AssetTag12010298EX AssetTag23121309EX \  
  --tags Key=Name,Value=CodeDeployDemo-OnPrem
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [AddTagsToOnPremisesInstances](#) la Referencia de AWS CLI comandos.

## batch-get-application-revisions

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-application-revisions`.

### AWS CLI

Para recuperar información sobre las revisiones de la aplicación

En el siguiente `batch-get-application-revisions` ejemplo, se recupera información sobre la revisión especificada almacenada en un GitHub repositorio.

```
aws deploy batch-get-application-revisions \  
  --application-name my-codedeploy-application \  
  --revisions "[{\\"githubLocation\\": {\\"commitId\\":  
  \\"fa85936EXAMPLEa31736c051f10d77297EXAMPLE\\",\\"repository\\": \\"my-github-token/my-  
  repository\\"},\\"revisionType\\": \\"GitHub\\"}]"
```

Salida:

```
{  
  "revisions": [  
    {  
      "genericRevisionInfo": {  
        "description": "Application revision registered by Deployment ID: d-  
A1B2C3111",  
        "lastUsedTime": 1556912355.884,  
        "registerTime": 1556912355.884,  
        "firstUsedTime": 1556912355.884,  
        "deploymentGroups": []
```

```

    },
    "revisionLocation": {
      "revisionType": "GitHub",
      "gitHubLocation": {
        "commitId": "fa85936EXAMPLEa31736c051f10d77297EXAMPLE",
        "repository": "my-github-token/my-repository"
      }
    }
  ],
  "applicationName": "my-codedeploy-application",
  "errorMessage": ""
}

```

Para obtener más información, consulte [BatchGetApplicationRevisions](#) la AWS CodeDeploy API Referencia.

- Para API obtener más información, consulte [BatchGetApplicationRevisions](#) la Referencia de AWS CLI comandos.

## batch-get-applications

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-applications`.

### AWS CLI

Para obtener información sobre varias aplicaciones

En el siguiente `batch-get-applications` ejemplo, se muestra información sobre varias aplicaciones asociadas a la AWS cuenta del usuario.

```
aws deploy batch-get-applications --application-names WordPress_App MyOther_App
```

Salida:

```

{
  "applicationsInfo": [
    {
      "applicationName": "WordPress_App",
      "applicationId": "d9dd6993-f171-44fa-a811-211e4EXAMPLE",
      "createTime": 1407878168.078,
      "linkedToGitHub": false
    },
  ],
}

```

```

    {
      "applicationName": "MyOther_App",
      "applicationId": "8ca57519-31da-42b2-9194-8bb16EXAMPLE",
      "createTime": 1407453571.63,
      "linkedToGitHub": false
    }
  ]
}

```

- Para API obtener más información, consulte [BatchGetApplications](#) la Referencia de AWS CLI comandos.

## batch-get-deployment-groups

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-deployment-groups`.

### AWS CLI

Para recuperar información sobre uno o más grupos de despliegue

En el siguiente `batch-get-deployment-groups` ejemplo, se recupera información sobre dos de los grupos de implementación que están asociados a la CodeDeploy aplicación especificada.

```

aws deploy batch-get-deployment-groups \
  --application-name my-codedeploy-application \
  --deployment-group-names ["my-deployment-group-1","my-deployment-group-2"]

```

Salida:

```

{
  "deploymentGroupsInfo": [
    {
      "deploymentStyle": {
        "deploymentOption": "WITHOUT_TRAFFIC_CONTROL",
        "deploymentType": "IN_PLACE"
      },
      "autoRollbackConfiguration": {
        "enabled": false
      },
      "onPremisesTagSet": {
        "onPremisesTagSetList": []
      },
    },
  ],
}

```

```

    "serviceRoleArn": "arn:aws:iam::123456789012:role/
CodeDeployServiceRole",
    "lastAttemptedDeployment": {
        "endTime": 1556912366.415,
        "status": "Failed",
        "createTime": 1556912355.884,
        "deploymentId": "d-A1B2C3111"
    },
    "autoScalingGroups": [],
    "deploymentGroupName": "my-deployment-group-1",
    "ec2TagSet": {
        "ec2TagSetList": [
            [
                {
                    "Type": "KEY_AND_VALUE",
                    "Value": "my-EC2-instance",
                    "Key": "Name"
                }
            ]
        ]
    },
    "deploymentGroupId": "a1b2c3d4-5678-90ab-cdef-11111example",
    "triggerConfigurations": [],
    "applicationName": "my-codedeploy-application",
    "computePlatform": "Server",
    "deploymentConfigName": "CodeDeployDefault.AllAtOnce"
},
{
    "deploymentStyle": {
        "deploymentOption": "WITHOUT_TRAFFIC_CONTROL",
        "deploymentType": "IN_PLACE"
    },
    "autoRollbackConfiguration": {
        "enabled": false
    },
    "onPremisesTagSet": {
        "onPremisesTagSetList": []
    },
    "serviceRoleArn": "arn:aws:iam::123456789012:role/
CodeDeployServiceRole",
    "autoScalingGroups": [],
    "deploymentGroupName": "my-deployment-group-2",
    "ec2TagSet": {
        "ec2TagSetList": [

```

```

        [
            {
                "Type": "KEY_AND_VALUE",
                "Value": "my-EC2-instance",
                "Key": "Name"
            }
        ]
    ],
    "deploymentGroupId": "a1b2c3d4-5678-90ab-cdef-2222example",
    "triggerConfigurations": [],
    "applicationName": "my-codedeploy-application",
    "computePlatform": "Server",
    "deploymentConfigName": "CodeDeployDefault.AllAtOnce"
}
],
"errorMessage": ""
}

```

Para obtener más información, consulte [BatchGetDeploymentGroups](#) la AWS CodeDeploy API Referencia.

- Para API obtener más información, consulte [BatchGetDeploymentGroups](#) la Referencia de AWS CLI comandos.

## batch-get-deployment-targets

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-deployment-targets`.

### AWS CLI

Para recuperar los objetivos asociados a un despliegue

El siguiente `batch-get-deployment-targets` ejemplo devuelve información sobre uno de los objetivos asociados a la implementación especificada.

```

aws deploy batch-get-deployment-targets \
  --deployment-id "d-1A2B3C4D5" \
  --target-ids "i-01a2b3c4d5e6f1111"

```

Salida:

```
{
```

```
"deploymentTargets": [
  {
    "deploymentTargetType": "InstanceTarget",
    "instanceTarget": {
      "lifecycleEvents": [
        {
          "startTime": 1556918592.162,
          "lifecycleEventName": "ApplicationStop",
          "status": "Succeeded",
          "endTime": 1556918592.247,
          "diagnostics": {
            "scriptName": "",
            "errorCode": "Success",
            "logTail": "",
            "message": "Succeeded"
          }
        },
        {
          "startTime": 1556918593.193,
          "lifecycleEventName": "DownloadBundle",
          "status": "Succeeded",
          "endTime": 1556918593.981,
          "diagnostics": {
            "scriptName": "",
            "errorCode": "Success",
            "logTail": "",
            "message": "Succeeded"
          }
        },
        {
          "startTime": 1556918594.805,
          "lifecycleEventName": "BeforeInstall",
          "status": "Succeeded",
          "endTime": 1556918681.807,
          "diagnostics": {
            "scriptName": "",
            "errorCode": "Success",
            "logTail": "",
            "message": "Succeeded"
          }
        }
      ],
      "targetArn": "arn:aws:ec2:us-west-2:123456789012:instance/i-01a2b3c4d5e6f1111",
    }
  }
]
```

```

        "deploymentId": "d-1A2B3C4D5",
        "lastUpdatedAt": 1556918687.504,
        "targetId": "i-01a2b3c4d5e6f1111",
        "status": "Succeeded"
    }
}
]
}

```

Para obtener más información, consulte [BatchGetDeploymentTargets](#) la AWS CodeDeploy API Referencia.

- Para API obtener más información, consulte [BatchGetDeploymentTargets](#) la Referencia de AWS CLI comandos.

## batch-get-deployments

En el siguiente ejemplo de código, se muestra cómo usar batch-get-deployments.

### AWS CLI

Para obtener información sobre varios despliegues

En el siguiente batch-get-deployments ejemplo, se muestra información sobre varios despliegues que están asociados a la cuenta del AWS usuario.

```
aws deploy batch-get-deployments --deployment-ids d-A1B2C3111 d-A1B2C3222
```

Salida:

```

{
  "deploymentsInfo": [
    {
      "applicationName": "WordPress_App",
      "status": "Failed",
      "deploymentOverview": {
        "Failed": 0,
        "InProgress": 0,
        "Skipped": 0,
        "Succeeded": 1,
        "Pending": 0
      },
      "deploymentConfigName": "CodeDeployDefault.OneAtATime",
    }
  ]
}

```



```

    "creator": "user",
    "deploymentGroupName": "WordPress_DG",
    "revision": {
      "revisionType": "S3",
      "s3Location": {
        "bundleType": "zip",
        "version": "uTecLusEXAMPLEFXtfUcyfV8bEXAMPLE",
        "bucket": "CodeDeployDemoBucket",
        "key": "WordPressApp.zip"
      }
    },
    "deploymentId": "d-A1B2C3111",
    "createTime": 1408480721.9,
    "completeTime": 1408480741.822
  },
  {
    "applicationName": "MyOther_App",
    "status": "Failed",
    "deploymentOverview": {
      "Failed": 1,
      "InProgress": 0,
      "Skipped": 0,
      "Succeeded": 0,
      "Pending": 0
    },
    "deploymentConfigName": "CodeDeployDefault.OneAtATime",
    "creator": "user",
    "errorInformation": {
      "message": "Deployment failed: Constraint default violated: No hosts
succeeded.",
      "code": "HEALTH_CONSTRAINTS"
    },
    "deploymentGroupName": "MyOther_DG",
    "revision": {
      "revisionType": "S3",
      "s3Location": {
        "bundleType": "zip",
        "eTag": "\"dd56cfdEXAMPLE8e768f9d77fEXAMPLE\"",
        "bucket": "CodeDeployDemoBucket",
        "key": "MyOtherApp.zip"
      }
    },
    "deploymentId": "d-A1B2C3222",
    "createTime": 1409764576.589,

```

```

        "completeTime": 1409764596.101
      }
    ]
  }

```

- Para API obtener más información, consulte [BatchGetDeployments](#) la Referencia de AWS CLI comandos.

## batch-get-on-premises-instances

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-on-premises-instances`.

### AWS CLI

Para obtener información sobre una o más instancias locales

En el siguiente `batch-get-on-premises-instances` ejemplo, se obtiene información sobre dos instancias locales.

```

aws deploy batch-get-on-premises-instances --instance-
names AssetTag12010298EX AssetTag23121309EX

```

Salida:

```

{
  "instanceInfos": [
    {
      "iamUserArn": "arn:aws:iam::123456789012:user/AWS/CodeDeploy/
AssetTag12010298EX",
      "tags": [
        {
          "Value": "CodeDeployDemo-OnPrem",
          "Key": "Name"
        }
      ],
      "instanceName": "AssetTag12010298EX",
      "registerTime": 1425579465.228,
      "instanceArn": "arn:aws:codedeploy:us-west-2:123456789012:instance/
AssetTag12010298EX_4IwLNI2Alh"
    },
    {

```

```

        "iamUserArn": "arn:aws:iam::123456789012:user/AWS/CodeDeploy/
AssetTag23121309EX",
        "tags": [
            {
                "Value": "CodeDeployDemo-OnPrem",
                "Key": "Name"
            }
        ],
        "instanceName": "AssetTag23121309EX",
        "registerTime": 1425595585.988,
        "instanceArn": "arn:aws:codedeploy:us-west-2:80398EXAMPLE:instance/
AssetTag23121309EX_PomUy64Was"
    }
]
}

```

- Para API obtener más información, consulte [BatchGetOnPremisesInstances](#) la Referencia de AWS CLI comandos.

## continue-deployment

En el siguiente ejemplo de código, se muestra cómo usar `continue-deployment`.

### AWS CLI

Para empezar a redirigir el tráfico sin esperar a que transcurra un tiempo de espera especificado.

El siguiente `continue-deployment` ejemplo comienza a redirigir el tráfico desde las instancias del entorno original que están listas para empezar a transferir el tráfico a las instancias del entorno de reemplazo.

```

aws deploy continue-deployment \
  --deployment-id "d-A1B2C3111" \
  --deployment-wait-type "READY_WAIT"

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [ContinueDeployment](#) la AWS CodeDeploy API Referencia.

- Para API obtener más información, consulte [ContinueDeployment](#) la Referencia de AWS CLI comandos.

## create-application

En el siguiente ejemplo de código, se muestra cómo usar `create-application`.

### AWS CLI

Para crear una aplicación

El siguiente `create-application` ejemplo crea una aplicación y la asocia a la AWS cuenta del usuario.

```
aws deploy create-application --application-name MyOther_App
```

Salida:

```
{
  "applicationId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
}
```

- Para API obtener más información, consulte [CreateApplication](#) la Referencia de AWS CLI comandos.

## create-deployment-config

En el siguiente ejemplo de código, se muestra cómo usar `create-deployment-config`.

### AWS CLI

Para crear una configuración de despliegue personalizada

En el siguiente `create-deployment-config` ejemplo, se crea una configuración de despliegue personalizada y se asocia a la AWS cuenta del usuario.

```
aws deploy create-deployment-config \
  --deployment-config-name ThreeQuartersHealthy \
  --minimum-healthy-hosts type=FLEET_PERCENT,value=75
```

Salida:

```
{
  "deploymentConfigId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
}
```

```
}
```

- Para API obtener más información, consulte [CreateDeploymentConfig](#) la Referencia de AWS CLI comandos.

## create-deployment-group

En el siguiente ejemplo de código, se muestra cómo usar `create-deployment-group`.

### AWS CLI

Para crear un grupo de despliegue

En el siguiente `create-deployment-group` ejemplo, se crea un grupo de despliegue y se asocia a la aplicación especificada y a la AWS cuenta del usuario.

```
aws deploy create-deployment-group \  
  --application-name WordPress_App \  
  --auto-scaling-groups CodeDeployDemo-ASG \  
  --deployment-config-name CodeDeployDefault.OneAtATime \  
  --deployment-group-name WordPress_DG \  
  --ec2-tag-filters Key=Name,Value=CodeDeployDemo,Type=KEY_AND_VALUE \  
  --service-role-arn arn:aws:iam::123456789012:role/CodeDeployDemoRole
```

Salida:

```
{  
  "deploymentGroupId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"  
}
```

- Para API obtener más información, consulte [CreateDeploymentGroup](#) la Referencia de AWS CLI comandos.

## create-deployment

En el siguiente ejemplo de código, se muestra cómo usar `create-deployment`.

### AWS CLI

Ejemplo 1: Para crear una CodeDeploy implementación mediante la plataforma informática EC2 / On-Premise

En el siguiente create-deployment ejemplo, se crea una implementación y se asocia a la cuenta del AWS usuario.

```
aws deploy create-deployment \
  --application-name WordPress_App \
  --deployment-config-name CodeDeployDefault.OneAtATime \
  --deployment-group-name WordPress_DG \
  --description "My demo deployment" \
  --s3-
location bucket=CodeDeployDemoBucket,bundleType=zip,eTag=dd56cfEXAMPLE8e768f9d77fEXAMPLE,ke
```

Salida:

```
{
  "deploymentId": "d-A1B2C3111"
}
```

Ejemplo 2: Para crear una CodeDeploy implementación con la plataforma de ECS cómputo Amazon

En el siguiente create-deployment ejemplo, se utilizan los dos archivos siguientes para implementar un ECS servicio de Amazon.

Contenido del archivo create-deployment.json:

```
{
  "applicationName": "ecs-deployment",
  "deploymentGroupName": "ecs-deployment-dg",
  "revision": {
    "revisionType": "S3",
    "s3Location": {
      "bucket": "ecs-deployment-bucket",
      "key": "appspec.yaml",
      "bundleType": "YAML"
    }
  }
}
```

Ese archivo, a su vez, recupera el siguiente archivo appspec.yaml de un bucket de S3 llamado ecs-deployment-bucket.

```
version: 0.0
Resources:
  - TargetService:
    Type: AWS::ECS::Service
    Properties:
      TaskDefinition: "arn:aws:ecs:region:123456789012:task-definition/ecs-task-
def:2"
      LoadBalancerInfo:
        ContainerName: "sample-app"
        ContainerPort: 80
        PlatformVersion: "LATEST"
```

Comando:

```
aws deploy create-deployment \
  --cli-input-json file://create-deployment.json \
  --region us-east-1
```

Salida:

```
{
  "deploymentId": "d-1234ABCDE"
}
```

Para obtener más información, consulte [CreateDeployment](#) la AWS CodeDeploy API Referencia.

- Para API obtener más información, consulte [CreateDeployment](#) la Referencia de AWS CLI comandos.

## delete-application

En el siguiente ejemplo de código, se muestra cómo usar delete-application.

### AWS CLI

Eliminación de una aplicación

En el siguiente delete-application ejemplo, se elimina la aplicación especificada que está asociada a la AWS cuenta del usuario.

```
aws deploy delete-application --application-name WordPress_App
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteApplication](#) la Referencia de AWS CLI comandos.

## delete-deployment-config

En el siguiente ejemplo de código, se muestra cómo usar `delete-deployment-config`.

### AWS CLI

Para eliminar una configuración de despliegue

En el siguiente `delete-deployment-config` ejemplo, se elimina una configuración de despliegue personalizada que está asociada a la AWS cuenta del usuario.

```
aws deploy delete-deployment-config --deployment-config-name ThreeQuartersHealthy
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteDeploymentConfig](#) la Referencia de AWS CLI comandos.

## delete-deployment-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-deployment-group`.

### AWS CLI

Para eliminar un grupo de despliegue

En el siguiente `delete-deployment-group` ejemplo, se elimina un grupo de despliegue que está asociado a la aplicación especificada.

```
aws deploy delete-deployment-group \  
  --application-name WordPress_App \  
  --deployment-group-name WordPress_DG
```

Salida:

```
{  
  "hooksNotCleanedUp": []
```



```
}
```

- Para API obtener más información, consulte [DeleteDeploymentGroup](#) la Referencia de AWS CLI comandos.

## **delete-git-hub-account-token**

En el siguiente ejemplo de código, se muestra cómo usar `delete-git-hub-account-token`.

### AWS CLI

Para eliminar la conexión de una GitHub cuenta

En el siguiente `delete-git-hub-account-token` ejemplo, se elimina la conexión de la cuenta especificada GitHub .

```
aws deploy delete-git-hub-account-token --token-name my-github-account
```

Salida:

```
{
  "tokenName": "my-github-account"
}
```

Para obtener más información, consulte [DeleteGitHubAccountToken](#) la AWS CodeDeploy API Referencia.

- Para API obtener más información, consulte [DeleteGitHubAccountToken](#) la Referencia de AWS CLI comandos.

## **deregister-on-premises-instance**

En el siguiente ejemplo de código, se muestra cómo usar `deregister-on-premises-instance`.

### AWS CLI

Para anular el registro de una instancia local

En el siguiente `deregister-on-premises-instance` ejemplo, se anula el registro de una instancia local AWS CodeDeploy, pero no se elimina el IAM usuario asociado a la instancia ni se disocia de la instancia en las etiquetas de AWS CodeDeploy la instancia local. Tampoco

desinstala el AWS CodeDeploy agente de la instancia ni elimina el archivo de configuración local de la instancia.

```
aws deploy deregister-on-premises-instance --instance-name AssetTag12010298EX
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeregisterOnPremisesInstance](#) la Referencia de AWS CLI comandos.

## deregister

En el siguiente ejemplo de código, se muestra cómo usar `deregister`.

### AWS CLI

Para anular el registro de una instancia local

En el siguiente `deregister` ejemplo se anula el registro de una instancia local con. AWS CodeDeploy No elimina el IAM usuario asociado a la instancia. Las etiquetas locales se disocian de AWS CodeDeploy la instancia en las etiquetas locales. No desinstala el AWS CodeDeploy agente de la instancia ni elimina el archivo de configuración local de la instancia.

```
aws deploy deregister \  
  --instance-name AssetTag12010298EX \  
  --no-delete-iam-user \  
  --region us-west-2
```

Salida:

```
Retrieving on-premises instance information... DONE  
IamUserArn: arn:aws:iam::80398EXAMPLE:user/AWS/CodeDeploy/AssetTag12010298EX  
Tags: Key=Name,Value=CodeDeployDemo-OnPrem  
Removing tags from the on-premises instance... DONE  
Deregistering the on-premises instance... DONE  
Run the following command on the on-premises instance to uninstall the codedeploy-agent:  
aws deploy uninstall
```

- Para API obtener más información, consulte [Anular el registro](#) en AWS CLI la referencia de comandos.

## get-application-revision

En el siguiente ejemplo de código, se muestra cómo usar `get-application-revision`.

### AWS CLI

Para obtener información sobre la revisión de una aplicación

En el siguiente `get-application-revision` ejemplo, se muestra información sobre una revisión de la aplicación asociada a la aplicación especificada.

```
aws deploy get-application-revision \  
  --application-name WordPress_App \  
  --s3-  
location bucket=CodeDeployDemoBucket,bundleType=zip,eTag=dd56cfdEXAMPLE8e768f9d77fEXAMPLE,ke
```

Salida:

```
{  
  "applicationName": "WordPress_App",  
  "revisionInfo": {  
    "description": "Application revision registered by Deployment ID: d-  
A1B2C3111",  
    "registerTime": 1411076520.009,  
    "deploymentGroups": "WordPress_DG",  
    "lastUsedTime": 1411076520.009,  
    "firstUsedTime": 1411076520.009  
  },  
  "revision": {  
    "revisionType": "S3",  
    "s3Location": {  
      "bundleType": "zip",  
      "eTag": "dd56cfdEXAMPLE8e768f9d77fEXAMPLE",  
      "bucket": "CodeDeployDemoBucket",  
      "key": "WordPressApp.zip"  
    }  
  }  
}
```

- Para API obtener más información, consulte [GetApplicationRevision](#) la Referencia de AWS CLI comandos.

## get-application

En el siguiente ejemplo de código, se muestra cómo usar `get-application`.

### AWS CLI

Para obtener información sobre una aplicación

En el siguiente `get-application` ejemplo, se muestra información sobre una aplicación asociada a la AWS cuenta del usuario.

```
aws deploy get-application --application-name WordPress_App
```

Salida:

```
{
  "application": {
    "applicationName": "WordPress_App",
    "applicationId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "createTime": 1407878168.078,
    "linkedToGitHub": false
  }
}
```

- Para API obtener más información, consulte [GetApplication](#) la Referencia de AWS CLI comandos.

## get-deployment-config

En el siguiente ejemplo de código, se muestra cómo usar `get-deployment-config`.

### AWS CLI

Para obtener información sobre una configuración de despliegue

En el siguiente `get-deployment-config` ejemplo, se muestra información sobre una configuración de despliegue asociada a la AWS cuenta del usuario.

```
aws deploy get-deployment-config --deployment-config-name ThreeQuartersHealthy
```

Salida:

```
{
  "deploymentConfigInfo": {
    "deploymentConfigId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "minimumHealthyHosts": {
      "type": "FLEET_PERCENT",
      "value": 75
    },
    "createTime": 1411081164.379,
    "deploymentConfigName": "ThreeQuartersHealthy"
  }
}
```

- Para API obtener más información, consulte [GetDeploymentConfig](#) la Referencia de AWS CLI comandos.

## get-deployment-group

En el siguiente ejemplo de código, se muestra cómo usar `get-deployment-group`.

AWS CLI

Para ver información sobre un grupo de despliegue

En el siguiente `get-deployment-group` ejemplo, se muestra información sobre un grupo de despliegues asociado a la aplicación especificada.

```
aws deploy get-deployment-group \
  --application-name WordPress_App \
  --deployment-group-name WordPress_DG
```

Salida:

```
{
  "deploymentGroupInfo": {
    "applicationName": "WordPress_App",
    "autoScalingGroups": [
      "CodeDeployDemo-ASG"
    ],
    "deploymentConfigName": "CodeDeployDefault.OneAtATime",
```

```

    "ec2TagFilters": [
      {
        "Type": "KEY_AND_VALUE",
        "Value": "CodeDeployDemo",
        "Key": "Name"
      }
    ],
    "deploymentGroupId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "serviceRoleArn": "arn:aws:iam::123456789012:role/CodeDeployDemoRole",
    "deploymentGroupName": "WordPress_DG"
  }
}

```

- Para API obtener más información, consulte [GetDeploymentGroup](#) la Referencia de AWS CLI comandos.

## get-deployment-instance

En el siguiente ejemplo de código, se muestra cómo usar `get-deployment-instance`.

### AWS CLI

Para obtener información sobre una instancia de despliegue

En el siguiente `get-deployment-instance` ejemplo, se muestra información sobre una instancia de despliegue que está asociada a la implementación especificada.

```
aws deploy get-deployment-instance --deployment-id d-QA4G4F9EX --instance-id i-902e9fEX
```

Salida:

```

{
  "instanceSummary": {
    "instanceId": "arn:aws:ec2:us-east-1:80398EXAMPLE:instance/i-902e9fEX",
    "lifecycleEvents": [
      {
        "status": "Succeeded",
        "endTime": 1408480726.569,
        "startTime": 1408480726.437,
        "lifecycleEventName": "ApplicationStop"
      }
    ],
  },
}

```

```
{
  "status": "Succeeded",
  "endTime": 1408480728.016,
  "startTime": 1408480727.665,
  "lifecycleEventName": "DownloadBundle"
},
{
  "status": "Succeeded",
  "endTime": 1408480729.744,
  "startTime": 1408480729.125,
  "lifecycleEventName": "BeforeInstall"
},
{
  "status": "Succeeded",
  "endTime": 1408480730.979,
  "startTime": 1408480730.844,
  "lifecycleEventName": "Install"
},
{
  "status": "Failed",
  "endTime": 1408480732.603,
  "startTime": 1408480732.1,
  "lifecycleEventName": "AfterInstall"
},
{
  "status": "Skipped",
  "endTime": 1408480732.606,
  "lifecycleEventName": "ApplicationStart"
},
{
  "status": "Skipped",
  "endTime": 1408480732.606,
  "lifecycleEventName": "ValidateService"
}
],
"deploymentId": "d-QA4G4F9EX",
"lastUpdatedAt": 1408480733.152,
"status": "Failed"
}
}
```

- Para API obtener más información, consulte [GetDeploymentInstance](#) la Referencia de AWS CLI comandos.

## get-deployment-target

En el siguiente ejemplo de código, se muestra cómo usar `get-deployment-target`.

### AWS CLI

Para devolver información sobre un objetivo de despliegue

El siguiente `get-deployment-target` ejemplo devuelve información sobre un objetivo de despliegue que está asociado a la implementación especificada.

```
aws deploy get-deployment-target \
  --deployment-id "d-A1B2C3111" \
  --target-id "i-a1b2c3d4e5f611111"
```

Salida:

```
{
  "deploymentTarget": {
    "deploymentTargetType": "InstanceTarget",
    "instanceTarget": {
      "lastUpdatedAt": 1556918687.504,
      "targetId": "i-a1b2c3d4e5f611111",
      "targetArn": "arn:aws:ec2:us-west-2:123456789012:instance/i-
a1b2c3d4e5f611111",
      "status": "Succeeded",
      "lifecycleEvents": [
        {
          "status": "Succeeded",
          "diagnostics": {
            "errorCode": "Success",
            "message": "Succeeded",
            "logTail": "",
            "scriptName": ""
          },
          "lifecycleEventName": "ApplicationStop",
          "startTime": 1556918592.162,
          "endTime": 1556918592.247
        },
        {
          "status": "Succeeded",
          "diagnostics": {
            "errorCode": "Success",
```



```
        "message": "Succeeded",
        "logTail": "",
        "scriptName": ""
    },
    "lifecycleEventName": "DownloadBundle",
    "startTime": 1556918593.193,
    "endTime": 1556918593.981
},
{
    "status": "Succeeded",
    "diagnostics": {
        "errorCode": "Success",
        "message": "Succeeded",
        "logTail": "",
        "scriptName": ""
    },
    "lifecycleEventName": "BeforeInstall",
    "startTime": 1556918594.805,
    "endTime": 1556918681.807
},
{
    "status": "Succeeded",
    "diagnostics": {
        "errorCode": "Success",
        "message": "Succeeded",
        "logTail": "",
        "scriptName": ""
    },
    "lifecycleEventName": "Install",
    "startTime": 1556918682.696,
    "endTime": 1556918683.005
},
{
    "status": "Succeeded",
    "diagnostics": {
        "errorCode": "Success",
        "message": "Succeeded",
        "logTail": "",
        "scriptName": ""
    },
    "lifecycleEventName": "AfterInstall",
    "startTime": 1556918684.135,
    "endTime": 1556918684.216
},
```

```
{
  "status": "Succeeded",
  "diagnostics": {
    "errorCode": "Success",
    "message": "Succeeded",
    "logTail": "",
    "scriptName": ""
  },
  "lifecycleEventName": "ApplicationStart",
  "startTime": 1556918685.211,
  "endTime": 1556918685.295
},
{
  "status": "Succeeded",
  "diagnostics": {
    "errorCode": "Success",
    "message": "Succeeded",
    "logTail": "",
    "scriptName": ""
  },
  "lifecycleEventName": "ValidateService",
  "startTime": 1556918686.65,
  "endTime": 1556918686.747
}
],
"deploymentId": "d-A1B2C3111"
}
}
```

Para obtener más información, consulte [GetDeploymentTarget](#) la AWS CodeDeploy API Referencia.

- Para API obtener más información, consulte [GetDeploymentTarget](#) la Referencia de AWS CLI comandos.

## get-deployment

En el siguiente ejemplo de código, se muestra cómo usar get-deployment.

### AWS CLI

Para obtener información sobre una implementación

En el siguiente `get-deployment` ejemplo, se muestra información sobre una implementación asociada a la AWS cuenta del usuario.

```
aws deploy get-deployment --deployment-id d-A1B2C3123
```

Salida:

```
{
  "deploymentInfo": {
    "applicationName": "WordPress_App",
    "status": "Succeeded",
    "deploymentOverview": {
      "Failed": 0,
      "InProgress": 0,
      "Skipped": 0,
      "Succeeded": 1,
      "Pending": 0
    },
    "deploymentConfigName": "CodeDeployDefault.OneAtATime",
    "creator": "user",
    "description": "My WordPress app deployment",
    "revision": {
      "revisionType": "S3",
      "s3Location": {
        "bundleType": "zip",
        "eTag": "\"dd56cfdEXAMPLE8e768f9d77fEXAMPLE\"",
        "bucket": "CodeDeployDemoBucket",
        "key": "WordPressApp.zip"
      }
    },
    "deploymentId": "d-A1B2C3123",
    "deploymentGroupName": "WordPress_DG",
    "createTime": 1409764576.589,
    "completeTime": 1409764596.101,
    "ignoreApplicationStopFailures": false
  }
}
```

- Para API obtener más información, consulte [GetDeployment](#) la Referencia de AWS CLI comandos.

## get-on-premises-instance

En el siguiente ejemplo de código, se muestra cómo usar `get-on-premises-instance`.

### AWS CLI

Para obtener información sobre una instancia local

En el siguiente `get-on-premises-instance` ejemplo, se recupera información sobre la instancia local especificada.

```
aws deploy get-on-premises-instance --instance-name AssetTag12010298EX
```

Salida:

```
{
  "instanceInfo": {
    "iamUserArn": "arn:aws:iam::123456789012:user/AWS/CodeDeploy/
AssetTag12010298EX",
    "tags": [
      {
        "Value": "CodeDeployDemo-OnPrem",
        "Key": "Name"
      }
    ],
    "instanceName": "AssetTag12010298EX",
    "registerTime": 1425579465.228,
    "instanceArn": "arn:aws:codedeploy:us-east-1:123456789012:instance/
AssetTag12010298EX_4IwLNI2Alh"
  }
}
```

- Para API obtener más información, consulte la Referencia [GetOnPremisesInstance](#) de AWS CLI comandos.

## install

En el siguiente ejemplo de código, se muestra cómo usar `install`.

### AWS CLI

Para instalar una instancia local

En el siguiente `install` ejemplo, se copia el archivo de configuración local de la ubicación especificada de la instancia a la ubicación de la instancia en la que el AWS CodeDeploy agente espera encontrarlo. También instala el AWS CodeDeploy agente en la instancia. No crea ningún IAM usuario, ni registra la instancia local ni asocia ninguna etiqueta de instancia local a la AWS CodeDeploy instancia. AWS CodeDeploy

```
aws deploy install \  
  --override-config \  
  --config-file C:\temp\codedeploy.onpremises.yml \  
  --region us-west-2 \  
  --agent-installer s3://aws-codedeploy-us-west-2/latest/codedeploy-agent.msi
```

Salida:

```
Creating the on-premises instance configuration file... DONE  
Installing the AWS CodeDeploy Agent... DONE
```

- Para API obtener más información, consulta la referencia sobre la [instalación](#) en AWS CLI comandos.

## list-application-revisions

En el siguiente ejemplo de código, se muestra cómo usar `list-application-revisions`.

AWS CLI

Para obtener información sobre las revisiones de la aplicación

En el siguiente `list-application-revisions` ejemplo, se muestra información sobre todas las revisiones de la aplicación que están asociadas a la aplicación especificada.

```
aws deploy list-application-revisions \  
  --application-name WordPress_App \  
  --s-3-bucket CodeDeployDemoBucket \  
  --deployed exclude \  
  --s-3-key-prefix WordPress_ \  
  --sort-by LastUsedTime \  
  --sort-order descending
```

Salida:

```
{
  "revisions": [
    {
      "revisionType": "S3",
      "s3Location": {
        "version": "uTecLusvCB_JqHFxtfUcyfV8bEXAMPLE",
        "bucket": "CodeDeployDemoBucket",
        "key": "WordPress_App.zip",
        "bundleType": "zip"
      }
    },
    {
      "revisionType": "S3",
      "s3Location": {
        "version": "tMk.UxgDpMEVb7V187ZM6wVAWEXAMPLE",
        "bucket": "CodeDeployDemoBucket",
        "key": "WordPress_App_2-0.zip",
        "bundleType": "zip"
      }
    }
  ]
}
```

- Para API obtener más información, consulte [ListApplicationRevisions](#) la Referencia de AWS CLI comandos.

## list-applications

En el siguiente ejemplo de código, se muestra cómo usar `list-applications`.

### AWS CLI

Para obtener información sobre las aplicaciones

En el siguiente `list-applications` ejemplo, se muestra información sobre todas las aplicaciones asociadas a la AWS cuenta del usuario.

```
aws deploy list-applications
```

Salida:

```
{
```

```
"applications": [  
  "WordPress_App",  
  "MyOther_App"  
]  
}
```

- Para API obtener más información, consulte [ListApplications](#) la Referencia de AWS CLI comandos.

## list-deployment-configs

En el siguiente ejemplo de código, se muestra cómo usar `list-deployment-configs`.

### AWS CLI

Para obtener información sobre las configuraciones de despliegue

En el siguiente `list-deployment-configs` ejemplo, se muestra información sobre todas las configuraciones de despliegue asociadas a la AWS cuenta del usuario.

```
aws deploy list-deployment-configs
```

Salida:

```
{  
  "deploymentConfigsList": [  
    "ThreeQuartersHealthy",  
    "CodeDeployDefault.AllAtOnce",  
    "CodeDeployDefault.HalfAtATime",  
    "CodeDeployDefault.OneAtATime"  
  ]  
}
```

- Para API obtener más información, consulte [ListDeploymentConfigs](#) la Referencia de AWS CLI comandos.

## list-deployment-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-deployment-groups`.

## AWS CLI

Para obtener información sobre los grupos de despliegue

En el siguiente `list-deployment-groups` ejemplo, se muestra información sobre todos los grupos de despliegue que están asociados a la aplicación especificada.

```
aws deploy list-deployment-groups --application-name WordPress_App
```

Salida:

```
{
  "applicationName": "WordPress_App",
  "deploymentGroups": [
    "WordPress_DG",
    "WordPress_Beta_DG"
  ]
}
```

- Para API obtener más información, consulte [ListDeploymentGroups](#) la Referencia de AWS CLI comandos.

## `list-deployment-instances`

En el siguiente ejemplo de código, se muestra cómo usar `list-deployment-instances`.

## AWS CLI

Para obtener información sobre las instancias de despliegue

En el siguiente `list-deployment-instances` ejemplo, se muestra información sobre todas las instancias de despliegue asociadas a la implementación especificada.

```
aws deploy list-deployment-instances \
  --deployment-id d-A1B2C3111 \
  --instance-status-filter Succeeded
```

Salida:

```
{
```



```

    "instancesList": [
      "i-EXAMPLE11",
      "i-EXAMPLE22"
    ]
  }

```

- Para API obtener más información, consulte [ListDeploymentInstances](#) la Referencia de AWS CLI comandos.

## list-deployment-targets

En el siguiente ejemplo de código, se muestra cómo usar `list-deployment-targets`.

### AWS CLI

Para recuperar una lista de IDs los objetivos asociados a un despliegue

El siguiente `list-deployment-targets` ejemplo recupera una lista de objetivos IDs asociados a las implementaciones que tienen el estado «Fallido» o «». InProgress

```

aws deploy list-deployment-targets \
  --deployment-id "d-A1B2C3111" \
  --target-filters "{\"TargetStatus\": [\"Failed\", \"InProgress\"]}"

```

Salida:

```

{
  "targetIds": [
    "i-0f1558aaf90e5f1f9"
  ]
}

```

Para obtener más información, consulte [ListDeploymentTargets](#) la AWS CodeDeploy API Referencia.

- Para API obtener más información, consulte [ListDeploymentTargets](#) la Referencia de AWS CLI comandos.

## list-deployments

En el siguiente ejemplo de código, se muestra cómo usar `list-deployments`.

## AWS CLI

Para obtener información sobre las implementaciones

En el siguiente `list-deployments` ejemplo, se muestra información sobre todas las implementaciones asociadas a la aplicación y al grupo de implementaciones especificados.

```
aws deploy list-deployments \  
  --application-name WordPress_App \  
  --create-time-range start=2014-08-19T00:00:00,end=2014-08-20T00:00:00 \  
  --deployment-group-name WordPress_DG \  
  --include-only-statuses Failed
```

Salida:

```
{  
  "deployments": [  
    "d-EXAMPLE11",  
    "d-EXAMPLE22",  
    "d-EXAMPLE33"  
  ]  
}
```

- Para API obtener más información, consulte [ListDeployments](#) la Referencia de AWS CLI comandos.

## `list-git-hub-account-token-names`

En el siguiente ejemplo de código, se muestra cómo usar `list-git-hub-account-token-names`.

## AWS CLI

Para ver una lista de los nombres de las conexiones almacenadas a GitHub las cuentas

En el siguiente `list-git-hub-account-token-names` ejemplo, se muestran los nombres de las conexiones almacenadas a GitHub las cuentas del AWS usuario actual.

```
aws deploy list-git-hub-account-token-names
```

Salida:

```
{
  "tokenNameList": [
    "my-first-token",
    "my-second-token",
    "my-third-token"
  ]
}
```

Para obtener más información, consulte [ListGitHubAccountTokenNames](#) la AWS CodeDeploy API Referencia.

- Para API obtener más información, consulte [ListGitHubAccountTokenNames](#) la Referencia de AWS CLI comandos.

## list-on-premises-instances

En el siguiente ejemplo de código, se muestra cómo usar `list-on-premises-instances`.

AWS CLI

Para obtener información sobre una o más instancias locales

En el siguiente `list-on-premises-instances` ejemplo, se recupera una lista de los nombres de instancias locales disponibles para las instancias que están registradas AWS CodeDeploy y que también tienen la etiqueta de instancia local especificada asociada a la instancia. AWS CodeDeploy

```
aws deploy list-on-premises-instances \
  --registration-status Registered \
  --tag-filters Key=Name,Value=CodeDeployDemo-OnPrem,Type=KEY_AND_VALUE
```

Salida:

```
{
  "instanceNames": [
    "AssetTag12010298EX"
  ]
}
```

- Para API obtener más información, consulta la Referencia de [ListOnPremisesInstances](#) comandos AWS CLI .

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un recurso (aplicación)

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas aplicadas a una aplicación denominada `testApp` en CodeDeploy.

```
aws deploy list-tags-for-resource \
  --resource-arn arn:aws:codedeploy:us-west-2:111122223333:application:testApp
```

Salida:

```
{
  "Tags": [
    {
      "Key": "Type",
      "Value": "testType"
    },
    {
      "Key": "Name",
      "Value": "testName"
    }
  ]
}
```

Para obtener más información, consulte [Etiquetar instancias para grupos de implementación CodeDeploy en](#) la Guía del AWS CodeDeploy usuario.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## push

En el siguiente ejemplo de código, se muestra cómo usar `push`.

## AWS CLI

Para empaquetar e implementar una revisión de aplicación AWS CodeDeploy compatible en Amazon S3

El siguiente push ejemplo agrupa e implementa una revisión de la aplicación en Amazon S3 y, a continuación, asocia la revisión de la aplicación a la aplicación especificada.

```
aws deploy push \  
  --application-name WordPress_App \  
  --description "This is my deployment" \  
  --ignore-hidden-files \  
  --s3-location s3://CodeDeployDemoBucket/WordPressApp.zip \  
  --source /tmp/MyLocalDeploymentFolder/
```

El resultado describe cómo utilizar el `create-deployment` comando para crear una implementación que utilice la revisión de la aplicación cargada.

To deploy with this revision, run:

```
aws deploy create-deployment --application-name WordPress_App  
  --deployment-config-name <deployment-config-name> --  
deployment-group-name <deployment-group-name> --s3-location  
  bucket=CodeDeployDemoBucket,key=WordPressApp.zip,bundleType=zip,eTag="cecc9b8EXAMPLE50a6e71"
```

- Para API obtener más información, consulte la referencia de AWS CLI comandos [Push](#) in.

## register-application-revision

En el siguiente ejemplo de código, se muestra cómo usar `register-application-revision`.

## AWS CLI

Para registrar información sobre una revisión de una aplicación ya cargada

El siguiente `register-application-revision` ejemplo registra información sobre una revisión de aplicación ya cargada almacenada en Amazon S3 con. AWS CodeDeploy

```
aws deploy register-application-revision \  
  --application-name WordPress_App \  
  --description "Revised WordPress application" \  
  --source /tmp/MyLocalDeploymentFolder/
```

```
--s3-
```

```
location bucket=CodeDeployDemoBucket,key=RevisedWordPressApp.zip,bundleType=zip,eTag=cecc9b8
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte la Referencia [RegisterApplicationRevision](#) de AWS CLI comandos.

## register-on-premises-instance

En el siguiente ejemplo de código, se muestra cómo usar `register-on-premises-instance`.

### AWS CLI

Para registrar una instancia local

En el siguiente `register-on-premises-instance` ejemplo, se registra una instancia local con. AWS CodeDeploy No crea el IAM usuario especificado ni asocia en AWS CodeDeploy ninguna instancia local etiquetas a la instancia registrada.

```
aws deploy register-on-premises-instance \  
  --instance-name AssetTag12010298EX \  
  --iam-user-arn arn:aws:iam::80398EXAMPLE:user/CodeDeployDemoUser-OnPrem
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulta [RegisterOnPremisesInstance](#) la Referencia de AWS CLI comandos.

## register

En el siguiente ejemplo de código, se muestra cómo usar `register`.

### AWS CLI

Para registrar una instancia local

En el siguiente `register` ejemplo, se registra una instancia local AWS CodeDeploy, se asocia AWS CodeDeploy la etiqueta de instancia local especificada a la instancia registrada y se crea un archivo de configuración local que se puede copiar en la instancia. No crea el IAM usuario ni instala el AWS CodeDeploy agente en la instancia.

```
aws deploy register \  
  --instance-name AssetTag12010298EX \  
  --iam-user-arn arn:aws:iam::80398EXAMPLE:user/CodeDeployUser-OnPrem \  
  --tags Key=Name,Value=CodeDeployDemo-OnPrem \  
  --region us-west-2
```

Salida:

```
Registering the on-premises instance... DONE  
Adding tags to the on-premises instance... DONE  
Copy the on-premises configuration file named codedeploy.onpremises.yml to the on-  
premises instance, and run the following command on the on-premises instance to  
install and configure the AWS CodeDeploy Agent:  
aws deploy install --config-file codedeploy.onpremises.yml
```

- Para API obtener más información, consulte [Registrar](#) en la referencia de AWS CLI comandos.

## remove-tags-from-on-premises-instances

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags-from-on-premises-instances`.

### AWS CLI

Para eliminar etiquetas de una o más instancias locales

En el siguiente `remove-tags-from-on-premises-instances` ejemplo, se disocian las etiquetas locales especificadas AWS CodeDeploy de las instancias locales. No anula el registro de las instancias locales AWS CodeDeploy, ni desinstala el AWS CodeDeploy agente de la instancia, ni elimina el archivo de configuración local de las instancias, ni elimina los IAM usuarios que están asociados a las instancias.

```
aws deploy remove-tags-from-on-premises-instances \  
  --instance-names AssetTag12010298EX AssetTag23121309EX \  
  --tags Key=Name,Value=CodeDeployDemo-OnPrem
```

Este comando no genera ninguna salida.

- Para obtener API más información, consulte la Referencia de comandos [RemoveTagsFromOnPremisesInstances](#).AWS CLI

## stop-deployment

En el siguiente ejemplo de código, se muestra cómo usar `stop-deployment`.

### AWS CLI

Para intentar detener un despliegue

El siguiente `stop-deployment` ejemplo intenta detener una implementación en curso que está asociada a la AWS cuenta del usuario.

```
aws deploy stop-deployment --deployment-id D-A1B2C3111
```

Salida:

```
{
  "status": "Succeeded",
  "statusMessage": "No more commands will be scheduled for execution in the
deployment instances"
}
```

- Para API obtener más información, consulte la Referencia [StopDeployment](#) de comandos AWS CLI .

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar un recurso (aplicación)

En el siguiente `tag-resource` ejemplo, se agregan dos etiquetas con las claves Nombre y Tipo `testName` y valores `testType` a una aplicación denominada `testApp` in CodeDeploy. :

```
aws deploy tag-resource \
  --resource-arn arn:aws:codedeploy:us-west-2:111122223333:application:testApp \
  --tags Key=Name,Value=testName Key=Type,Value=testType
```

Si se ejecuta correctamente, este comando no genera ninguna salida.



Para obtener más información, consulte [Etiquetar instancias para grupos de implementación CodeDeploy en](#) la Guía del AWS CodeDeploy usuario.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## **uninstall**

En el siguiente ejemplo de código, se muestra cómo usar `uninstall`.

### AWS CLI

Para desinstalar una instancia local

En el siguiente `uninstall` ejemplo, se desinstala el AWS CodeDeploy agente de la instancia local y se elimina el archivo de configuración local de la instancia. No anula el registro de la instancia ni desasocia ninguna etiqueta de instancia local de la instancia, ni AWS CodeDeploy elimina el usuario asociado a la IAM instancia. AWS CodeDeploy

```
aws deploy uninstall
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulta [Desinstalar](#) en la referencia de comandos.AWS CLI

## **untag-resource**

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar etiquetas de un recurso (aplicación)

En el siguiente `untag-resource` ejemplo, se eliminan dos etiquetas con las claves Nombre y Tipo de una aplicación denominada `testApp` in CodeDeploy.

```
aws deploy untag-resource \  
  --resource-arn arn:aws:codedeploy:us-west-2:111122223333:application:testApp \  
  --tag-keys Name Type
```

Si se ejecuta correctamente, este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar instancias para grupos de implementación CodeDeploy en](#) la Guía del AWS CodeDeploy usuario.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-application

En el siguiente ejemplo de código, se muestra cómo usar `update-application`.

### AWS CLI

Para cambiar los detalles de una aplicación

El siguiente `update-application` ejemplo cambia el nombre de una aplicación que está asociada a la AWS cuenta del usuario.

```
aws deploy update-application \  
  --application-name WordPress_App \  
  --new-application-name My_WordPress_App
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UpdateApplication](#) la Referencia de AWS CLI comandos.

## update-deployment-group

En el siguiente ejemplo de código, se muestra cómo usar `update-deployment-group`.

### AWS CLI

Para cambiar la información sobre un grupo de despliegue

En el siguiente `update-deployment-group` ejemplo, se cambia la configuración de un grupo de despliegue que está asociado a la aplicación especificada.

```
aws deploy update-deployment-group \  
  --application-name WordPress_App \  
  --auto-scaling-groups My_CodeDeployDemo_ASG \  
  --deployment-config CodeDeployDefaultAllAtOnce
```

```
--current-deployment-group-name WordPress_DG \  
--deployment-config-name CodeDeployDefault.AllAtOnce \  
--ec2-tag-filters Key=Name, Type=KEY_AND_VALUE, Value=My_CodeDeployDemo \  
--new-deployment-group-name My_WordPress_DepGroup \  
--service-role-arn arn:aws:iam::80398EXAMPLE:role/CodeDeployDemo-2
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UpdateDeploymentGroup](#) la Referencia de AWS CLI comandos.

## CodeGuru Ejemplos de revisores que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface CodeGuru Reviewer.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **associate-repository**

En el siguiente ejemplo de código, se muestra cómo usar `associate-repository`.

AWS CLI

Ejemplo 1: Para crear una asociación de repositorios de Bitbucket

En el siguiente `associate-repository` ejemplo, se crea una asociación de repositorios utilizando un repositorio de Bitbucket existente.

```
aws codeguru-reviewer associate-repository \  

```

```
--repository 'Bitbucket={Owner=sample-owner, Name=mySampleRepo,
ConnectionArn=arn:aws:codestar-connections:us-west-2:123456789012:connection/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 }'
```

Salida:

```
{
  "RepositoryAssociation": {
    "ProviderType": "Bitbucket",
    "Name": "mySampleRepo",
    "LastUpdatedTimeStamp": 1596216896.979,
    "AssociationId": "association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "CreatedTimeStamp": 1596216896.979,
    "ConnectionArn": "arn:aws:codestar-connections:us-
west-2:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "State": "Associating",
    "StateReason": "Pending Repository Association",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "Owner": "sample-owner"
  }
}
```

Para obtener más información, consulte [Crear una asociación de repositorios de Bitbucket en Amazon CodeGuru Reviewer](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

Ejemplo 2: Para crear una asociación de repositorios GitHub empresariales

El siguiente `associate-repository` ejemplo crea una asociación de repositorios utilizando un repositorio GitHub empresarial existente.

```
aws codeguru-reviewer associate-repository \
--repository 'GitHubEnterpriseServer={Owner=sample-owner, Name=mySampleRepo,
ConnectionArn=arn:aws:codestar-connections:us-west-2:123456789012:connection/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 }'
```

Salida:

```
{
  "RepositoryAssociation": {
    "ProviderType": "GitHubEnterpriseServer",
```

```

    "Name": "mySampleRepo",
    "LastUpdatedTimeStamp": 1596216896.979,
    "AssociationId": "association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "CreatedTimeStamp": 1596216896.979,
    "ConnectionArn": "arn:aws:codestar-connections:us-
west-2:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "State": "Associating",
    "StateReason": "Pending Repository Association",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "Owner": "sample-owner"
  }
}

```

Para obtener más información, consulte [Crear una asociación de repositorios de GitHub Enterprise Server en Amazon CodeGuru Reviewer](#) en la Guía del usuario de Amazon Codeguru Reviewer.

Ejemplo 3: Para crear una asociación de AWS CodeCommit repositorios

El siguiente `associate-repository` ejemplo crea una asociación de repositorios utilizando un AWS CodeCommit repositorio existente.

```

aws codeguru-reviewer associate-repository \
  --repository CodeCommit={Name=mySampleRepo}

```

Salida:

```

{
  "RepositoryAssociation": {
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Name": "My-ecs-beta-repo",
    "LastUpdatedTimeStamp": 1595634764.029,
    "ProviderType": "CodeCommit",
    "CreatedTimeStamp": 1595634764.029,
    "Owner": "544120495673",
    "State": "Associating",
    "StateReason": "Pending Repository Association",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:544120495673:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}

```

Para obtener más información, consulte [Crear una asociación de AWS CodeCommit repositorios en Amazon CodeGuru Reviewer](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [AssociateRepository](#) la Referencia de AWS CLI comandos.

## create-code-review

En el siguiente ejemplo de código, se muestra cómo usar `create-code-review`.

### AWS CLI

Para crear una revisión de código.

A continuación, se `create-code-review` crea una revisión del código en la `mainline` rama de un AWS CodeCommit repositorio que recibe el nombre `my-repository-name`.

```
aws codeguru-reviewer create-code-review \
  --name my-code-review \
  --repository-association-arn arn:aws:codeguru-reviewer:us-west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --type '{"RepositoryAnalysis": {"RepositoryHead": {"BranchName": "mainline"}}}'
```

Salida:

```
{
  "CodeReview": {
    "Name": "my-code-review",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222:code-review:RepositoryAnalysis-my-code-review",
    "RepositoryName": "my-repository-name",
    "Owner": "123456789012",
    "ProviderType": "CodeCommit",
    "State": "Pending",
    "StateReason": "CodeGuru Reviewer has received the request, and a code review is scheduled.",
    "CreatedTimeStamp": 1618873489.195,
    "LastUpdatedTimeStamp": 1618873489.195,
    "Type": "RepositoryAnalysis",
    "SourceCodeType": {
      "RepositoryHead": {
```

```

        "BranchName": "mainline"
    }
},
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
}

```

Para obtener más información, consulte [Crear reseñas de código en Amazon CodeGuru Reviewer](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [CreateCodeReview](#) la Referencia de AWS CLI comandos.

## describe-code-review

En el siguiente ejemplo de código, se muestra cómo usar `describe-code-review`.

### AWS CLI

Enumere los detalles sobre una revisión de código.

A continuación, se `describe-code-review` muestra información sobre una revisión del código en la rama «principal» de un AWS CodeCommit repositorio denominado "my-repo-name».

```

aws codeguru-reviewer put-recommendation-feedback \
  --code-review-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678 \
  --recommendation-
id 3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb \
  --reactions ThumbsUp

```

### Salida

```

{
  "CodeReview": {
    "Name": "My-ecs-beta-repo-master-xs6di4kfd4j269dz",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222:code-
review:RepositoryAnalysis-my-repo-name",

```

```

    "RepositoryName": "My-ecs-beta-repo",
    "Owner": "123456789012",
    "ProviderType": "CodeCommit",
    "State": "Pending",
    "StateReason": "CodeGuru Reviewer is reviewing the source code.",
    "CreatedTimeStamp": 1618874226.226,
    "LastUpdatedTimeStamp": 1618874233.689,
    "Type": "RepositoryAnalysis",
    "SourceCodeType": {
      "RepositoryHead": {
        "BranchName": "mainline"
      }
    },
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}

```

Para obtener más información, consulte [Ver detalles de revisión de código](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [DescribeCodeReview](#) la Referencia de AWS CLI comandos.

## describe-recommendation-feedback

En el siguiente ejemplo de código, se muestra cómo usar `describe-recommendation-feedback`.

### AWS CLI

Para ver información sobre los comentarios sobre una recomendación

A continuación, `describe-recommendation-feedback` se muestra información sobre los comentarios sobre una recomendación. Esta recomendación tiene una ThumbsUp reacción.

```

aws codeguru-reviewer describe-recommendation-feedback \
  --code-review-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678 \
  --recommendation-
id 3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb

```



Salida:

```
{
  "RecommendationFeedback": {
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678",
    "RecommendationId":
"3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb",
    "Reactions": [
      "ThumbsUp"
    ],
    "UserId": "aws-user-id",
    "CreatedTimeStamp": 1618877070.313,
    "LastUpdatedTimeStamp": 1618877948.881
  }
}
```

Para obtener más información, consulte [Ver recomendaciones y enviar comentarios](#) y [Paso 4: Enviar comentarios](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [DescribeRecommendationFeedback](#) la Referencia de AWS CLI comandos.

## describe-repository-association

En el siguiente ejemplo de código, se muestra cómo usar `describe-repository-association`.

AWS CLI

Ejemplo 1: Para devolver información sobre una asociación de GitHub repositorios

En el siguiente `describe-repository-association` ejemplo, se devuelve información sobre una asociación de repositorios que utiliza un repositorio GitHub empresarial y se encuentra en ese `Associated` estado.

```
aws codeguru-reviewer describe-repository-association \
  --association-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{
  "RepositoryAssociation": {
    "AssociationId": "b822717e-0711-4e8a-bada-0e738289c75e",
    "Name": "mySampleRepo",
    "LastUpdatedTimeStamp": 1588102637.649,
    "ProviderType": "GitHub",
    "CreatedTimeStamp": 1588102615.636,
    "Owner": "sample-owner",
    "State": "Associated",
    "StateReason": "Pull Request Notification configuration successful",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}
```

Para obtener más información, consulte [Crear una asociación de repositorios de GitHub Enterprise Server en Amazon CodeGuru Reviewer](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

Ejemplo 2: Para devolver información sobre una asociación de repositorios fallida

El siguiente describe-repository-association ejemplo devuelve información sobre una asociación de repositorios que utiliza un repositorio GitHub empresarial y se encuentra en ese Failed estado.

```
aws codeguru-reviewer describe-repository-association \
  --association-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{
  "RepositoryAssociation": {
    "ProviderType": "GitHubEnterpriseServer",
    "Name": "mySampleRepo",
    "LastUpdatedTimeStamp": 1596217036.892,
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "CreatedTimeStamp": 1596216896.979,
    "ConnectionArn": "arn:aws:codestar-connections:us-
west-2:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "State": "Failed",
    "StateReason": "Failed, Please retry.",
  }
}
```

```

    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "Owner": "sample-owner"
  }
}

```

Para obtener más información, consulte [Crear una asociación de repositorios de GitHub Enterprise Server en Amazon CodeGuru Reviewer](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

Ejemplo 3: Para devolver información sobre una asociación de repositorios disociada

El siguiente describe-repository-association ejemplo devuelve información sobre una asociación de repositorios que utiliza un repositorio GitHub empresarial y se encuentra en ese estado. Disassociating

```

aws codeguru-reviewer describe-repository-association \
  --association-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

```

Salida:

```

{
  "RepositoryAssociation": {
    "ProviderType": "GitHubEnterpriseServer",
    "Name": "mySampleRepo",
    "LastUpdatedTimeStamp": 1596217036.892,
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "CreatedTimeStamp": 1596216896.979,
    "ConnectionArn": "arn:aws:codestar-connections:us-
west-2:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "State": "Disassociating",
    "StateReason": "Source code access removal in progress",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "Owner": "sample-owner"
  }
}

```

Para obtener más información, consulte [Crear una asociación de repositorios de GitHub Enterprise Server en Amazon CodeGuru Reviewer](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [DescribeRepositoryAssociation](#) la Referencia de AWS CLI comandos.

## disassociate-repository

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-repository`.

### AWS CLI

Para desasociar una asociación de repositorios

A continuación, `disassociate-repository` se desasocia una asociación de repositorios que utiliza un AWS CodeCommit repositorio.

```
aws codeguru-reviewer disassociate-repository \  
  --association-arn arn:aws:codeguru-reviewer:us-  
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "RepositoryAssociation": {  
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "AssociationArn": "arn:aws:codeguru-reviewer:us-  
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "Name": "my-repository",  
    "Owner": "123456789012",  
    "ProviderType": "CodeCommit",  
    "State": "Disassociating",  
    "LastUpdatedTimeStamp": 1618939174.759,  
    "CreatedTimeStamp": 1595636947.096  
  },  
  "Tags": {  
    "Status": "Secret",  
    "Team": "Saanvi"  
  }  
}
```

Para obtener más información, consulte [Desasociar un repositorio en CodeGuru Reviewer en la Guía del usuario de Amazon CodeGuru Reviewer](#).

- Para API obtener más información, consulte [DisassociateRepository](#) la Referencia de AWS CLI comandos.

## list-code-reviews

En el siguiente ejemplo de código, se muestra cómo usar `list-code-reviews`.

### AWS CLI

Para ver una lista de las revisiones de código creadas en tu AWS cuenta en los últimos 90 días.

En el siguiente `list-code-reviews` ejemplo, se enumeran las revisiones de código creadas en los últimos 90 días mediante solicitudes de extracción de cambios.

```
aws codeguru-reviewer list-code-reviews \  
  --type PullRequest
```

Salida:

```
{  
  "CodeReviewSummaries": [  
    {  
      "LastUpdatedTimeStamp": 1588897288.054,  
      "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "ProviderType": "GitHub",  
      "PullRequestId": "5",  
      "MetricsSummary": {  
        "MeteredLinesOfCodeCount": 24,  
        "FindingsCount": 1  
      },  
      "CreatedTimeStamp": 1588897068.512,  
      "State": "Completed",  
      "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-  
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "Owner": "sample-owner",  
      "RepositoryName": "sample-repository-name",  
      "Type": "PullRequest"  
    },  
    {  
      "LastUpdatedTimeStamp": 1588869793.263,  
      "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "ProviderType": "GitHub",
```

```
    "PullRequestId": "4",
    "MetricsSummary": {
      "MeteredLinesOfCodeCount": 29,
      "FindingsCount": 0
    },
    "CreatedTimeStamp": 1588869575.949,
    "State": "Completed",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "Owner": "sample-owner",
    "RepositoryName": "sample-repository-name",
    "Type": "PullRequest"
  },
  {
    "LastUpdatedTimeStamp": 1588870511.211,
    "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "ProviderType": "GitHub",
    "PullRequestId": "4",
    "MetricsSummary": {
      "MeteredLinesOfCodeCount": 2,
      "FindingsCount": 0
    },
    "CreatedTimeStamp": 1588870292.425,
    "State": "Completed",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "Owner": "sample-owner",
    "RepositoryName": "sample-repository-name",
    "Type": "PullRequest"
  },
  {
    "LastUpdatedTimeStamp": 1588118522.452,
    "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
    "ProviderType": "GitHub",
    "PullRequestId": "3",
    "MetricsSummary": {
      "MeteredLinesOfCodeCount": 29,
      "FindingsCount": 0
    },
    "CreatedTimeStamp": 1588118301.131,
    "State": "Completed",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
    "Owner": "sample-owner",
```

```

    "RepositoryName": "sample-repository-name",
    "Type": "PullRequest"
  },
  {
    "LastUpdatedTimeStamp": 1588112205.207,
    "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE55555",
    "ProviderType": "GitHub",
    "PullRequestId": "2",
    "MetricsSummary": {
      "MeteredLinesOfCodeCount": 25,
      "FindingsCount": 0
    },
    "CreatedTimeStamp": 1588111987.443,
    "State": "Completed",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE55555",
    "Owner": "sample-owner",
    "RepositoryName": "sample-repository-name",
    "Type": "PullRequest"
  },
  {
    "LastUpdatedTimeStamp": 1588104489.981,
    "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE66666",
    "ProviderType": "GitHub",
    "PullRequestId": "1",
    "MetricsSummary": {
      "MeteredLinesOfCodeCount": 25,
      "FindingsCount": 0
    },
    "CreatedTimeStamp": 1588104270.223,
    "State": "Completed",
    "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE66666",
    "Owner": "sample-owner",
    "RepositoryName": "sample-repository-name",
    "Type": "PullRequest"
  }
]
}

```

Para obtener más información, consulte [Ver todas las reseñas de código](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [ListCodeReviews](#) la Referencia de AWS CLI comandos.

## **list-recommendation-feedback**

En el siguiente ejemplo de código, se muestra cómo usar `list-recommendation-feedback`.

### AWS CLI

Para enumerar los comentarios de los clientes sobre una recomendación en un repositorio asociado

A continuación se `list-recommendation-feedback` enumeran los comentarios de los clientes sobre todas las recomendaciones de una revisión de código. Esta revisión de código contiene un comentario, un ThumbsUp «», de un cliente.

```
aws codeguru-reviewer list-recommendation-feedback \
  --code-review-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678
```

Salida:

```
{
  "RecommendationFeedbackSummaries": [
    {
      "RecommendationId":
"3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb",
      "Reactions": [
        "ThumbsUp"
      ],
      "UserId": "aws-user-id"
    }
  ]
}
```

Para obtener más información, consulte el [paso 4: Enviar comentarios](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [ListRecommendationFeedback](#) la Referencia de AWS CLI comandos.



## list-recommendations

En el siguiente ejemplo de código, se muestra cómo usar `list-recommendations`.

### AWS CLI

Para enumerar las recomendaciones para una revisión completa del código

En el siguiente `list-recommendations` ejemplo, se enumeran las recomendaciones para completar la revisión del código. Esta revisión del código tiene una recomendación.

```
aws codeguru-reviewer list-recommendations \  
  --code-review-arn arn:aws:codeguru-reviewer:us-west-2:544120495673:code-  
  review:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "RecommendationSummaries": [  
    {  
      "Description": "\n\nProblem  \n You are using a `ConcurrentHashMap`,  
but your usage of `containsKey()` and `get()` may not be thread-safe at lines: **63  
and 64**. In between the check and the `get()` another thread can remove the key  
and the `get()` will return `null`. The remove that can remove the key is at line:  
**59**.\n\nFix  \n Consider calling `get()`, checking instead of your current  
check if the returned object is `null`, and then using that object only, without  
calling `get()` again.\n\nMore info  \n [View an example on GitHub](https://  
github.com/apache/hadoop/blob/f16cf877e565084c66bc63605659b157c4394dc8/hadoop-tools/  
hadoop-aws/src/main/java/org/apache/hadoop/fs/s3a/s3guard/S3Guard.java#L302-L304)  
(external link).",  
      "RecommendationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "StartLine": 63,  
      "EndLine": 64,  
      "FilePath": "src/main/java/com/company/sample/application/  
CreateOrderThread.java"  
    }  
  ]  
}
```

Para obtener más información, consulte el [paso 4: Enviar comentarios](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [ListRecommendations](#) la Referencia de AWS CLI comandos.

## list-repository-associations

En el siguiente ejemplo de código, se muestra cómo usar `list-repository-associations`.

### AWS CLI

Para ver una lista de las asociaciones de repositorios de su AWS cuenta

En el siguiente `list-repository-associations` ejemplo, se devuelve una lista de los objetos resumidos de las asociaciones de repositorios de tu cuenta. Puede filtrar la lista devuelta por `ProviderTypeName`, `State`, y `Owner`.

```
aws codeguru-reviewer list-repository-associations
```

Salida:

```
{
  "RepositoryAssociationSummaries": [
    {
      "LastUpdatedTimeStamp": 1595886609.616,
      "Name": "test",
      "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Owner": "sample-owner",
      "State": "Associated",
      "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ProviderType": "Bitbucket"
    },
    {
      "LastUpdatedTimeStamp": 1595636969.035,
      "Name": "CodeDeploy-CodePipeline-ECS-Tutorial",
      "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "Owner": "123456789012",
      "State": "Associated",
      "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "ProviderType": "CodeCommit"
    },
    {
```

```
    "LastUpdatedTimeStamp": 1595634785.983,
    "Name": "My-ecs-beta-repo",
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "Owner": "123456789012",
    "State": "Associated",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "ProviderType": "CodeCommit"
  },
  {
    "LastUpdatedTimeStamp": 1590712811.77,
    "Name": "MyTestCodeCommit",
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
    "Owner": "123456789012",
    "State": "Associated",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
    "ProviderType": "CodeCommit"
  },
  {
    "LastUpdatedTimeStamp": 1588102637.649,
    "Name": "aws-codeguru-profiler-sample-application",
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE55555",
    "Owner": "sample-owner",
    "State": "Associated",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE55555",
    "ProviderType": "GitHub"
  },
  {
    "LastUpdatedTimeStamp": 1588028233.995,
    "Name": "codeguru-profiler-demo-app",
    "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE66666",
    "Owner": "sample-owner",
    "State": "Associated",
    "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE66666",
    "ProviderType": "GitHub"
  }
]
}
```

Para obtener más información, consulte [Ver todas las asociaciones de repositorios en CodeGuru Reviewer](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [ListRepositoryAssociations](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un repositorio asociado

A continuación, se `list-tags-for-resource` enumeran las etiquetas de un repositorio asociado. Este repositorio asociado tiene dos etiquetas.

```
aws codeguru-reviewer list-tags-for-resource \
  --resource-arn arn:aws:codeguru-reviewer:us-
  west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{
  "Tags": {
    "Status": "Secret",
    "Team": "Saanvi"
  }
}
```

Para obtener más información, consulte [Ver etiquetas de un repositorio asociado a CodeGuru Reviewer \(AWS CLI\)](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## put-recommendation-feedback

En el siguiente ejemplo de código, se muestra cómo usar `put-recommendation-feedback`.

## AWS CLI

Para añadir una recomendación a una revisión de código

A continuación, `put-recommendation-feedback` se incluye una `ThumbsUp` recomendación sobre una revisión de código.

```
aws codeguru-reviewer put-recommendation-feedback \  
  --code-review-arn \arn:aws:codeguru-reviewer:us-west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-review:RepositoryAnalysis-my-repository-name-branch-abcdefg12345678 \  
  --recommendation-id 3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb \  
  --reactions ThumbsUp
```

Este comando no genera ninguna salida.

Para obtener más información, consulte el [paso 4: Enviar comentarios](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [PutRecommendationFeedback](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

## AWS CLI

Para añadir una etiqueta a un repositorio asociado

A continuación, `tag-resource` se añaden dos etiquetas a un repositorio asociado

```
aws codeguru-reviewer tag-resource \  
  --resource-arn arn:aws:codeguru-reviewer:us-west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --tags Status=Secret,Team=Saarvi
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir una etiqueta a un repositorio asociado a CodeGuru Reviewer \(AWS CLI\)](#) y [Agregar o actualizar etiquetas para un repositorio asociado a CodeGuru Reviewer \(AWS CLI\)](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para quitar la etiqueta de un repositorio asociado

Lo siguiente `untag-resource` elimina dos etiquetas con las claves «Secreto» y «Equipo» de un repositorio asociado.

```
aws codeguru-reviewer untag-resource \  
  --resource-arn arn:aws:codeguru-reviewer:us-  
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --tag-keys Status Team
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar etiquetas de un repositorio asociado a CodeGuru Reviewer \(AWS CLI\)](#) en la Guía del usuario de Amazon CodeGuru Reviewer.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## CodePipeline ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with CodePipeline.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **acknowledge-job**

En el siguiente ejemplo de código, se muestra cómo usar `acknowledge-job`.

#### AWS CLI

Para recuperar información sobre un trabajo específico

En este ejemplo se devuelve información sobre un trabajo específico, incluido el estado de ese trabajo, si existe. Esto solo se usa para los trabajadores del trabajo y las acciones personalizadas. Para determinar el valor de `nonce` y el identificador del trabajo, utilice `aws poll-for-jobs codepipeline`.

Comando:

```
aws codepipeline acknowledge-job --job-id f4f4ff82-2d11-EXAMPLE --nonce 3
```

Salida:

```
{  
  "status": "InProgress"  
}
```

- Para API obtener más información, consulte la Referencia [AcknowledgeJob](#) de AWS CLI comandos.

### **create-custom-action-type**

En el siguiente ejemplo de código, se muestra cómo usar `create-custom-action-type`.

#### AWS CLI

Para crear una acción personalizada

En este ejemplo, se crea una acción personalizada para AWS CodePipeline utilizar un JSON archivo ya creado (aquí denominado `MyCustomAction.json`) que contiene la estructura de la

acción personalizada. Para obtener más información sobre los requisitos para crear una acción personalizada, incluida la estructura del archivo, consulte la Guía del AWS CodePipeline usuario.

```
aws codepipeline create-custom-action-type --cli-input-json file://  
MyCustomAction.json
```

Contenido del JSON archivo `MyCustomAction.json`:

```
{  
  "category": "Build",  
  "provider": "MyJenkinsProviderName",  
  "version": "1",  
  "settings": {  
    "entityUrlTemplate": "https://192.0.2.4/job/{Config:ProjectName}/",  
    "executionUrlTemplate": "https://192.0.2.4/job/{Config:ProjectName}/  
lastSuccessfulBuild/{ExternalExecutionId}/"  
  },  
  "configurationProperties": [  
    {  
      "name": "MyJenkinsExampleBuildProject",  
      "required": true,  
      "key": true,  
      "secret": false,  
      "queryable": false,  
      "description": "The name of the build project must be provided when this  
action is added to the pipeline.",  
      "type": "String"  
    }  
  ],  
  "inputArtifactDetails": {  
    "maximumCount": 1,  
    "minimumCount": 0  
  },  
  "outputArtifactDetails": {  
    "maximumCount": 1,  
    "minimumCount": 0  
  }  
}
```

Este comando devuelve la estructura de la acción personalizada.

- Para API obtener más información, consulte [CreateCustomActionType](#) la Referencia de AWS CLI comandos.



## create-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `create-pipeline`.

### AWS CLI

Para crear una canalización

En este ejemplo, se crea una canalización AWS CodePipeline utilizando un JSON archivo ya creado (aquí denominado `MySecondPipeline.json`) que contiene la estructura de la canalización. Para obtener más información sobre los requisitos para crear una canalización, incluida la estructura del archivo, consulta la Guía del AWS CodePipeline usuario.

Comando:

```
aws codepipeline create-pipeline --cli-input-json file://MySecondPipeline.json
```

JSON contenido de muestra del archivo:

```
{
  "pipeline": {
    "roleArn": "arn:aws:iam::111111111111:role/AWS-CodePipeline-Service",
    "stages": [
      {
        "name": "Source",
        "actions": [
          {
            "inputArtifacts": [],
            "name": "Source",
            "actionTypeId": {
              "category": "Source",
              "owner": "AWS",
              "version": "1",
              "provider": "S3"
            },
            "outputArtifacts": [
              {
                "name": "MyApp"
              }
            ],
            "configuration": {
              "S3Bucket": "awscodepipeline-demo-bucket",
              "S3ObjectKey": "aws-codepipeline-s3-aws-codedeploy_linux.zip"
            }
          }
        ]
      }
    ]
  }
}
```

```
    },
    "runOrder": 1
  }
]
},
{
  "name": "Beta",
  "actions": [
    {
      "inputArtifacts": [
        {
          "name": "MyApp"
        }
      ],
      "name": "CodePipelineDemoFleet",
      "actionTypeId": {
        "category": "Deploy",
        "owner": "AWS",
        "version": "1",
        "provider": "CodeDeploy"
      },
      "outputArtifacts": [],
      "configuration": {
        "ApplicationName": "CodePipelineDemoApplication",
        "DeploymentGroupName": "CodePipelineDemoFleet"
      },
      "runOrder": 1
    }
  ]
}
],
"artifactStore": {
  "type": "S3",
  "location": "codepipeline-us-east-1-11EXAMPLE11"
},
"name": "MySecondPipeline",
"version": 1
}
}
```

**Salida:**

This command returns the structure of the pipeline.

- Para API obtener más información, consulte [CreatePipeline](#) la Referencia de AWS CLI comandos.

## delete-custom-action-type

En el siguiente ejemplo de código, se muestra cómo usar `delete-custom-action-type`.

### AWS CLI

Para eliminar una acción personalizada

En este ejemplo, se elimina una acción personalizada AWS CodePipeline mediante un JSON archivo ya creado (aquí denominado `DeleteMyCustomAction.json`) que contiene el tipo de acción, el nombre del proveedor y el número de versión de la acción que se va a eliminar. Utilice el `list-action-types` comando para ver los valores correctos de la categoría, la versión y el proveedor.

Comando:

```
aws codepipeline delete-custom-action-type --cli-input-json file://DeleteMyCustomAction.json
```

JSON contenido de ejemplo de archivo:

```
{
  "category": "Build",
  "version": "1",
  "provider": "MyJenkinsProviderName"
}
```

Salida:

```
None.
```

- Para API obtener más información, consulte [DeleteCustomActionType](#) la Referencia de AWS CLI comandos.

## delete-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `delete-pipeline`.

## AWS CLI

Para eliminar una canalización

En este ejemplo, se elimina una canalización cuyo nombre `MySecondPipeline` es. AWS CodePipeline Usa el comando `list-pipelines` para ver una lista de las canalizaciones asociadas a tu cuenta. AWS

Comando:

```
aws codepipeline delete-pipeline --name MySecondPipeline
```

Salida:

```
None.
```

- Para obtener API más información, consulta la Referencia de comandos [DeletePipeline](#).AWS CLI

## **delete-webhook**

En el siguiente ejemplo de código, se muestra cómo usar `delete-webhook`.

## AWS CLI

Para eliminar un webhook

En el siguiente `delete-webhook` ejemplo, se elimina un webhook para una acción fuente de la GitHub versión 1. Debes usar el `deregister-webhook-with-third-party` comando para anular el registro del webhook antes de eliminarlo.

```
aws codepipeline delete-webhook \  
  --name my-webhook
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar el webhook de la GitHub fuente en la Guía del usuario](#).AWS CodePipeline

- Para API obtener más información, consulte [DeleteWebhook](#) la Referencia de AWS CLI comandos.

## deregister-webhook-with-third-party

En el siguiente ejemplo de código, se muestra cómo usar `deregister-webhook-with-third-party`.

### AWS CLI

Para anular el registro de un webhook

En el siguiente `deregister-webhook-with-third-party` ejemplo, se elimina un webhook para una GitHub acción fuente de la versión 1. Debe anular el registro del webhook antes de eliminarlo.

```
aws codepipeline deregister-webhook-with-third-party \  
  --webhook-name my-webhook
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar el webhook de la GitHub fuente](#) en la Guía del AWS CodePipeline usuario.

- Para API obtener más información, consulte [DeregisterWebhookWithThirdParty](#) la Referencia de AWS CLI comandos.

## disable-stage-transition

En el siguiente ejemplo de código, se muestra cómo usar `disable-stage-transition`.

### AWS CLI

Para deshabilitar la transición a una etapa de una canalización

En este ejemplo, se deshabilitan las transiciones a la fase beta de la `MyFirstPipeline` canalización en AWS CodePipeline.

Comando:

```
aws codepipeline disable-stage-transition --pipeline-name MyFirstPipeline --stage-  
name Beta --transition-type Inbound
```

Salida:

```
None .
```

- Para API obtener más información, consulte [DisableStageTransition](#) la Referencia de AWS CLI comandos.

## **enable-stage-transition**

En el siguiente ejemplo de código, se muestra cómo usar `enable-stage-transition`.

### AWS CLI

Para permitir la transición a una etapa de una canalización

En este ejemplo, se permiten las transiciones a la fase beta de la `MyFirstPipeline` canalización en AWS CodePipeline.

Comando:

```
aws codepipeline enable-stage-transition --pipeline-name MyFirstPipeline --stage-name Beta --transition-type Inbound
```

Salida:

```
None .
```

- Para API obtener más información, consulte [EnableStageTransition](#) la Referencia de AWS CLI comandos.

## **get-job-details**

En el siguiente ejemplo de código, se muestra cómo usar `get-job-details`.

### AWS CLI

Para obtener detalles de un trabajo

Este ejemplo devuelve detalles sobre un trabajo cuyo identificador está representado por `f4f4ff82-2d11-`. **EXAMPLE** Este comando solo se usa para acciones personalizadas. Cuando se

llama a este comando, AWS CodePipeline devuelve las credenciales temporales del bucket de Amazon S3 que se utiliza para almacenar los artefactos de la canalización, si es necesario para la acción personalizada. Este comando también devolverá cualquier valor secreto definido para la acción, si se ha definido alguno.

Comando:

```
aws codepipeline get-job-details --job-id f4f4ff82-2d11-EXAMPLE
```

Salida:

```
{
  "jobDetails": {
    "accountId": "111111111111",
    "data": {
      "actionConfiguration": {
        "__type": "ActionConfiguration",
        "configuration": {
          "ProjectName": "MyJenkinsExampleTestProject"
        }
      },
      "actionTypeId": {
        "__type": "ActionTypeId",
        "category": "Test",
        "owner": "Custom",
        "provider": "MyJenkinsProviderName",
        "version": "1"
      },
      "artifactCredentials": {
        "__type": "AWSSessionCredentials",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "secretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY",
        "sessionToken":
          "fICCQD6m7oRw0uX0jANBqkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwQw
          +a4GmWIWJ21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/
          f0wYK8m9TrDHudUZg3qX4waLG5M43q7Wgc/
          MbQITx0USQv7c7ugFFDzQGBzSzwY6786m86gpEIbb30hjZnzcVQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQ
          +auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0FkbFFBjvSfpJI1J0zdbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs
      },
      "inputArtifacts": [
        {
          "__type": "Artifact",
          "location": {
```

```
    "s3Location": {
      "bucketName": "codepipeline-us-east-1-11EXAMPLE11",
      "objectKey": "MySecondPipeline/MyAppBuild/EXAMPLE"
    },
    "type": "S3"
  },
  "name": "MyAppBuild"
}
],
"outputArtifacts": [],
"pipelineContext": {
  "__type": "PipelineContext",
  "action": {
    "name": "MyJenkinsTest-Action"
  },
  "pipelineName": "MySecondPipeline",
  "stage": {
    "name": "Testing"
  }
}
},
"id": "f4f4ff82-2d11-EXAMPLE"
}
}
```

- Para API obtener más información, consulte [GetJobDetails](#) la Referencia de AWS CLI comandos.

## get-pipeline-state

En el siguiente ejemplo de código, se muestra cómo usar `get-pipeline-state`.

### AWS CLI

Para obtener información sobre el estado de una tubería

En este ejemplo se devuelve el estado más reciente de una canalización denominada `MyFirstPipeline`.

Comando:

```
aws codepipeline get-pipeline-state --name MyFirstPipeline
```



**Salida:**

```
{
  "created": 1446137312.204,
  "pipelineName": "MyFirstPipeline",
  "pipelineVersion": 1,
  "stageStates": [
    {
      "actionStates": [
        {
          "actionName": "Source",
          "entityUrl": "https://console.aws.amazon.com/s3/home?#",
          "latestExecution": {
            "lastStatusChange": 1446137358.328,
            "status": "Succeeded"
          }
        }
      ],
      "stageName": "Source"
    },
    {
      "actionStates": [
        {
          "actionName": "CodePipelineDemoFleet",
          "entityUrl": "https://console.aws.amazon.com/codedeploy/home?#/applications/CodePipelineDemoApplication/deployment-groups/CodePipelineDemoFleet",
          "latestExecution": {
            "externalExecutionId": "d-EXAMPLE",
            "externalExecutionUrl": "https://console.aws.amazon.com/codedeploy/home?#/deployments/d-EXAMPLE",
            "lastStatusChange": 1446137493.131,
            "status": "Succeeded",
            "summary": "Deployment Succeeded"
          }
        }
      ],
      "inboundTransitionState": {
        "enabled": true
      },
      "stageName": "Beta"
    }
  ],
  "updated": 1446137312.204
}
```

```
}
```

- Para API obtener más información, consulte [GetPipelineState](#) la Referencia de AWS CLI comandos.

## get-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `get-pipeline`.

### AWS CLI

Para ver la estructura de una tubería

En este ejemplo se devuelve la estructura de una canalización denominada `MyFirstPipeline`.

Comando:

```
aws codepipeline get-pipeline --name MyFirstPipeline
```

Salida:

```
{
  "pipeline": {
    "roleArn": "arn:aws:iam::111111111111:role/AWS-CodePipeline-Service",
    "stages": [
      {
        "name": "Source",
        "actions": [
          {
            "inputArtifacts": [],
            "name": "Source",
            "actionTypeId": {
              "category": "Source",
              "owner": "AWS",
              "version": "1",
              "provider": "S3"
            },
            "outputArtifacts": [
              {
                "name": "MyApp"
              }
            ],
            "configuration": {
```

```

        "S3Bucket": "awscodepipeline-demo-bucket",
        "S3ObjectKey": "aws-codepipeline-s3-aws-
codedeploy_linux.zip"
    },
    "runOrder": 1
  }
]
},
{
  "name": "Beta",
  "actions": [
    {
      "inputArtifacts": [
        {
          "name": "MyApp"
        }
      ],
      "name": "CodePipelineDemoFleet",
      "actionTypeId": {
        "category": "Deploy",
        "owner": "AWS",
        "version": "1",
        "provider": "CodeDeploy"
      },
      "outputArtifacts": [],
      "configuration": {
        "ApplicationName": "CodePipelineDemoApplication",
        "DeploymentGroupName": "CodePipelineDemoFleet"
      },
      "runOrder": 1
    }
  ]
}
],
"artifactStore": {
  "type": "S3",
  "location": "codepipeline-us-east-1-11EXAMPLE11"
},
"name": "MyFirstPipeline",
"version": 1
}
}

```

- Para API obtener más información, consulte [GetPipeline](#) la Referencia de AWS CLI comandos.

## list-action-executions

En el siguiente ejemplo de código, se muestra cómo usar `list-action-executions`.

### AWS CLI

Para ver una lista de las ejecuciones de acciones

En el siguiente `list-action-executions` ejemplo, se muestran los detalles de la ejecución de acciones de una canalización, como el identificador de ejecución de la acción, los artefactos de entrada, los artefactos de salida, el resultado de la ejecución y el estado.

```
aws codepipeline list-action-executions \  
  --pipeline-name myPipeline
```

Salida:

```
{  
  "actionExecutionDetails": [  
    {  
      "pipelineExecutionId": "EXAMPLE0-adfc-488e-bf4c-1111111720d3",  
      "actionExecutionId": "EXAMPLE4-2ee8-4853-bd6a-111111158148",  
      "pipelineVersion": 12,  
      "stageName": "Deploy",  
      "actionName": "Deploy",  
      "startTime": 1598572628.6,  
      "lastUpdateTime": 1598572661.255,  
      "status": "Succeeded",  
      "input": {  
        "actionTypeId": {  
          "category": "Deploy",  
          "owner": "AWS",  
          "provider": "CodeDeploy",  
          "version": "1"  
        },  
        "configuration": {  
          "ApplicationName": "my-application",  
          "DeploymentGroupName": "my-deployment-group"  
        },  
        "resolvedConfiguration": {  
          "ApplicationName": "my-application",  
          "DeploymentGroupName": "my-deployment-group"  
        },  
      },  
    ],  
  ],  
}
```

```
    "region": "us-east-1",
    "inputArtifacts": [
      {
        "name": "SourceArtifact",
        "s3location": {
          "bucket": "artifact-bucket",
          "key": "myPipeline/SourceArti/key"
        }
      }
    ],
    "namespace": "DeployVariables"
  },
  "output": {
    "outputArtifacts": [],
    "executionResult": {
      "externalExecutionId": "d-EXAMPLEE5",
      "externalExecutionSummary": "Deployment Succeeded",
      "externalExecutionUrl": "https://myaddress.com"
    },
    "outputVariables": {}
  }
},
{
  "pipelineExecutionId": "EXAMPLE0-adfc-488e-bf4c-1111111720d3",
  "actionExecutionId": "EXAMPLE5-abb4-4192-9031-11111113a7b0",
  "pipelineVersion": 12,
  "stageName": "Source",
  "actionName": "Source",
  "startTime": 1598572624.387,
  "lastUpdateTime": 1598572628.16,
  "status": "Succeeded",
  "input": {
    "actionTypeId": {
      "category": "Source",
      "owner": "AWS",
      "provider": "CodeCommit",
      "version": "1"
    },
    "configuration": {
      "BranchName": "production",
      "PollForSourceChanges": "false",
      "RepositoryName": "my-repo"
    },
    "resolvedConfiguration": {
```

```

        "BranchName": "production",
        "PollForSourceChanges": "false",
        "RepositoryName": "my-repo"
    },
    "region": "us-east-1",
    "inputArtifacts": [],
    "namespace": "SourceVariables"
},
"output": {
    "outputArtifacts": [
        {
            "name": "SourceArtifact",
            "s3location": {
                "bucket": "my-bucket",
                "key": "myPipeline/SourceArti/key"
            }
        }
    ],
    "executionResult": {
        "externalExecutionId":
"11111111ad99dcd35914c00b7fbea13995EXAMPLE",
        "externalExecutionSummary": "Edited template.yml",
        "externalExecutionUrl": "https://myaddress.com"
    },
    "outputVariables": {
        "AuthorDate": "2020-05-08T17:45:43Z",
        "BranchName": "production",
        "CommitId": "EXAMPLEad99dcd35914c00b7fbea139951111111",
        "CommitMessage": "Edited template.yml",
        "CommitterDate": "2020-05-08T17:45:43Z",
        "RepositoryName": "my-repo"
    }
}
},
. . . .

```

Para obtener más información, consulte [Ver las ejecuciones de acciones \(CLI\)](#) en la Guía del AWS CodePipeline usuario.

- Para API obtener más información, consulte [ListActionExecutions](#) la Referencia de AWS CLI comandos.

## list-action-types

En el siguiente ejemplo de código, se muestra cómo usar `list-action-types`.

### AWS CLI

Para ver los tipos de acciones disponibles

Si se usa solo, el `list-action-types` comando devuelve la estructura de todas las acciones disponibles en su AWS cuenta. En este ejemplo, se utiliza la `action-owner-filter` opción `--` para devolver solo las acciones personalizadas.

Comando:

```
aws codepipeline list-action-types --action-owner-filter Custom
```

Salida:

```
{
  "actionTypes": [
    {
      "inputArtifactDetails": {
        "maximumCount": 5,
        "minimumCount": 0
      },
      "actionConfigurationProperties": [
        {
          "secret": false,
          "required": true,
          "name": "MyJenkinsExampleBuildProject",
          "key": true,
          "queryable": true
        }
      ],
      "outputArtifactDetails": {
        "maximumCount": 5,
        "minimumCount": 0
      },
      "id": {
        "category": "Build",
        "owner": "Custom",
        "version": "1",
```

```

        "provider": "MyJenkinsProviderName"
    },
    "settings": {
        "entityUrlTemplate": "http://192.0.2.4/job/{Config:ProjectName}",
        "executionUrlTemplate": "http://192.0.2.4/job/{Config:ProjectName}/
{ExternalExecutionId}"
    }
},
{
    "inputArtifactDetails": {
        "maximumCount": 5,
        "minimumCount": 0
    },
    "actionConfigurationProperties": [
        {
            "secret": false,
            "required": true,
            "name": "MyJenkinsExampleTestProject",
            "key": true,
            "queryable": true
        }
    ],
    "outputArtifactDetails": {
        "maximumCount": 5,
        "minimumCount": 0
    },
    "id": {
        "category": "Test",
        "owner": "Custom",
        "version": "1",
        "provider": "MyJenkinsProviderName"
    },
    "settings": {
        "entityUrlTemplate": "http://192.0.2.4/job/{Config:ProjectName}",
        "executionUrlTemplate": "http://192.0.2.4/job/{Config:ProjectName}/
{ExternalExecutionId}"
    }
}
]
}

```

- Para API obtener más información, consulte [ListActionTypes](#) la Referencia de AWS CLI comandos.



## list-pipeline-executions

En el siguiente ejemplo de código, se muestra cómo usar `list-pipeline-executions`.

### AWS CLI

Para ver el historial de ejecución de la canalización

En el siguiente `list-pipeline-executions` ejemplo, se muestra el historial de ejecución de una canalización de tu AWS cuenta.

```
aws codepipeline list-pipeline-executions \  
  --pipeline-name MyPipeline
```

Salida:

```
{  
  "pipelineExecutionSummaries": [  
    {  
      "lastUpdateTime": 1496380678.648,  
      "pipelineExecutionId": "7cf7f7cb-3137-539g-j458-d7eu3EXAMPLE",  
      "startTime": 1496380258.243,  
      "status": "Succeeded"  
    },  
    {  
      "lastUpdateTime": 1496591045.634,  
      "pipelineExecutionId": "3137f7cb-8d494hj4-039j-d84l-d7eu3EXAMPLE",  
      "startTime": 1496590401.222,  
      "status": "Succeeded"  
    },  
    {  
      "lastUpdateTime": 1496946071.6456,  
      "pipelineExecutionId": "4992f7jf-7cf7-913k-k334-d7eu3EXAMPLE",  
      "startTime": 1496945471.5645,  
      "status": "Succeeded"  
    }  
  ]  
}
```

Para obtener más información, consulta [Ver el historial de ejecuciones](#) en la Guía del AWS CodePipeline usuario.

- Para API obtener más información, consulte [ListPipelineExecutions](#) la Referencia de AWS CLI comandos.

## list-pipelines

En el siguiente ejemplo de código, se muestra cómo usar `list-pipelines`.

### AWS CLI

Para ver una lista de canalizaciones

En este ejemplo, se enumeran todas AWS CodePipeline las canalizaciones asociadas a la cuenta del AWS usuario.

Comando:

```
aws codepipeline list-pipelines
```

Salida:

```
{
  "pipelines": [
    {
      "updated": 1439504274.641,
      "version": 1,
      "name": "MyFirstPipeline",
      "created": 1439504274.641
    },
    {
      "updated": 1436461837.992,
      "version": 2,
      "name": "MySecondPipeline",
      "created": 1436460801.381
    }
  ]
}
```

- Para API obtener más información, consulte [ListPipelines](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

## AWS CLI

Para enumerar las etiquetas

En el siguiente `list-tags-for-resource` ejemplo, se recupera una lista de todas las etiquetas adjuntas al recurso de canalización especificado.

```
aws codepipeline list-tags-for-resource \  
  --resource-arn arn:aws:codepipeline:us-east-1:123456789012:MyPipeline
```

Salida:

```
{  
  "tags": {  
    "Project": "ProjectA",  
    "IscontainerBased": "true"  
  }  
}
```

Para obtener más información, consulte [Ver las etiquetas de una canalización \(CLI\)](#) en la Guía del AWS CodePipeline usuario.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## list-webhooks

En el siguiente ejemplo de código, se muestra cómo usar `list-webhooks`.

## AWS CLI

Para enumerar los webhooks

En el siguiente `list-webhooks` ejemplo, se recupera una lista de todas las etiquetas adjuntas al recurso de canalización especificado.

```
aws codepipeline list-webhooks \  
  --endpoint-url "https://codepipeline.eu-central-1.amazonaws.com" \  
  --region "eu-central-1"
```

Salida:

```
{
  "webhooks": [
    {
      "url": "https://webhooks.domain.com/
trigger1111111111EXAMPLE1111111111111111111": {
        "authenticationConfiguration": {
          "SecretToken": "Secret"
        },
        "name": "my-webhook",
        "authentication": "GITHUB_HMAC",
        "targetPipeline": "my-Pipeline",
        "targetAction": "Source",
        "filters": [
          {
            "jsonPath": "$.ref",
            "matchEquals": "refs/heads/{Branch}"
          }
        ]
      },
      "arn": "arn:aws:codepipeline:eu-central-1:123456789012:webhook:my-
webhook"
    }
  ]
}
```

Para obtener más información, consulta la sección [Lista los webhooks de tu cuenta](#) en la Guía del AWS CodePipeline usuario.

- Para API obtener más información, consulte [ListWebhooks](#) la Referencia de AWS CLI comandos.

## poll-for-jobs

En el siguiente ejemplo de código, se muestra cómo usar `poll-for-jobs`.

### AWS CLI

Para ver los trabajos disponibles

Este ejemplo devuelve información sobre cualquier trabajo en el que pueda actuar un trabajador. En este ejemplo, se utiliza un JSON archivo predefinido (`MyActionTypeInfo.json`) para proporcionar información sobre el tipo de acción para el que el trabajador procesa los trabajos.

Este comando solo se usa para acciones personalizadas. Cuando se invoca este comando, AWS CodePipeline devuelve las credenciales temporales del bucket de Amazon S3 que se utiliza para almacenar los artefactos de la canalización. Este comando también devolverá los valores secretos definidos para la acción, si los hay.

Comando:

```
aws codepipeline poll-for-jobs --cli-input-json file://MyActionTypeInfo.json
```

JSON contenido de ejemplo de archivo:

```
{
  "actionTypeId": {
    "category": "Test",
    "owner": "Custom",
    "provider": "MyJenkinsProviderName",
    "version": "1"
  },
  "maxBatchSize": 5,
  "queryParam": {
    "ProjectName": "MyJenkinsTestProject"
  }
}
```

Salida:

```
{
  "jobs": [
    {
      "accountId": "111111111111",
      "data": {
        "actionConfiguration": {
          "__type": "ActionConfiguration",
          "configuration": {
            "ProjectName": "MyJenkinsExampleTestProject"
          }
        },
        "actionTypeId": {
          "__type": "ActionTypeId",
          "category": "Test",
          "owner": "Custom",
          "provider": "MyJenkinsProviderName",
```

```

    "version": "1"
  },
  "artifactCredentials": {
    "__type": "AWSSessionCredentials",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "secretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
    "sessionToken":
    "fICcQD6m7oRw0uX0jANBqkqhkiG9w0BAQUFADCBiDELMaKGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwd
+a4GmWIWJ21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/
f0wYK8m9TrDHudUZg3qX4waLG5M43q7Wgc/
MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpEIbb30hjZnzcVQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQ
+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0FkbFFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs
  },
  "inputArtifacts": [
    {
      "__type": "Artifact",
      "location": {
        "s3Location": {
          "bucketName": "codepipeline-us-east-1-11EXAMPLE11",
          "objectKey": "MySecondPipeline/MyAppBuild/EXAMPLE"
        },
        "type": "S3"
      },
      "name": "MyAppBuild"
    }
  ],
  "outputArtifacts": [],
  "pipelineContext": {
    "__type": "PipelineContext",
    "action": {
      "name": "MyJenkinsTest-Action"
    },
    "pipelineName": "MySecondPipeline",
    "stage": {
      "name": "Testing"
    }
  },
  "id": "ef66c259-64f9-EXAMPLE",
  "nonce": "3"
}
]
}

```

- Para API obtener más información, consulte [PollForJobs](#) la Referencia de AWS CLI comandos.

## put-webhook

En el siguiente ejemplo de código, se muestra cómo usar `put-webhook`.

### AWS CLI

Para crear un webhook

En el siguiente `put-webhook` ejemplo, se crea un webhook para una acción fuente de GitHub la versión 1. Tras crear el webhook, debe utilizar el comando `register-webhook-with-third-party` para registrarlo.

```
aws codepipeline put-webhook \  
  --cli-input-json file://webhook_json.json \  
  --region "eu-central-1"
```

Contenidos de `webhook_json.json`:

```
{  
  "webhook": {  
    "name": "my-webhook",  
    "targetPipeline": "pipeline_name",  
    "targetAction": "source_action_name",  
    "filters": [  
      {  
        "jsonPath": "$.ref",  
        "matchEquals": "refs/heads/{Branch}"  
      }  
    ],  
    "authentication": "GITHUB_HMAC",  
    "authenticationConfiguration": {  
      "SecretToken": "secret"  
    }  
  }  
}
```

Salida:

```
{
```

```
"webhook": {
  "url": "https://webhooks.domain.com/
trigger1111111111EXAMPLE1111111111111111111",
  "definition": {
    "authenticationConfiguration": {
      "SecretToken": "secret"
    },
    "name": "my-webhook",
    "authentication": "GITHUB_HMAC",
    "targetPipeline": "pipeline_name",
    "targetAction": "Source",
    "filters": [
      {
        "jsonPath": "$.ref",
        "matchEquals": "refs/heads/{Branch}"
      }
    ]
  },
  "arn": "arn:aws:codepipeline:eu-central-1:123456789012:webhook:my-webhook"
},
"tags": [
  {
    "key": "Project",
    "value": "ProjectA"
  }
]
}
```

Para obtener más información, consulte [Crear un webhook para una GitHub fuente](#) en la Guía del AWS CodePipeline usuario.

- Para API obtener más información, consulte [PutWebhook](#) la Referencia de AWS CLI comandos.

## retry-stage-execution

En el siguiente ejemplo de código, se muestra cómo usar `retry-stage-execution`.

### AWS CLI

Para volver a intentar una acción fallida

En el siguiente `retry-stage-execution` ejemplo, se vuelve a intentar una etapa en la que se ha producido un error en la acción.



```
aws codepipeline retry-stage-execution \  
  --pipeline-name MyPipeline \  
  --stage-name Deploy \  
  --pipeline-execution-id b59babff-5f34-EXAMPLE \  
  --retry-mode FAILED_ACTIONS
```

Salida:

```
{  
  "pipelineExecutionId": "b59babff-5f34-EXAMPLE"  
}
```

Para obtener más información, consulte [Reintentar las acciones fallidas \(CLI\)](#) en la Guía del AWS CodePipeline usuario.

- Para API obtener más información, consulte [RetryStageExecution](#) la Referencia de AWS CLI comandos.

## start-pipeline-execution

En el siguiente ejemplo de código, se muestra cómo usar `start-pipeline-execution`.

### AWS CLI

Para ejecutar la última revisión a través de una canalización

En este ejemplo, se ejecuta la última revisión presente en la fase de origen de una canalización a través de la canalización denominada "MyFirstPipeline».

Comando:

```
aws codepipeline start-pipeline-execution --name MyFirstPipeline
```

Salida:

```
{  
  "pipelineExecutionId": "3137f7cb-7cf7-EXAMPLE"  
}
```

- Para API obtener más información, consulte [StartPipelineExecution](#) la Referencia de AWS CLI comandos.

## stop-pipeline-execution

En el siguiente ejemplo de código, se muestra cómo usar `stop-pipeline-execution`.

### AWS CLI

Para detener la ejecución de una canalización

En el siguiente `stop-pipeline-execution` ejemplo, por defecto, se espera hasta que finalicen las acciones en curso y, a continuación, se detiene la canalización. No puede elegir detener y esperar si la ejecución ya se encuentra en un estado `Stopping` (Deteniéndose). Puede optar por detener y abandonar una ejecución que ya se encuentra en un estado `Stopping` (Deteniéndose).

```
aws codepipeline stop-pipeline-execution \  
  --pipeline-name MyFirstPipeline \  
  --pipeline-execution-id d-EXAMPLE \  
  --reason "Stopping pipeline after the build action is done"
```

Este comando no devuelve ningún resultado.

Para obtener más información, consulte [Detener la ejecución de una canalización \(CLI\)](#) en la Guía del AWS CodePipeline usuario.

- Para API obtener más información, consulte [StopPipelineExecution](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar un recurso

El siguiente `tag-resource` ejemplo asocia un conjunto de etiquetas proporcionadas a una canalización. Utilice este comando para añadir o editar etiquetas.

```
aws codepipeline tag-resource \  
  --resource-arn arn:aws:codepipeline:us-east-1:123456789012:MyPipeline \  
  --tags key=Project,value=ProjectA key=IscontainerBased,value=true
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir etiquetas a una canalización \(CLI\)](#) en la Guía del AWS CodePipeline usuario.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar AWS etiquetas de un recurso de conexiones

En el siguiente `untag-resource` ejemplo, se elimina una etiqueta del recurso especificado.

```
aws codepipeline untag-resource \  
  --resource-arn arn:aws:codepipeline:us-east-1:123456789012:MyPipeline \  
  --tag-keys Project IscontainerBased
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar etiquetas de una canalización \(CLI\)](#) en la Guía del AWS CodePipeline usuario.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `update-pipeline`.

### AWS CLI

Para actualizar la estructura de una tubería

En este ejemplo, se utiliza el comando `update-pipeline` con el argumento `--cli-input-json`. En este ejemplo, se usa un JSON archivo predefinido (`MyFirstPipeline.json`) para actualizar la estructura de una canalización. AWS CodePipeline reconoce el nombre de la canalización contenido en

el JSON archivo y, a continuación, aplica cualquier cambio de los campos modificados de la estructura de la canalización para actualizarla.

Utilice las siguientes pautas al crear el JSON archivo predefinido:

Si está trabajando con una estructura de canalización recuperada mediante el comando `get-pipeline`, debe eliminar la sección de metadatos de la estructura de canalización del JSON archivo (los «metadatos»: {} líneas y los campos «creado», «ARNcanalización» y «actualizado» que contienen). El nombre de la canalización no se puede cambiar.

Comando:

```
aws codepipeline update-pipeline --cli-input-json file://MyFirstPipeline.json
```

Contenido del JSON archivo de muestra:

```
{
  "pipeline": {
    "roleArn": "arn:aws:iam::111111111111:role/AWS-CodePipeline-Service",
    "stages": [
      {
        "name": "Source",
        "actions": [
          {
            "inputArtifacts": [],
            "name": "Source",
            "actionTypeId": {
              "category": "Source",
              "owner": "AWS",
              "version": "1",
              "provider": "S3"
            },
            "outputArtifacts": [
              {
                "name": "MyApp"
              }
            ],
            "configuration": {
              "S3Bucket": "awscodepipeline-demo-bucket2",
              "S3ObjectKey": "aws-codepipeline-s3-aws-codedeploy_linux.zip"
            },
            "runOrder": 1
          }
        ]
      }
    ]
  }
}
```

```

    }
  ]
},
{
  "name": "Beta",
  "actions": [
    {
      "inputArtifacts": [
        {
          "name": "MyApp"
        }
      ],
      "name": "CodePipelineDemoFleet",
      "actionTypeId": {
        "category": "Deploy",
        "owner": "AWS",
        "version": "1",
        "provider": "CodeDeploy"
      },
      "outputArtifacts": [],
      "configuration": {
        "ApplicationName": "CodePipelineDemoApplication",
        "DeploymentGroupName": "CodePipelineDemoFleet"
      },
      "runOrder": 1
    }
  ]
}
],
"artifactStore": {
  "type": "S3",
  "location": "codepipeline-us-east-1-11EXAMPLE11"
},
"name": "MyFirstPipeline",
"version": 1
}
}

```

Salida:

```

{
  "pipeline": {
    "artifactStore": {

```

```
    "location": "codepipeline-us-east-1-11EXAMPLE11",
    "type": "S3"
  },
  "name": "MyFirstPipeline",
  "roleArn": "arn:aws:iam::111111111111:role/AWS-CodePipeline-Service",
  "stages": [
    {
      "actions": [
        {
          "actionTypeId": {
            "__type": "ActionTypeId",
            "category": "Source",
            "owner": "AWS",
            "provider": "S3",
            "version": "1"
          },
          "configuration": {
            "S3Bucket": "awscodepipeline-demo-bucket2",
            "S3ObjectKey": "aws-codepipeline-s3-aws-codedeploy_linux.zip"
          },
          "inputArtifacts": [],
          "name": "Source",
          "outputArtifacts": [
            {
              "name": "MyApp"
            }
          ],
          "runOrder": 1
        }
      ],
      "name": "Source"
    },
    {
      "actions": [
        {
          "actionTypeId": {
            "__type": "ActionTypeId",
            "category": "Deploy",
            "owner": "AWS",
            "provider": "CodeDeploy",
            "version": "1"
          },
          "configuration": {
            "ApplicationName": "CodePipelineDemoApplication",
```

```
        "DeploymentGroupName": "CodePipelineDemoFleet"
    },
    "inputArtifacts": [
        {
            "name": "MyApp"
        }
    ],
    "name": "CodePipelineDemoFleet",
    "outputArtifacts": [],
    "runOrder": 1
    }
],
"name": "Beta"
}
],
"version": 3
}
}
```

- Para API obtener más información, consulte [UpdatePipeline](#) la Referencia de AWS CLI comandos.

## AWS CodeStar ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS CodeStar.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **associate-team-member**

En el siguiente ejemplo de código, se muestra cómo usar `associate-team-member`.

#### AWS CLI

Para añadir un miembro del equipo a un proyecto

El siguiente `associate-team-member` ejemplo convierte al `intern` usuario en espectador del proyecto con el ID especificado.

```
aws codestar associate-team-member \  
  --project-id my-project \  
  --user-arn arn:aws:iam::123456789012:user/intern \  
  --project-role Viewer
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [AssociateTeamMember](#) la Referencia de AWS CLI comandos.

### **create-project**

En el siguiente ejemplo de código, se muestra cómo usar `create-project`.

#### AWS CLI

Para crear un proyecto

El siguiente `create-project` ejemplo utiliza un archivo JSON de entrada para crear un CodeStar proyecto.

```
aws codestar create-project \  
  --cli-input-json file://create-project.json
```

Contenidos de `create-project.json`:

```
{  
  "name": "Custom Project",  
  "id": "custom-project",
```



```

    "sourceCode": [
      {
        "source": {
          "s3": {
            "bucketName": "codestar-artifacts",
            "bucketKey": "nodejs-function.zip"
          }
        },
        "destination": {
          "codeCommit": {
            "name": "codestar-custom-project"
          }
        }
      }
    ],
    "toolchain": {
      "source": {
        "s3": {
          "bucketName": "codestar-artifacts",
          "bucketKey": "toolchain.yml"
        }
      },
      "roleArn": "arn:aws:iam::123456789012:role/service-role/aws-codestar-
service-role",
      "stackParameters": {
        "ProjectId": "custom-project"
      }
    }
  }
}

```

**Salida:**

```

{
  "id": "my-project",
  "arn": "arn:aws:codestar:us-east-2:123456789012:project/custom-project"
}

```

Para ver un tutorial que incluye ejemplos de código y plantillas para un proyecto personalizado, consulte [Crear un proyecto AWS CodeStar con la AWS CLI < https://docs.aws.amazon.com/codestar/latest/userguide/cli-tutorial.html >](https://docs.aws.amazon.com/codestar/latest/userguide/cli-tutorial.html) de la Guía del usuario.AWS CodeStar

- Para API obtener más información, consulte la Referencia de comandos. [CreateProject](#) AWS CLI

## create-user-profile

En el siguiente ejemplo de código, se muestra cómo usar `create-user-profile`.

### AWS CLI

Para crear un perfil de usuario

En el siguiente `create-user-profile` ejemplo, se crea un perfil de usuario para el IAM usuario con los valores especificadosARN.

```
aws codestar create-user-profile \  
  --user-arn arn:aws:iam::123456789012:user/intern \  
  --display-name Intern \  
  --email-address intern@example.com
```

Salida:

```
{  
  "userArn": "arn:aws:iam::123456789012:user/intern",  
  "displayName": "Intern",  
  "emailAddress": "intern@example.com",  
  "sshPublicKey": "",  
  "createdTimestamp": 1572552308.607,  
  "lastModifiedTimestamp": 1572552308.607  
}
```

- Para API obtener más información, consulte [CreateUserProfile](#) la Referencia de AWS CLI comandos.

## delete-project

En el siguiente ejemplo de código, se muestra cómo usar `delete-project`.

### AWS CLI

Para eliminar un proyecto

En el siguiente `delete-project` ejemplo, se elimina el proyecto especificado.

```
aws codestar delete-project \  
  --project-id my-project
```

Salida:

```
{
  "projectArn": "arn:aws:codestar:us-east-2:123456789012:project/my-project"
}
```

- Para API obtener más información, consulte [DeleteProject](#) la Referencia de AWS CLI comandos.

## delete-user-profile

En el siguiente ejemplo de código, se muestra cómo usar `delete-user-profile`.

AWS CLI

Para eliminar un perfil de usuario

En el siguiente `delete-user-profile` ejemplo, se elimina el perfil de usuario del usuario especificadoARN.

```
aws codestar delete-user-profile \
  --user-arn arn:aws:iam::123456789012:user/intern
```

Salida:

```
{
  "userArn": "arn:aws:iam::123456789012:user/intern"
}
```

- Para API obtener más información, consulte [DeleteUserProfile](#) la Referencia de AWS CLI comandos.

## describe-project

En el siguiente ejemplo de código, se muestra cómo usar `describe-project`.

AWS CLI

Para ver un proyecto

En el siguiente describe-project ejemplo, se recuperan los detalles del proyecto especificado.

```
aws codestar describe-project \  
  --id my-project
```

Salida:

```
{  
  "name": "my project",  
  "id": "my-project",  
  "arn": "arn:aws:codestar:us-west-2:123456789012:project/my-project",  
  "description": "My first CodeStar project.",  
  "createdTimeStamp": 1572547510.128,  
  "status": {  
    "state": "CreateComplete"  
  }  
}
```

- Para API obtener más información, consulte [DescribeProject](#) la Referencia de AWS CLI comandos.

## describe-user-profile

En el siguiente ejemplo de código, se muestra cómo usar describe-user-profile.

AWS CLI

Para ver un perfil de usuario

En el siguiente describe-user-profile ejemplo, se recuperan detalles sobre el perfil de usuario del usuario especificadoARN.

```
aws codestar describe-user-profile \  
  --user-arn arn:aws:iam::123456789012:user/intern
```

Salida:

```
{  
  "userArn": "arn:aws:iam::123456789012:user/intern",  
  "displayName": "Intern",
```

```
"emailAddress": "intern@example.com",
"sshPublicKey": "intern",
"createdTimestamp": 1572552308.607,
"lastModifiedTimestamp": 1572553495.47
}
```

- Para API obtener más información, consulte [DescribeUserProfile](#) la Referencia de AWS CLI comandos.

## disassociate-team-member

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-team-member`.

### AWS CLI

Para eliminar a un miembro del equipo

En el siguiente `disassociate-team-member` ejemplo, se elimina del proyecto al usuario con lo especificado `ARNmy-project`.

```
aws codestar disassociate-team-member \
  --project-id my-project \
  --user-arn arn:aws:iam::123456789012:user/intern
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DisassociateTeamMember](#) la Referencia de AWS CLI comandos.

## list-projects

En el siguiente ejemplo de código, se muestra cómo usar `list-projects`.

### AWS CLI

Para ver los proyectos

El siguiente `list-projects` ejemplo recupera una lista de proyectos de la región actual.

```
aws codestar list-projects
```

Salida:

```
{
  "projects": [
    {
      "projectId": "intern-projects",
      "projectArn": "arn:aws:codestar:us-west-2:123456789012:project/intern-
projects"
    },
    {
      "projectId": "my-project",
      "projectArn": "arn:aws:codestar:us-west-2:123456789012:project/my-
project"
    }
  ]
}
```

- Para API obtener más información, consulte [ListProjects](#) la Referencia de AWS CLI comandos.

## list-resources

En el siguiente ejemplo de código, se muestra cómo usar `list-resources`.

### AWS CLI

Para ver los recursos

El siguiente `list-resources` ejemplo recupera una lista de recursos para el proyecto especificado.

```
aws codestar list-resources \
  --id my-project
```

Salida:

```
{
  "resources": [
    {
      "id": "arn:aws:execute-api:us-east-2:123456789012:r3wxmplbv8"
    },
    {
      "id": "arn:aws:codedeploy:us-
east-2:123456789012:application:awscodestar-my-project-lambda-
ServerlessDeploymentApplication-PF0LXMPL1KA0"
    }
  ]
}
```

```
    },
    {
      "id": "arn:aws:s3::aws-codestar-us-east-2-123456789012-my-project-pipe"
    },
    {
      "id": "arn:aws:lambda:us-east-2:123456789012:function:awscodestar-my-
project-lambda-GetHelloWorld-16W3LVXMPLNNS"
    },
    {
      "id": "arn:aws:cloudformation:us-east-2:123456789012:stack/awscodestar-
my-project-lambda/b4904ea0-fc20-xmpl-bec6-029123b1cc42"
    },
    {
      "id": "arn:aws:cloudformation:us-east-2:123456789012:stack/awscodestar-
my-project/1b133f30-fc20-xmpl-a93a-0688c4290cb8"
    },
    {
      "id": "arn:aws:iam::123456789012:role/CodeStarWorker-my-project-
ToolChain"
    },
    {
      "id": "arn:aws:iam::123456789012:policy/CodeStar_my-
project_PermissionsBoundary"
    },
    {
      "id": "arn:aws:s3::aws-codestar-us-east-2-123456789012-my-project-app"
    },
    {
      "id": "arn:aws:codepipeline:us-east-2:123456789012:my-project-Pipeline"
    },
    {
      "id": "arn:aws:codedeploy:us-east-2:123456789012:deploymentgroup:my-
project/awscodestar-my-project-lambda-GetHelloWorldDeploymentGroup-P7YWXMPLT0QB"
    },
    {
      "id": "arn:aws:iam::123456789012:role/CodeStar-my-project-Execution"
    },
    {
      "id": "arn:aws:iam::123456789012:role/CodeStarWorker-my-project-
CodeDeploy"
    },
    {
      "id": "arn:aws:codebuild:us-east-2:123456789012:project/my-project"
    },
  ],
}
```

```
{
  "id": "arn:aws:iam::123456789012:role/CodeStarWorker-my-project-
CloudFormation"
},
{
  "id": "arn:aws:codecommit:us-east-2:123456789012:Go-project"
}
]
```

- Para API obtener más información, consulte [ListResources](#) la Referencia de AWS CLI comandos.

## list-tags-for-project

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-project`.

### AWS CLI

Para ver las etiquetas de un proyecto

En el siguiente `list-tags-for-project` ejemplo, se recuperan las etiquetas adjuntas al proyecto especificado.

```
aws codestar list-tags-for-project \
  --id my-project
```

Salida:

```
{
  "tags": {
    "Department": "Marketing",
    "Team": "Website"
  }
}
```

- Para API obtener más información, consulte [ListTagsForProject](#) la Referencia de AWS CLI comandos.



## list-team-members

En el siguiente ejemplo de código, se muestra cómo usar `list-team-members`.

### AWS CLI

Para ver una lista de los miembros del equipo

En el siguiente `list-team-members` ejemplo, se recupera una lista de usuarios asociados al proyecto especificado.

```
aws codestar list-team-members \  
  --project-id my-project
```

Salida:

```
{  
  "teamMembers": [  
    {  
      "userArn": "arn:aws:iam::123456789012:user/admin",  
      "projectRole": "Owner",  
      "remoteAccessAllowed": false  
    },  
    {  
      "userArn": "arn:aws:iam::123456789012:user/intern",  
      "projectRole": "Contributor",  
      "remoteAccessAllowed": false  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListTeamMembers](#) la Referencia de AWS CLI comandos.

## list-user-profiles

En el siguiente ejemplo de código, se muestra cómo usar `list-user-profiles`.

### AWS CLI

Para ver una lista de perfiles de usuario

El siguiente `list-user-profiles` ejemplo recupera una lista de todos los perfiles de usuario de la región actual.

```
aws codestar list-user-profiles
```

Salida:

```
{
  "userProfiles": [
    {
      "userArn": "arn:aws:iam::123456789012:user/admin",
      "displayName": "me",
      "emailAddress": "me@example.com",
      "sshPublicKey": ""
    },
    {
      "userArn": "arn:aws:iam::123456789012:user/intern",
      "displayName": "Intern",
      "emailAddress": "intern@example.com",
      "sshPublicKey": "intern"
    }
  ]
}
```

- Para API obtener más información, consulte [ListUserProfiles](#) la Referencia de AWS CLI comandos.

## tag-project

En el siguiente ejemplo de código, se muestra cómo usar `tag-project`.

AWS CLI

Para adjuntar una etiqueta a un proyecto

En el siguiente `tag-project` ejemplo, se agrega una etiqueta denominada `Department` y un valor de `Marketing` al proyecto especificado.

```
aws codestar tag-project \
  --id my-project \
```

```
--tags Department=Marketing
```

Salida:

```
{
  "tags": {
    "Department": "Marketing"
  }
}
```

- Para API obtener más información, consulte [TagProject](#) la Referencia de AWS CLI comandos.

## untag-project

En el siguiente ejemplo de código, se muestra cómo usar `untag-project`.

### AWS CLI

Para eliminar una etiqueta de un proyecto

En el siguiente `untag-project` ejemplo, se elimina cualquier etiqueta con un nombre clave `Team` de del proyecto especificado.

```
aws codestar untag-project \  
  --id my-project \  
  --tags Team
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte la Referencia [UntagProject](#) de AWS CLI comandos.

## update-project

En el siguiente ejemplo de código, se muestra cómo usar `update-project`.

### AWS CLI

Para actualizar un proyecto

En el siguiente `update-project` ejemplo, se agrega una descripción al proyecto especificado.

```
aws codestar update-project \  
  --id my-project \  
  --description "My first CodeStar project"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UpdateProject](#) la Referencia de AWS CLI comandos.

## update-team-member

En el siguiente ejemplo de código, se muestra cómo usar `update-team-member`.

### AWS CLI

Para modificar un miembro del equipo

El siguiente `update-team-member` ejemplo convierte al usuario especificado en colaborador de un proyecto y le concede acceso remoto a los recursos del proyecto.

```
aws codestar update-team-member \  
  --project-id my-project \  
  --user-arn arn:aws:iam::123456789012:user/intern \  
  --project-role Contributor -\  
  --remote-access-allowed
```

Salida:

```
{  
  "userArn": "arn:aws:iam::123456789012:user/intern",  
  "projectRole": "Contributor",  
  "remoteAccessAllowed": true  
}
```

- Para API obtener más información, consulte [UpdateTeamMember](#) la Referencia de AWS CLI comandos.

## update-user-profile

En el siguiente ejemplo de código, se muestra cómo usar `update-user-profile`.

## AWS CLI

Para modificar un perfil de usuario

En el siguiente `update-user-profile` ejemplo, se agrega la SSH clave especificada al usuario especificado.

```
aws codestar update-user-profile \  
  --ssh-public-key intern \  
  --user-arn arn:aws:iam::123456789012:user/intern
```

Salida:

```
{  
  "userArn": "arn:aws:iam::123456789012:user/intern",  
  "displayName": "Intern",  
  "emailAddress": "intern@example.com",  
  "sshPublicKey": "intern",  
  "createdTimestamp": 1572552308.607,  
  "lastModifiedTimestamp": 1572553495.47  
}
```

- Para API obtener más información, consulte [UpdateUserProfile](#) la Referencia de AWS CLI comandos.

## AWS CodeStar Ejemplos de notificaciones que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso AWS Command Line Interface de AWS CodeStar notificaciones.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **create-notification-rule**

En el siguiente ejemplo de código, se muestra cómo usar `create-notification-rule`.

#### AWS CLI

Para crear una regla de notificación

En el siguiente `create-notification-rule` ejemplo, se utiliza un JSON archivo denominado `rule.json` para crear una regla de notificación con el nombre `MyNotificationRule` de un repositorio nombrado `MyDemoRepo` en la AWS cuenta especificada. Las notificaciones con el tipo de FULL detalle se envían al SNS tema de Amazon de destino especificado cuando se crean las sucursales y las etiquetas.

```
aws codestar-notifications create-notification-rule \  
--cli-input-json file://rule.json
```

Contenidos de `rule.json`:

```
{  
  "Name": "MyNotificationRule",  
  "EventTypeIds": [  
    "codecommit-repository-branches-and-tags-created"  
  ],  
  "Resource": "arn:aws:codecommit:us-east-1:123456789012:MyDemoRepo",  
  "Targets": [  
    {  
      "TargetType": "SNS",  
      "TargetAddress": "arn:aws:sns:us-east-1:123456789012:MyNotificationTopic"  
    }  
  ],  
  "Status": "ENABLED",  
  "DetailType": "FULL"  
}
```

Salida:

```
{
```

```
"Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
}
```

Para obtener más información, consulte [Crear una regla de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [CreateNotificationRule](#) la Referencia de AWS CLI comandos.

## delete-notification-rule

En el siguiente ejemplo de código, se muestra cómo usar `delete-notification-rule`.

### AWS CLI

Para eliminar una regla de notificación

En el siguiente `delete-notification-rule` ejemplo, se elimina la regla de notificación especificada.

```
aws codestar-notifications delete-notification-rule \
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE
```

Salida:

```
{
  "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
}
```

Para obtener más información, consulte [Eliminar una regla de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [DeleteNotificationRule](#) la Referencia de AWS CLI comandos.

## delete-target

En el siguiente ejemplo de código, se muestra cómo usar `delete-target`.

## AWS CLI

Para eliminar un objetivo de regla de notificación

En el siguiente `delete-target` ejemplo, se elimina el objetivo especificado de todas las reglas de notificación configuradas para usarlo como destino y, a continuación, se elimina el objetivo.

```
aws codestar-notifications delete-target \  
  --target-address arn:aws:sns:us-east-1:123456789012:MyNotificationTopic \  
  --force-unsubscribe-all
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un objetivo de regla de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [DeleteTarget](#) la Referencia de AWS CLI comandos.

## describe-notification-rule

En el siguiente ejemplo de código, se muestra cómo usar `describe-notification-rule`.

## AWS CLI

Para recuperar los detalles de una regla de notificación

En el siguiente `describe-notification-rule` ejemplo, se recuperan los detalles de la regla de notificación especificada.

```
aws codestar-notifications describe-notification-rule \  
  --arn arn:aws:codestar-notifications:us-west-2:123456789012:notificationrule/  
dc82df7a-EXAMPLE
```

Salida:

```
{  
  "LastModifiedTimestamp": 1569199844.857,  
  "EventTypes": [  
    {  
      "ServiceName": "CodeCommit",  
      "EventTypeName": "Branches and tags: Created",
```



```

        "ResourceType": "Repository",
        "EventTypeId": "codecommit-repository-branches-and-tags-created"
    }
],
"Status": "ENABLED",
"DetailType": "FULL",
"Resource": "arn:aws:codecommit:us-west-2:123456789012:MyDemoRepo",
"Arn": "arn:aws:codestar-notifications:us-west-w:123456789012:notificationrule/
dc82df7a-EXAMPLE",
"Targets": [
    {
        "TargetStatus": "ACTIVE",
        "TargetAddress": "arn:aws:sns:us-
west-2:123456789012:MyNotificationTopic",
        "TargetType": "SNS"
    }
],
"Name": "MyNotificationRule",
"CreatedTimestamp": 1569199844.857,
"CreatedBy": "arn:aws:iam::123456789012:user/Mary_Major"
}

```

Para obtener más información, consulte [Ver las reglas de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [DescribeNotificationRule](#) la Referencia de AWS CLI comandos.

## list-event-types

En el siguiente ejemplo de código, se muestra cómo usar `list-event-types`.

### AWS CLI

Para obtener una lista de los tipos de eventos de una regla de notificación

En el siguiente `list-event-types` ejemplo, se recupera una lista filtrada de todos los tipos de eventos de notificación disponibles para CodeDeploy las aplicaciones. Si, por el contrario, no utiliza ningún filtro, el comando devuelve todos los tipos de eventos de notificación para todos los tipos de recursos.

```
aws codestar-notifications list-event-types \
```

```
--filters Name=SERVICE_NAME,Value=CodeDeploy
```

Salida:

```
{
  "EventTypes": [
    {
      "EventTypeId": "codedeploy-application-deployment-succeeded",
      "ServiceName": "CodeDeploy",
      "EventTypeName": "Deployment: Succeeded",
      "ResourceType": "Application"
    },
    {
      "EventTypeId": "codedeploy-application-deployment-failed",
      "ServiceName": "CodeDeploy",
      "EventTypeName": "Deployment: Failed",
      "ResourceType": "Application"
    },
    {
      "EventTypeId": "codedeploy-application-deployment-started",
      "ServiceName": "CodeDeploy",
      "EventTypeName": "Deployment: Started",
      "ResourceType": "Application"
    }
  ]
}
```

Para obtener más información, consulte [Crear una regla de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [ListEventTypes](#) la Referencia de AWS CLI comandos.

## list-notification-rules

En el siguiente ejemplo de código, se muestra cómo usar `list-notification-rules`.

### AWS CLI

Para recuperar una lista de reglas de notificación

En el siguiente `list-notification-rules` ejemplo, se recupera una lista de todas las reglas de notificación de la AWS región especificada.

```
aws codestar-notifications list-notification-rules --region us-east-1
```

Salida:

```
{
  "NotificationRules": [
    {
      "Id": "dc82df7a-EXAMPLE",
      "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/dc82df7a-EXAMPLE"
    },
    {
      "Id": "8d1f0983-EXAMPLE",
      "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/8d1f0983-EXAMPLE"
    }
  ]
}
```

Para obtener más información, consulte [Ver las reglas de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [ListNotificationRules](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

AWS CLI

Para obtener una lista de etiquetas adjuntas a una regla de notificación

En el siguiente `list-tags-for-resource` ejemplo, se recupera una lista de todas las etiquetas adjuntas a la regla de notificación especificada. En este ejemplo, la regla de notificación actualmente no tiene ninguna etiqueta asociada.

```
aws codestar-notifications list-tags-for-resource \
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/fe1efd35-EXAMPLE
```

Salida:

```
{
  "Tags": {}
}
```

Para obtener más información, consulte [Crear una regla de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## list-targets

En el siguiente ejemplo de código, se muestra cómo usar `list-targets`.

AWS CLI

Para recuperar una lista de los objetivos de las reglas de notificación

En el siguiente `list-targets` ejemplo, se recupera una lista de todos los objetivos de las reglas de notificación de la AWS región especificada.

```
aws codestar-notifications list-targets \
  --region us-east-1
```

Salida:

```
{
  "Targets": [
    {
      "TargetAddress": "arn:aws:sns:us-
east-1:123456789012:MySNSTopicForNotificationRules",
      "TargetType": "SNS",
      "TargetStatus": "ACTIVE"
    },
    {
      "TargetAddress": "arn:aws:sns:us-
east-1:123456789012:MySNSTopicForNotificationsAboutMyDemoRepo",
      "TargetType": "SNS",
      "TargetStatus": "ACTIVE"
    }
  ]
}
```

```

    }
  ]
}

```

Para obtener más información, consulte [Ver los objetivos de las reglas de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [ListTargets](#) la Referencia de AWS CLI comandos.

## subscribe

En el siguiente ejemplo de código, se muestra cómo usar `subscribe`.

### AWS CLI

Para añadir un objetivo a una regla de notificación

En el siguiente `subscribe` ejemplo, se añade un SNS tema de Amazon como destino para la regla de notificación especificada.

```

aws codestar-notifications subscribe \
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/dc82df7a-EXAMPLE \
  --target TargetType=SNS,TargetAddress=arn:aws:sns:us-east-1:123456789012:MyNotificationTopic

```

Salida:

```

{
  "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/dc82df7a-EXAMPLE"
}

```

Para obtener más información, consulte [Añadir o eliminar un SNS tema de Amazon como objetivo de una regla de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [Suscríbete](#) en AWS CLI Command Reference.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

## AWS CLI

Para añadir una etiqueta a una regla de notificación

En el siguiente `tag-resource` ejemplo, se agrega una etiqueta con el nombre de clave `Team` y el valor de `Li_Juan` a la regla de notificación especificada.

```
aws codestar-notifications tag-resource \  
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/  
fe1efd35-EXAMPLE \  
  --tags Team=Li_Juan
```

Salida:

```
{  
  "Tags": {  
    "Team": "Li_Juan"  
  }  
}
```

Para obtener más información, consulte [Crear una regla de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## unsubscribe

En el siguiente ejemplo de código, se muestra cómo usar `unsubscribe`.

## AWS CLI

Para eliminar un objetivo de una regla de notificación

En el siguiente `unsubscribe` ejemplo, se elimina un SNS tema de Amazon como objetivo de la regla de notificación especificada.

```
aws codestar-notifications unsubscribe \  
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/  
dc82df7a-EXAMPLE \  
  --target TargetType=SNS,TargetAddress=arn:aws:sns:us-  
east-1:123456789012:MyNotificationTopic
```

Salida:

```
{
  "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
  "TargetAddress": "arn:aws:sns:us-east-1:123456789012:MyNotificationTopic"
}
```

Para obtener más información, consulte [Añadir o eliminar un SNS tema de Amazon como objetivo de una regla de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [Cancelar la suscripción](#) en la referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

AWS CLI

Para eliminar una etiqueta de una regla de notificación

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta con el nombre `Team` de clave de la regla de notificación especificada.

```
aws codestar-notifications untag-resource \
  --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
fe1efd35-EXAMPLE \
  --tag-keys Team
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Editar una regla de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [UntagResource](#) en la Referencia de AWS CLI comandos.

## update-notification-rule

En el siguiente ejemplo de código, se muestra cómo usar `update-notification-rule`.

## AWS CLI

Para actualizar una regla de notificación

En el siguiente `update-notification-rule` ejemplo, se actualiza una regla de notificación nombrada `MyNotificationRule` en la AWS cuenta `123456789012` mediante un JSON archivo denominado `update.json`.

```
aws codestar-notifications update-notification-rule \  
  --cli-input-json file://update.json
```

Contenidos de `update.json`:

```
{  
  "Name": "MyUpdatedNotificationRule",  
  "EventTypeId": [ "codecommit-repository-branches-and-tags-created" ],  
  "Resource": "arn:aws:codecommit:us-east-1:123456789012:MyDemoRepo",  
  "Targets": [ {  
    "TargetType": "SNS",  
    "TargetAddress": "arn:aws:sns:us-east-1:123456789012:MyNotificationTopic"  
  } ],  
  "Status": "ENABLED",  
  "DetailType": "FULL"  
}
```

Salida:

```
{  
  "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/  
dc82df7a-EXAMPLE"  
}
```

Para obtener más información, consulte [Editar una regla de notificación](#) en la Guía del usuario de AWS Developer Tools Console.

- Para API obtener más información, consulte [UpdateNotificationRule](#) la Referencia de AWS CLI comandos.



## CodeConnections ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with CodeConnections.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-connection**

En el siguiente ejemplo de código, se muestra cómo usar `create-connection`.

#### AWS CLI

Para crear una conexión

El siguiente `create-connection` ejemplo muestra cómo crear una conexión a un repositorio de terceros. En este ejemplo, se crea una conexión en la que el proveedor externo es Bitbucket.

Una conexión creada a través de AWS CLI o AWS CloudFormation está en estado Pendiente de forma predeterminada. Tras crear una conexión con el CLI o AWS CloudFormation, utilice la consola para editar la conexión y hacer que su estado sea Disponible.

```
aws codestar-connections create-connection \  
  --provider-type Bitbucket \  
  --connection-name MyConnection
```

Salida:

```
{
```

```
"ConnectionArn": "arn:aws:codestar-connections:us-east-1:123456789012:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f"
}
```

Para obtener más información, consulte [Crear una conexión](#) en la Guía del usuario de la consola Developer Tools.

- Para API obtener más información, consulte [CreateConnection](#) la Referencia de AWS CLI comandos.

## create-host

En el siguiente ejemplo de código, se muestra cómo usar `create-host`.

### AWS CLI

Para crear un host

En el siguiente `create-host` ejemplo, se muestra cómo crear un host que represente el punto final de la infraestructura en la que está instalado el proveedor externo. En este ejemplo, se crea un host en el que el proveedor externo instalado es GitHub Enterprise Server.

Un host creado mediante el AWS CLI está en estado Pendiente de forma predeterminada. Después de crear un host con el CLI, utilice la consola o el CLI para configurar el host y hacer que su estado sea Disponible.

```
aws codestar-connections create-host \
  --name MyHost \
  --provider-type GitHubEnterpriseServer \
  --provider-endpoint "https://my-instance.dev"
```

Salida:

```
{
  "HostArn": "arn:aws:codestar-connections:us-east-1:123456789012:host/My-Host-28aef605"
}
```

Para obtener más información, consulte [Crear un host \(CLI\)](#) en la Guía del usuario de la consola Developer Tools.

- Para API obtener más información, consulte [CreateHost](#) la Referencia de AWS CLI comandos.

## delete-connection

En el siguiente ejemplo de código, se muestra cómo usar `delete-connection`.

### AWS CLI

Para eliminar una conexión

En el siguiente `delete-connection` ejemplo, se muestra cómo eliminar una conexión.

```
aws codestar-connections delete-connection \  
  --connection-arn arn:aws:codestar-connections:us-west-2:123456789012:connection/  
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar una conexión \(CLI\)](#) en la Guía del usuario de la consola de Developer Tools.

- Para API obtener más información, consulte [DeleteConnection](#) la Referencia de AWS CLI comandos.

## delete-host

En el siguiente ejemplo de código, se muestra cómo usar `delete-host`.

### AWS CLI

Para eliminar un host

En el siguiente `delete-host` ejemplo, se muestra cómo eliminar un host. Para poder eliminar un alojamiento, debe eliminar todas las conexiones asociadas al alojamiento.

```
aws codestar-connections delete-host \  
  --host-arn "arn:aws:codestar-connections:us-east-1 :123456789012:host/My-  
Host-28aef605"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un host \(CLI\)](#) en la Guía del usuario de la consola Developer Tools.

- Para API obtener más información, consulte [DeleteHost](#) la Referencia de AWS CLI comandos.

## get-connection

En el siguiente ejemplo de código, se muestra cómo usar `get-connection`.

### AWS CLI

Para obtener información sobre una conexión

En el siguiente `get-connection` ejemplo, se muestran detalles sobre una conexión.

```
aws codestar-connections get-connection \  
  --connection-arn arn:aws:codestar-connections:us-east-1:123456789012:connection/  
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
```

Salida:

```
{  
  "Connection": {  
    "ConnectionName": "MyConnection",  
    "ConnectionArn": "arn:aws:codestar-connections:us-  
east-1:123456789012:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",  
    "ProviderType": "Bitbucket",  
    "OwnerAccountId": "123456789012",  
    "ConnectionStatus": "AVAILABLE"  
  }  
}
```

Para obtener más información, consulte [Ver los detalles de la conexión](#) en la Guía del usuario de la consola Developer Tools.

- Para API obtener más información, consulte [GetConnection](#) la Referencia de AWS CLI comandos.

## get-host

En el siguiente ejemplo de código, se muestra cómo usar `get-host`.

### AWS CLI

Para obtener información sobre un host

En el siguiente `get-host` ejemplo, se muestran detalles sobre un anfitrión:

```
aws codestar-connections get-host \  
  --host-arn arn:aws:codestar-connections:us-east-1:123456789012:host/MyHost-28aef605
```

Salida:

```
{  
  "Name": "MyHost",  
  "Status": "AVAILABLE",  
  "ProviderType": "GitHubEnterpriseServer",  
  "ProviderEndpoint": "https://test-instance-1.dev/"  
}
```

Para obtener más información, consulte [Ver los detalles del host \(CLI\)](#) en la Guía del usuario de la consola Developer Tools.

- Para API obtener más información, consulte [GetHost](#) la Referencia de AWS CLI comandos.

## list-connections

En el siguiente ejemplo de código, se muestra cómo usar `list-connections`.

### AWS CLI

Para enumerar las conexiones

En el siguiente `list-connections` ejemplo, se recupera una lista de todas las conexiones de tu cuenta para el tipo de proveedor de Bitbucket. :

```
aws codestar-connections list-connections \  
  --provider-type Bitbucket \  
  --max-results 5 \  
  --next-token: next-token
```

Salida:

```
{  
  "Connections": [  
    {  
      "ConnectionName": "my-connection",  
      "ProviderType": "Bitbucket",
```

```
    "Status": "PENDING",
    "ARN": "arn:aws:codestar-connections:us-east-1:123456789012:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
    "OwnerAccountId": "123456789012"
  },
  {
    "ConnectionName": "my-other-connection",
    "ProviderType": "Bitbucket",
    "Status": "AVAILABLE",
    "ARN": "arn:aws:codestar-connections:us-east-1:123456789012:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
    "OwnerAccountId": "123456789012"
  },
],
"NextToken": "next-token"
}
```

Para obtener más información, consulte [Listar conexiones \(CLI\)](#) en la Guía del usuario de la consola Developer Tools.

- Para API obtener más información, consulte [ListConnections](#) la Referencia de AWS CLI comandos.

## list-hosts

En el siguiente ejemplo de código, se muestra cómo usar `list-hosts`.

### AWS CLI

Para enumerar los hosts

En el siguiente `list-hosts` ejemplo, se recupera una lista de todos los hosts de su cuenta.

```
aws codestar-connections list-hosts
```

Salida:

```
{
  "Hosts": [
    {
      "Name": "My-Host",
      "HostArn": "arn:aws:codestar-connections:us-east-1:123456789012:host/My-
Host-28aef605",

```

```
        "ProviderType": "GitHubEnterpriseServer",
        "ProviderEndpoint": "https://my-instance.test.dev",
        "Status": "AVAILABLE"
    }
]
}
```

Para obtener más información, consulte [List hosts \(CLI\)](#) en la Guía del usuario de la consola Developer Tools.

- Para API obtener más información, consulte [ListHosts](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas

En el siguiente `list-tags-for-resource` ejemplo, se recupera una lista de todas las etiquetas adjuntas al recurso de conexiones especificado.

```
aws codestar-connections list-tags-for-resource \
  --resource-arn arn:aws:codestar-connections:us-east-1:123456789012:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
```

Salida:

```
{
  "Tags": [
    {
      "Key": "Project",
      "Value": "ProjectA"
    },
    {
      "Key": "ReadOnly",
      "Value": "true"
    }
  ]
}
```

Para obtener más información, consulte [Ver las etiquetas de un recurso de conexiones](#) en la Guía del usuario de la consola Developer Tools.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

### AWS CLI

Para etiquetar un recurso

El siguiente tag-resource ejemplo asocia un conjunto de etiquetas proporcionadas a una conexión. Utilice este comando para añadir o editar etiquetas.

```
aws codestar-connections tag-resource \  
  --resource-arn arn:aws:codestar-connections:us-east-1:123456789012:connection/  
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f \  
  --tags Key=Project,Value=ProjectA Key=IscontainerBased,Value=true
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir etiquetas a un recurso de conexiones](#) en la Guía del usuario de la consola de Developer Tools.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar untag-resource.

### AWS CLI

Para eliminar AWS etiquetas de un recurso de conexiones

Lo siguiente untag-resource elimina una etiqueta del recurso especificado.

```
aws codestar-connections untag-resource \  
  --resource-arn arn:aws:codestar-connections:us-east-1:123456789012:connection/  
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f \  
  --tag-key ProjectA
```



```
--tag-keys Project ReadOnly
```

Salida:

```
{  
  "Tags": []  
}
```

Para obtener más información, consulte [Eliminar etiquetas de un recurso de conexiones](#) en la Guía del usuario de la consola de Developer Tools.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon Cognito Identity utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante Amazon Cognito Identity. AWS Command Line Interface

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-identity-pool**

En el siguiente ejemplo de código, se muestra cómo usar `create-identity-pool`.

AWS CLI

Para crear un grupo de identidades con el proveedor de grupos de identidades de Cognito

En este ejemplo se crea un grupo de identidades denominado `MyIdentityPool`. Tiene un proveedor de grupo de identidades de Cognito. No se permiten identidades no autenticadas.

Comando:

```
aws cognito-identity create-identity-pool --identity-pool-  
name MyIdentityPool --no-allow-unauthenticated-identities --cognito-  
identity-providers ProviderName="cognito-idp.us-west-2.amazonaws.com/us-  
west-2_aaaaaaaaa",ClientId="3n4b5urk1ft4fl3mg5e62d9ado",ServerSideTokenCheck=false
```

Salida:

```
{  
  "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",  
  "IdentityPoolName": "MyIdentityPool",  
  "AllowUnauthenticatedIdentities": false,  
  "CognitoIdentityProviders": [  
    {  
      "ProviderName": "cognito-idp.us-west-2.amazonaws.com/us-west-2_1111111111",  
      "ClientId": "3n4b5urk1ft4fl3mg5e62d9ado",  
      "ServerSideTokenCheck": false  
    }  
  ]  
}
```

- Para API obtener más información, consulte [CreateIdentityPool](#) la Referencia de AWS CLI comandos.

## delete-identities

En el siguiente ejemplo de código, se muestra cómo usar delete-identities.

AWS CLI

Para eliminar un grupo de identidades

En este ejemplo se elimina un grupo de identidades.

Comando:

```
aws cognito-identity delete-identity-pool --identity-ids-to-delete "us-  
west-2:11111111-1111-1111-1111-111111111111"
```

Salida:

```
{
  "UnprocessedIdentityIds": []
}
```

- Para API obtener más información, consulte [DeleteIdentities](#) la Referencia de AWS CLI comandos.

## delete-identity-pool

En el siguiente ejemplo de código, se muestra cómo usar `delete-identity-pool`.

### AWS CLI

Para eliminar un grupo de identidades

En el siguiente ejemplo de `delete-identity-pool` se elimina el grupo de identidades especificado.

Comando:

```
aws cognito-identity delete-identity-pool \
  --identity-pool-id "us-west-2:111111111-1111-1111-1111-111111111111"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteIdentityPool](#) la Referencia de AWS CLI comandos.

## describe-identity-pool

En el siguiente ejemplo de código, se muestra cómo usar `describe-identity-pool`.

### AWS CLI

Para describir un grupo de identidades

En este ejemplo se describe un grupo de identidades.

Comando:

```
aws cognito-identity describe-identity-pool --identity-pool-id "us-
west-2:111111111-1111-1111-1111-111111111111"
```

Salida:

```
{
  "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
  "IdentityPoolName": "MyIdentityPool",
  "AllowUnauthenticatedIdentities": false,
  "CognitoIdentityProviders": [
    {
      "ProviderName": "cognito-idp.us-west-2.amazonaws.com/us-west-2_1111111111",
      "ClientId": "3n4b5urk1ft4fl3mg5e62d9ado",
      "ServerSideTokenCheck": false
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeIdentityPool](#) la Referencia de AWS CLI comandos.

## get-identity-pool-roles

En el siguiente ejemplo de código, se muestra cómo usar `get-identity-pool-roles`.

AWS CLI

Para obtener los roles del grupo de identidades

En este ejemplo se obtienen los roles del grupo de identidades.

Comando:

```
aws cognito-identity get-identity-pool-roles --identity-pool-id "us-west-2:11111111-1111-1111-1111-111111111111"
```

Salida:

```
{
  "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
  "Roles": {
    "authenticated": "arn:aws:iam::111111111111:role/Cognito_MyIdentityPoolAuth_Role",
    "unauthenticated": "arn:aws:iam::111111111111:role/Cognito_MyIdentityPoolUnauth_Role"
  }
}
```

```
}  
}
```

- Para API obtener más información, consulte [GetIdentityPoolRoles](#) la Referencia de AWS CLI comandos.

## list-identity-pools

En el siguiente ejemplo de código, se muestra cómo usar `list-identity-pools`.

### AWS CLI

Para mostrar grupos de identidades

En este ejemplo, se muestran los grupos de identidades. Hay un máximo de 20 identidades en la lista.

Comando:

```
aws cognito-identity list-identity-pools --max-results 20
```

Salida:

```
{  
  "IdentityPools": [  
    {  
      "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",  
      "IdentityPoolName": "MyIdentityPool"  
    },  
    {  
      "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",  
      "IdentityPoolName": "AnotherIdentityPool"  
    },  
    {  
      "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",  
      "IdentityPoolName": "IdentityPoolRegionA"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListIdentityPools](#) la Referencia de AWS CLI comandos.

## set-identity-pool-roles

En el siguiente ejemplo de código, se muestra cómo usar `set-identity-pool-roles`.

### AWS CLI

Para establecer las funciones del grupo de identidades

El siguiente `set-identity-pool-roles` ejemplo establece un rol de grupo de identidades.

```
aws cognito-identity set-identity-pool-roles \  
  --identity-pool-id "us-west-2:11111111-1111-1111-1111-111111111111" \  
  --roles authenticated="arn:aws:iam::111111111111:role/  
Cognito_MyIdentityPoolAuth_Role"
```

- Para API obtener más información, consulte [SetIdentityPoolRoles](#) la Referencia de AWS CLI comandos.

## update-identity-pool

En el siguiente ejemplo de código, se muestra cómo usar `update-identity-pool`.

### AWS CLI

Para actualizar un grupo de identidades

En este ejemplo se actualiza un grupo de identidades. Establece el nombre en `MyIdentityPool`. Añade Cognito como proveedor de identidad. No permite las identidades no autenticadas.

Comando:

```
aws cognito-identity update-identity-pool --identity-pool-id "us-  
west-2:11111111-1111-1111-1111-111111111111" --identity-pool-  
name "MyIdentityPool" --no-allow-unauthenticated-identities --cognito-  
identity-providers ProviderName="cognito-idp.us-west-2.amazonaws.com/us-  
west-2_11111111",ClientId="3n4b5urk1ft4fl3mg5e62d9ado",ServerSideTokenCheck=false
```

Salida:

```
{  
  "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",  
  "IdentityPoolName": "MyIdentityPool",
```

```
"AllowUnauthenticatedIdentities": false,
"CognitoIdentityProviders": [
  {
    "ProviderName": "cognito-idp.us-west-2.amazonaws.com/us-west-2_111111111",
    "ClientId": "3n4b5urk1ft4f13mg5e62d9ado",
    "ServerSideTokenCheck": false
  }
]
}
```

- Para API obtener más información, consulte la Referencia de [UpdateIdentityPool](#) comandos AWS CLI .

## Ejemplos de Amazon Cognito Identity Provider que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante el AWS Command Line Interface uso de Amazon Cognito Identity Provider.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **add-custom-attributes**

En el siguiente ejemplo de código, se muestra cómo usar `add-custom-attributes`.

AWS CLI

Para añadir un atributo personalizado

En este ejemplo, se agrega un atributo personalizado `CustomAttr 1` a un grupo de usuarios. Es de tipo `String` y requiere un mínimo de 1 carácter y un máximo de 15. No es obligatorio.

Comando:

```
aws cognito-idp add-custom-attributes --user-pool-id us-west-2_aaaaaaaaa --custom-attributes Name="CustomAttr1",AttributeDataType="String",DeveloperOnlyAttribute=false,Required=false,S
```

- Para API obtener más información, consulte [AddCustomAttributes](#) la Referencia de AWS CLI comandos.

## **admim-disable-user**

En el siguiente ejemplo de código, se muestra cómo usar `admim-disable-user`.

AWS CLI

Para deshabilitar a un usuario

En este ejemplo, se deshabilita al usuario `jane@example.com`.

Comando:

```
aws cognito-idp admin-disable-user --user-pool-id us-west-2_aaaaaaaaa --username jane@example.com
```

- Para API obtener más información, consulte [AdminDisableUser](#) la Referencia de AWS CLI comandos.

## **admim-enable-user**

En el siguiente ejemplo de código, se muestra cómo usar `admim-enable-user`.

AWS CLI

Para habilitar un usuario

Este ejemplo habilita el nombre de usuario `jane@example.com`.

Comando:

```
aws cognito-idp admin-enable-user --user-pool-id us-west-2_aaaaaaaaa --username jane@example.com
```



- Para API obtener más información, consulte [AdminEnableUser](#) la Referencia de AWS CLI comandos.

## admin-add-user-to-group

En el siguiente ejemplo de código, se muestra cómo usar `admin-add-user-to-group`.

### AWS CLI

Para añadir un usuario a un grupo

En este ejemplo, se agrega la usuaria Jane al grupo MyGroup.

Comando:

```
aws cognito-idp admin-add-user-to-group --user-pool-id us-west-2_aaaaaaaaa --  
username Jane --group-name MyGroup
```

- Para API obtener más información, consulte [AdminAddUserToGroup](#) la Referencia de AWS CLI comandos.

## admin-confirm-sign-up

En el siguiente ejemplo de código, se muestra cómo usar `admin-confirm-sign-up`.

### AWS CLI

Para confirmar el registro del usuario

Este ejemplo confirma al usuario `jane@example.com`.

Comando:

```
aws cognito-idp admin-confirm-sign-up --user-pool-id us-west-2_aaaaaaaaa --  
username jane@example.com
```

- Para API obtener más información, consulte [AdminConfirmSignUp](#) la Referencia de AWS CLI comandos.

## admin-create-user

En el siguiente ejemplo de código, se muestra cómo usar `admin-create-user`.

### AWS CLI

Para crear un usuario

En el siguiente `admin-create-user` ejemplo, se crea un usuario con la dirección de correo electrónico y el número de teléfono especificados.

```
aws cognito-idp admin-create-user \  
  --user-pool-id us-west-2_aaaaaaaaa \  
  --username diego \  
  --user-attributes Name=email,Value=diego@example.com  
Name=phone_number,Value="+15555551212" \  
  --message-action SUPPRESS
```

Salida:

```
{  
  "User": {  
    "Username": "diego",  
    "Attributes": [  
      {  
        "Name": "sub",  
        "Value": "7325c1de-b05b-4f84-b321-9adc6e61f4a2"  
      },  
      {  
        "Name": "phone_number",  
        "Value": "+15555551212"  
      },  
      {  
        "Name": "email",  
        "Value": "diego@example.com"  
      }  
    ],  
    "UserCreateDate": 1548099495.428,  
    "UserLastModifiedDate": 1548099495.428,  
    "Enabled": true,  
    "UserStatus": "FORCE_CHANGE_PASSWORD"  
  }  
}
```

- Para API obtener más información, consulte [AdminCreateUser](#) la Referencia de AWS CLI comandos.

## admin-delete-user-attributes

En el siguiente ejemplo de código, se muestra cómo usar `admin-delete-user-attributes`.

### AWS CLI

Para eliminar un atributo de usuario

En este ejemplo, se elimina un atributo personalizado `CustomAttr 1` para el usuario `diego@example.com`.

Comando:

```
aws cognito-idp admin-delete-user-attributes --user-pool-id us-west-2_aaaaaaaaa --username diego@example.com --user-attribute-names "custom:CustomAttr1"
```

- Para API obtener más información, consulte [AdminDeleteUserAttributes](#) la Referencia de AWS CLI comandos.

## admin-delete-user

En el siguiente ejemplo de código, se muestra cómo usar `admin-delete-user`.

### AWS CLI

Para eliminar un usuario

En este ejemplo se elimina un usuario.

Comando:

```
aws cognito-idp admin-delete-user --user-pool-id us-west-2_aaaaaaaaa --username diego@example.com
```

- Para API obtener más información, consulte [AdminDeleteUser](#) la Referencia de AWS CLI comandos.

## admin-forget-device

En el siguiente ejemplo de código, se muestra cómo usar `admin-forget-device`.

### AWS CLI

Para olvidar un dispositivo

En este ejemplo se olvida el nombre de usuario `jane@example.com`

Comando:

```
aws cognito-idp admin-forget-device --user-pool-id us-west-2_aaaaaaaaa --  
username jane@example.com --device-key us-west-2_abcd_1234-5678
```

- Para API obtener más información, consulte [AdminForgetDevice](#) la Referencia de AWS CLI comandos.

## admin-get-device

En el siguiente ejemplo de código, se muestra cómo usar `admin-get-device`.

### AWS CLI

Para obtener un dispositivo

En este ejemplo se obtiene un dispositivo con el nombre de usuario `jane@example.com`

Comando:

```
aws cognito-idp admin-get-device --user-pool-id us-west-2_aaaaaaaaa --  
username jane@example.com --device-key us-west-2_abcd_1234-5678
```

- Para API obtener más información, consulte [AdminGetDevice](#) la Referencia de AWS CLI comandos.

## admin-get-user

En el siguiente ejemplo de código, se muestra cómo usar `admin-get-user`.

## AWS CLI

Para obtener un usuario

En este ejemplo se obtiene información sobre el nombre de usuario `jane@example.com`.

Comando:

```
aws cognito-idp admin-get-user --user-pool-id us-west-2_aaaaaaaaa --  
username jane@example.com
```

Salida:

```
{  
  "Username": "4320de44-2322-4620-999b-5e2e1c8df013",  
  "Enabled": true,  
  "UserStatus": "FORCE_CHANGE_PASSWORD",  
  "UserCreateDate": 1548108509.537,  
  "UserAttributes": [  
    {  
      "Name": "sub",  
      "Value": "4320de44-2322-4620-999b-5e2e1c8df013"  
    },  
    {  
      "Name": "email_verified",  
      "Value": "true"  
    },  
    {  
      "Name": "phone_number_verified",  
      "Value": "true"  
    },  
    {  
      "Name": "phone_number",  
      "Value": "+01115551212"  
    },  
    {  
      "Name": "email",  
      "Value": "jane@example.com"  
    }  
  ],  
  "UserLastModifiedDate": 1548108509.537  
}
```

- Para API obtener más información, consulte [AdminGetUser](#) la Referencia de AWS CLI comandos.

## admin-initiate-auth

En el siguiente ejemplo de código, se muestra cómo usar `admin-initiate-auth`.

### AWS CLI

Para iniciar la autorización

En este ejemplo, se inicia la autorización mediante el AUTH flujo ADMIN\_NO\_SRP \_ para el nombre de usuario `jane@example.com`

El cliente debe tener habilitado el inicio de sesión API para la autenticación basada en servidor (ADMIN\_NO\_). SRP AUTH

Use la información de la sesión en el valor devuelto para llamar a `-challenge`. `admin-respond-to-auth`

Comando:

```
aws cognito-idp admin-initiate-auth --user-pool-id us-west-2_aaaaaaaaa --
client-id 3n4b5urk1ft4fl3mg5e62d9ado --auth-flow ADMIN_NO_SRP_AUTH --auth-
parameters USERNAME=jane@example.com,PASSWORD=password
```

Salida:

```
{
  "ChallengeName": "NEW_PASSWORD_REQUIRED",
  "Session": "SESSION",
  "ChallengeParameters": {
    "USER_ID_FOR_SRP": "84514837-dcbc-4af1-abff-f3c109334894",
    "requiredAttributes": "[]",
    "userAttributes": "{\"email_verified\": \"true\", \"phone_number_verified\":
\"true\", \"phone_number\": \"+01xxx5550100\", \"email\": \"jane@example.com\"}"
  }
}
```

- Para API obtener más información, consulte [AdminInitiateAuth](#) la Referencia de AWS CLI comandos.

## admin-list-devices

En el siguiente ejemplo de código, se muestra cómo usar `admin-list-devices`.

### AWS CLI

Para enumerar los dispositivos de un usuario

En este ejemplo, se enumeran los dispositivos con el nombre de usuario `jane@example.com`.

Comando:

```
aws cognito-idp admin-list-devices --user-pool-id us-west-2_aaaaaaaaa --  
username jane@example.com
```

- Para API obtener más información, consulte [AdminListDevices](#) la Referencia de AWS CLI comandos.

## admin-list-groups-for-user

En el siguiente ejemplo de código, se muestra cómo usar `admin-list-groups-for-user`.

### AWS CLI

Para enumerar los grupos de un usuario

En este ejemplo, se muestran los grupos del nombre de usuario `jane@example.com`.

Comando:

```
aws cognito-idp admin-list-groups-for-user --user-pool-id us-west-2_aaaaaaaaa --  
username diego@example.com
```

Salida:

```
{  
  "Groups": [  
    {  
      "Description": "Sample group",  
      "Precedence": 1,  
      "LastModifiedDate": 1548097827.125,  
    }  
  ]  
}
```

```
    "RoleArn": "arn:aws:iam::111111111111:role/SampleRole",
    "GroupName": "SampleGroup",
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "CreationDate": 1548097827.125
  }
]
```

- Para API obtener más información, consulte [AdminListGroupForUser](#) la Referencia de AWS CLI comandos.

## admin-list-user-auth-events

En el siguiente ejemplo de código, se muestra cómo usar `admin-list-user-auth-events`.

### AWS CLI

Para enumerar los eventos de autorización de un usuario

En este ejemplo, se enumeran los eventos de autorización del nombre de usuario `diego@example.com`.

Comando:

```
aws cognito-idp admin-list-user-auth-events --user-pool-id us-west-2_aaaaaaaaa --
username diego@example.com
```

- Para API obtener más información, consulte [AdminListUserAuthEvents](#) la Referencia de AWS CLI comandos.

## admin-remove-user-from-group

En el siguiente ejemplo de código, se muestra cómo usar `admin-remove-user-from-group`.

### AWS CLI

Para eliminar un usuario de un grupo

En este ejemplo, se elimina `jane@example.com` de `SampleGroup`.

Comando:



```
aws cognito-idp admin-remove-user-from-group --user-pool-id us-west-2_aaaaaaaa --  
username jane@example.com --group-name SampleGroup
```

- Para API obtener más información, consulte [AdminRemoveUserFromGroup](#) la Referencia de AWS CLI comandos.

## admin-reset-user-password

En el siguiente ejemplo de código, se muestra cómo usar `admin-reset-user-password`.

### AWS CLI

Para restablecer una contraseña de usuario

En este ejemplo se restablece la contraseña de `diego@example.com`.

Comando:

```
aws cognito-idp admin-reset-user-password --user-pool-id us-west-2_aaaaaaaa --  
username diego@example.com
```

- Para API obtener más información, consulte [AdminResetUserPassword](#) la Referencia de AWS CLI comandos.

## admin-set-user-mfa-preference

En el siguiente ejemplo de código, se muestra cómo usar `admin-set-user-mfa-preference`.

### AWS CLI

Para establecer las MFA preferencias del usuario

En este ejemplo se establece la SMS MFA preferencia del nombre de usuario `diego@example.com`.

Comando:

```
aws cognito-idp admin-set-user-mfa-preference --user-pool-id us-west-2_aaaaaaaa --  
username diego@example.com --sms-mfa-settings Enabled=false,PreferredMfa=false
```

- Para API obtener más información, consulte [AdminSetUserMfaPreference](#) la Referencia de AWS CLI comandos.

## admin-set-user-settings

En el siguiente ejemplo de código, se muestra cómo usar `admin-set-user-settings`.

### AWS CLI

Para establecer la configuración de usuario

En este ejemplo, se establece la preferencia de MFA entrega del nombre de usuario `diego@example.com` en `EMAIL`.

Comando:

```
aws cognito-idp admin-set-user-settings --user-pool-id us-west-2_aaaaaaaa --  
username diego@example.com --mfa-options DeliveryMedium=EMAIL
```

- Para API obtener más información, consulte [AdminSetUserSettings](#) la Referencia de AWS CLI comandos.

## admin-update-auth-event-feedback

En el siguiente ejemplo de código, se muestra cómo usar `admin-update-auth-event-feedback`.

### AWS CLI

Para enviar comentarios sobre un evento de autorización

En este ejemplo, se establece el valor de los comentarios para un evento de autorización identificado por el identificador del evento en `Valid`.

Comando:

```
aws cognito-idp admin-update-auth-event-feedback --user-pool-id us-west-2_aaaaaaaa  
--username diego@example.com --event-id c2c2cf89-c0d3-482d-aba6-99d78a5b0bfe --  
feedback-value Valid
```

- Para API obtener más información, consulte la Referencia [AdminUpdateAuthEventFeedback](#) de AWS CLI comandos.

## admin-update-device-status

En el siguiente ejemplo de código, se muestra cómo usar `admin-update-device-status`.

### AWS CLI

Para actualizar el estado del dispositivo

En este ejemplo, se establece el estado de recuerdo del dispositivo identificado por la clave del dispositivo en `not_remember`.

Comando:

```
aws cognito-idp admin-update-device-status --user-pool-id us-west-2_aaaaaaaaa
--username diego@example.com --device-key xxxx --device-remembered-
status not_remembered
```

- Para obtener API más información, consulte la Referencia de comandos [AdminUpdateDeviceStatus](#).AWS CLI

## admin-update-user-attributes

En el siguiente ejemplo de código, se muestra cómo usar `admin-update-user-attributes`.

### AWS CLI

Para actualizar los atributos de usuario

En este ejemplo, se actualiza un atributo de usuario personalizado `CustomAttr 1` para el usuario `diego@example.com`.

Comando:

```
aws cognito-idp admin-update-user-attributes --user-pool-id us-
west-2_aaaaaaaaa --username diego@example.com --user-attributes
Name="custom:CustomAttr1",Value="Purple"
```

- Para API obtener más información, consulte [AdminUpdateUserAttributes](#) la Referencia de AWS CLI comandos.

## change-password

En el siguiente ejemplo de código, se muestra cómo usar `change-password`.

### AWS CLI

Para cambiar una contraseña

En este ejemplo se cambia una contraseña.

Comando:

```
aws cognito-idp change-password --previous-password OldPassword --proposed-  
password NewPassword --access-token ACCESS_TOKEN
```

- Para API obtener más información, consulte [ChangePassword](#) la Referencia de AWS CLI comandos.

## confirm-forgot-password

En el siguiente ejemplo de código, se muestra cómo usar `confirm-forgot-password`.

### AWS CLI

Para confirmar una contraseña olvidada

En este ejemplo se confirma una contraseña olvidada para el nombre de usuario `diego@example.com`.

Comando:

```
aws cognito-idp confirm-forgot-password --client-id 3n4b5urk1ft4fl3mg5e62d9ado --  
username=diego@example.com --password PASSWORD --confirmation-code CONF_CODE
```

- Para API obtener más información, consulte [ConfirmForgotPassword](#) la Referencia de AWS CLI comandos.

## confirm-sign-up

En el siguiente ejemplo de código, se muestra cómo usar `confirm-sign-up`.

## AWS CLI

Para confirmar la inscripción

Este ejemplo confirma el registro del nombre de usuario `diego@example.com`.

Comando:

```
aws cognito-idp confirm-sign-up --client-id 3n4b5urk1ft4fl3mg5e62d9ado --  
username=diego@example.com --confirmation-code CONF_CODE
```

- Para API obtener más información, consulte [ConfirmSignUp](#) la Referencia de AWS CLI comandos.

## create-group

En el siguiente ejemplo de código, se muestra cómo usar `create-group`.

## AWS CLI

Para crear un grupo

En este ejemplo se crea un grupo con una descripción.

Comando:

```
aws cognito-idp create-group --user-pool-id us-west-2_aaaaaaaaa --group-  
name MyNewGroup --description "New group."
```

Salida:

```
{  
  "Group": {  
    "GroupName": "MyNewGroup",  
    "UserPoolId": "us-west-2_aaaaaaaaa",  
    "Description": "New group.",  
    "LastModifiedDate": 1548270073.795,  
    "CreationDate": 1548270073.795  
  }  
}
```

Para crear un grupo con una función y una prioridad

En este ejemplo se crea un grupo con una descripción. También incluye una función y una prioridad.

Comando:

```
aws cognito-idp create-group --user-pool-id us-west-2_aaaaaaaaa --group-name MyNewGroupWithRole --description "New group with a role." --role-arn arn:aws:iam::111111111111:role/MyNewGroupRole --precedence 2
```

Salida:

```
{
  "Group": {
    "GroupName": "MyNewGroupWithRole",
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "Description": "New group with a role.",
    "RoleArn": "arn:aws:iam::111111111111:role/MyNewGroupRole",
    "Precedence": 2,
    "LastModifiedDate": 1548270211.761,
    "CreationDate": 1548270211.761
  }
}
```

- Para API obtener más información, consulte [CreateGroup](#) la Referencia de AWS CLI comandos.

## **create-user-import-job**

En el siguiente ejemplo de código, se muestra cómo usar `create-user-import-job`.

### AWS CLI

Para crear un trabajo de importación de usuarios

En este ejemplo se crea un trabajo de importación de usuarios denominado `MyImportJob`.

Para obtener más información sobre la importación de usuarios, consulte [Importación de usuarios a grupos de usuarios desde un CSV archivo](#).

Comando:

```
aws cognito-idp create-user-import-job --user-pool-id us-west-2_aaaaaaaaaa --
job-name MyImportJob --cloud-watch-logs-role-arn arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole
```

Salida:

```
{
  "UserImportJob": {
    "JobName": "MyImportJob",
    "JobId": "import-qQ0DCt2fRh",
    "UserPoolId": "us-west-2_aaaaaaaaaa",
    "PreSignedUrl": "PRE_SIGNED_URL",
    "CreationDate": 1548271795.471,
    "Status": "Created",
    "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
    "ImportedUsers": 0,
    "SkippedUsers": 0,
    "FailedUsers": 0
  }
}
```

Cargue el archivo.csv con curl utilizando la siguiente información prefirmada: URL

Comando:

```
curl -v -T "PATH_TO_CSV_FILE" -H "x-amz-server-side-encryption:aws:kms"
"PRE_SIGNED_URL"
```

- Para API obtener más información, consulte la Referencia de [CreateUserImportJob](#) comandos AWS CLI .

## create-user-pool-client

En el siguiente ejemplo de código, se muestra cómo usar create-user-pool-client.

AWS CLI

Para crear un cliente de grupo de usuarios

En este ejemplo, se crea un nuevo cliente de grupo de usuarios con dos flujos de autorización explícitos: USER \_ PASSWORD \_ AUTH y ADMIN \_ NO \_ SRP. AUTH

Comando:

```
aws cognito-idp create-user-pool-client --user-pool-id us-west-2_aaaaaaaaa
--client-name MyNewClient --no-generate-secret --explicit-auth-
flows "USER_PASSWORD_AUTH" "ADMIN_NO_SRP_AUTH"
```

Salida:

```
{
  "UserPoolClient": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "ClientName": "MyNewClient",
    "ClientId": "6p3bs000no6a4ue1idruvd05ad",
    "LastModifiedDate": 1548697449.497,
    "CreationDate": 1548697449.497,
    "RefreshTokenValidity": 30,
    "ExplicitAuthFlows": [
      "USER_PASSWORD_AUTH",
      "ADMIN_NO_SRP_AUTH"
    ],
    "AllowedOAuthFlowsUserPoolClient": false
  }
}
```

- Para API obtener más información, consulte la Referencia [CreateUserPoolClient](#) de AWS CLI comandos.

## create-user-pool-domain

En el siguiente ejemplo de código, se muestra cómo usar `create-user-pool-domain`.

AWS CLI

Para crear un dominio de grupo de usuarios

En este ejemplo, se crea un nuevo dominio de grupo de usuarios con dos flujos de autorización explícitos: `USER_PASSWORD_AUTH` y `ADMIN SRP_NO_._. AUTH`

Comando:

```
aws cognito-idp create-user-pool-domain --user-pool-id us-west-2_aaaaaaaaa --
domain my-new-domain
```



- Para API obtener más información, consulte la Referencia de [CreateUserPoolDomain](#) comandos AWS CLI .

## create-user-pool

En el siguiente ejemplo de código, se muestra cómo usar `create-user-pool`.

### AWS CLI

Para crear un grupo de usuarios con una configuración mínima

En este ejemplo, se crea un grupo de usuarios denominado `MyUserPool` con los valores predeterminados. No se requieren atributos ni clientes de aplicación. MFA y la seguridad avanzada está deshabilitada.

Comando:

```
aws cognito-idp create-user-pool --pool-name MyUserPool
```

Salida:

```
{
  "UserPool": {
    "SchemaAttributes": [
      {
        "Name": "sub",
        "StringAttributeConstraints": {
          "MinLength": "1",
          "MaxLength": "2048"
        },
        "DeveloperOnlyAttribute": false,
        "Required": true,
        "AttributeDataType": "String",
        "Mutable": false
      },
      {
        "Name": "name",
        "StringAttributeConstraints": {
          "MinLength": "0",
          "MaxLength": "2048"
        },
        "DeveloperOnlyAttribute": false,
```

```
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "given_name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "family_name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "middle_name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "nickname",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
```

```
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "preferred_username",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "profile",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "picture",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "website",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
```

```
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "email",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "AttributeDataType": "Boolean",
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "Name": "email_verified",
    "Mutable": true
  },
  {
    "Name": "gender",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "birthdate",
    "StringAttributeConstraints": {
      "MinLength": "10",
      "MaxLength": "10"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  },
```

```
{
  "Name": "zoneinfo",
  "StringAttributeConstraints": {
    "MinLength": "0",
    "MaxLength": "2048"
  },
  "DeveloperOnlyAttribute": false,
  "Required": false,
  "AttributeDataType": "String",
  "Mutable": true
},
{
  "Name": "locale",
  "StringAttributeConstraints": {
    "MinLength": "0",
    "MaxLength": "2048"
  },
  "DeveloperOnlyAttribute": false,
  "Required": false,
  "AttributeDataType": "String",
  "Mutable": true
},
{
  "Name": "phone_number",
  "StringAttributeConstraints": {
    "MinLength": "0",
    "MaxLength": "2048"
  },
  "DeveloperOnlyAttribute": false,
  "Required": false,
  "AttributeDataType": "String",
  "Mutable": true
},
{
  "AttributeDataType": "Boolean",
  "DeveloperOnlyAttribute": false,
  "Required": false,
  "Name": "phone_number_verified",
  "Mutable": true
},
{
  "Name": "address",
  "StringAttributeConstraints": {
    "MinLength": "0",
```

```

        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "updated_at",
    "NumberAttributeConstraints": {
        "MinValue": "0"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "Number",
    "Mutable": true
}
],
"MfaConfiguration": "OFF",
"Name": "MyUserPool",
"LastModifiedDate": 1547833345.777,
"AdminCreateUserConfig": {
    "UnusedAccountValidityDays": 7,
    "AllowAdminCreateUserOnly": false
},
"EmailConfiguration": {},
"Policies": {
    "PasswordPolicy": {
        "RequireLowercase": true,
        "RequireSymbols": true,
        "RequireNumbers": true,
        "MinimumLength": 8,
        "RequireUppercase": true
    }
},
"CreationDate": 1547833345.777,
"EstimatedNumberOfUsers": 0,
"Id": "us-west-2_aaaaaaaaa",
"LambdaConfig": {}
}
}

```

## Creación de un grupo de usuarios con dos atributos obligatorios

En este ejemplo se crea un grupo de usuarios MyUserPool. El grupo está configurado para aceptar un correo electrónico como atributo de nombre de usuario. También establece la dirección de origen del correo electrónico en una dirección validada mediante Amazon Simple Email Service.

Comando:

```
aws cognito-idp create-user-pool --pool-name MyUserPool --username-attributes "email" --email-configuration=SourceArn="arn:aws:ses:us-east-1:111111111111:identity/jane@example.com",ReplyToEmailAddress="jane@example.com"
```

Salida:

```
{
  "UserPool": {
    "SchemaAttributes": [
      {
        "Name": "sub",
        "StringAttributeConstraints": {
          "MinLength": "1",
          "MaxLength": "2048"
        },
        "DeveloperOnlyAttribute": false,
        "Required": true,
        "AttributeDataType": "String",
        "Mutable": false
      },
      {
        "Name": "name",
        "StringAttributeConstraints": {
          "MinLength": "0",
          "MaxLength": "2048"
        },
        "DeveloperOnlyAttribute": false,
        "Required": false,
        "AttributeDataType": "String",
        "Mutable": true
      },
      {
        "Name": "given_name",
        "StringAttributeConstraints": {
          "MinLength": "0",
```

```
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "family_name",
    "StringAttributeConstraints": {
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "middle_name",
    "StringAttributeConstraints": {
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "nickname",
    "StringAttributeConstraints": {
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "preferred_username",
    "StringAttributeConstraints": {
        "MinLength": "0",
```



```
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "profile",
    "StringAttributeConstraints": {
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "picture",
    "StringAttributeConstraints": {
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "website",
    "StringAttributeConstraints": {
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "email",
    "StringAttributeConstraints": {
        "MinLength": "0",
```

```
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "AttributeDataType": "Boolean",
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "Name": "email_verified",
    "Mutable": true
},
{
    "Name": "gender",
    "StringAttributeConstraints": {
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "birthdate",
    "StringAttributeConstraints": {
        "MinLength": "10",
        "MaxLength": "10"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
},
{
    "Name": "zoneinfo",
    "StringAttributeConstraints": {
        "MinLength": "0",
        "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
```

```
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "locale",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "phone_number",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "AttributeDataType": "Boolean",
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "Name": "phone_number_verified",
    "Mutable": true
  },
  {
    "Name": "address",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
```

```

        "Name": "updated_at",
        "NumberAttributeConstraints": {
            "MinValue": "0"
        },
        "DeveloperOnlyAttribute": false,
        "Required": false,
        "AttributeDataType": "Number",
        "Mutable": true
    }
],
"MfaConfiguration": "OFF",
"Name": "MyUserPool",
"LastModifiedDate": 1547837788.189,
"AdminCreateUserConfig": {
    "UnusedAccountValidityDays": 7,
    "AllowAdminCreateUserOnly": false
},
"EmailConfiguration": {
    "ReplyToEmailAddress": "jane@example.com",
    "SourceArn": "arn:aws:ses:us-east-1:111111111111:identity/
jane@example.com"
},
"Policies": {
    "PasswordPolicy": {
        "RequireLowercase": true,
        "RequireSymbols": true,
        "RequireNumbers": true,
        "MinimumLength": 8,
        "RequireUppercase": true
    }
},
"UsernameAttributes": [
    "email"
],
"CreationDate": 1547837788.189,
"EstimatedNumberOfUsers": 0,
"Id": "us-west-2_aaaaaaaaaa",
"LambdaConfig": {}
}
}

```

- Para API obtener más información, consulte [CreateUserPool](#) la Referencia de AWS CLI comandos.

## delete-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-group`.

### AWS CLI

Para eliminar un grupo

En este ejemplo se elimina un grupo.

Comando:

```
aws cognito-idp delete-group --user-pool-id us-west-2_aaaaaaaaa --group-name MyGroupName
```

- Para API obtener más información, consulte [DeleteGroup](#) la Referencia de AWS CLI comandos.

## delete-identity-provider

En el siguiente ejemplo de código, se muestra cómo usar `delete-identity-provider`.

### AWS CLI

Para eliminar un proveedor de identidad

En este ejemplo, se elimina un proveedor de identidad.

Comando:

```
aws cognito-idp delete-identity-provider --user-pool-id us-west-2_aaaaaaaaa --provider-name Facebook
```

- Para API obtener más información, consulte [DeleteIdentityProvider](#) la Referencia de AWS CLI comandos.

## delete-resource-server

En el siguiente ejemplo de código, se muestra cómo usar `delete-resource-server`.

### AWS CLI

Para eliminar un servidor de recursos

En este ejemplo, se elimina un servidor de recursos denominado `weather.example.com`.

Comando:

```
aws cognito-idp delete-resource-server --user-pool-id us-west-2_aaaaaaaaa --  
identifier weather.example.com
```

- Para obtener API más información, consulte la Referencia de comandos [DeleteResourceServer](#).AWS CLI

## **delete-user-attributes**

En el siguiente ejemplo de código, se muestra cómo usar `delete-user-attributes`.

AWS CLI

Para eliminar los atributos del usuario

En este ejemplo, se elimina el atributo de usuario "FAVORITE\_ANIMAL».

Comando:

```
aws cognito-idp delete-user-attributes --access-token ACCESS_TOKEN --user-attribute-  
names "FAVORITE_ANIMAL"
```

- Para API obtener más información, consulte [DeleteUserAttributes](#)la Referencia de AWS CLI comandos.

## **delete-user-pool-client**

En el siguiente ejemplo de código, se muestra cómo usar `delete-user-pool-client`.

AWS CLI

Para eliminar un cliente de grupo de usuarios

En este ejemplo, se elimina un cliente de grupo de usuarios.

Comando:

```
aws cognito-idp delete-user-pool-client --user-pool-id us-west-2_aaaaaaaaa --client-id 38fjsnc484p94kpbsnet7mpld0
```

- Para API obtener más información, consulte [DeleteUserPoolClient](#) la Referencia de AWS CLI comandos.

## delete-user-pool-domain

En el siguiente ejemplo de código, se muestra cómo usar `delete-user-pool-domain`.

### AWS CLI

Para eliminar un dominio de grupo de usuarios

En el siguiente `delete-user-pool-domain` ejemplo, se elimina un dominio de grupo de usuarios denominado `my-domain`

```
aws cognito-idp delete-user-pool-domain \  
  --user-pool-id us-west-2_aaaaaaaaa \  
  --domain my-domain
```

- Para API obtener más información, consulte [DeleteUserPoolDomain](#) la Referencia de AWS CLI comandos.

## delete-user-pool

En el siguiente ejemplo de código, se muestra cómo usar `delete-user-pool`.

### AWS CLI

Para eliminar un grupo de usuarios

En este ejemplo, se elimina un grupo de usuarios mediante el identificador del grupo de usuarios, `us-west-2_aaaaaaaaa`.

Comando:

```
aws cognito-idp delete-user-pool --user-pool-id us-west-2_aaaaaaaaa
```

- Para obtener más información, consulte la Referencia de comandos. API [DeleteUserPool](#) AWS CLI

## delete-user

En el siguiente ejemplo de código, se muestra cómo usar `delete-user`.

### AWS CLI

Para eliminar un usuario

En este ejemplo se elimina un usuario.

Comando:

```
aws cognito-idp delete-user --access-token ACCESS_TOKEN
```

- Para API obtener más información, consulte [DeleteUser](#) la Referencia de AWS CLI comandos.

## describe-identity-provider

En el siguiente ejemplo de código, se muestra cómo usar `describe-identity-provider`.

### AWS CLI

Para describir un proveedor de identidad

En este ejemplo, se describe un proveedor de identidad denominado Facebook.

Comando:

```
aws cognito-idp describe-identity-provider --user-pool-id us-west-2_aaaaaaaaa --  
provider-name Facebook
```

Salida:

```
{  
  "IdentityProvider": {  
    "UserPoolId": "us-west-2_aaaaaaaaa",  
    "ProviderName": "Facebook",  
    "ProviderType": "Facebook",  
    "ProviderDetails": {  
      "attributes_url": "https://graph.facebook.com/me?fields=",  
      "attributes_url_add_attributes": "true",  
      "authorize_scopes": "myscope",  
      "authorize_url": "https://www.facebook.com/v2.9/dialog/oauth",
```



```

    "client_id": "11111",
    "client_secret": "11111",
    "token_request_method": "GET",
    "token_url": "https://graph.facebook.com/v2.9/oauth/access_token"
  },
  "AttributeMapping": {
    "username": "id"
  },
  "IdpIdentifiers": [],
  "LastModifiedDate": 1548105901.736,
  "CreationDate": 1548105901.736
}
}

```

- Para API obtener más información, consulte [DescribeIdentityProvider](#) la Referencia de AWS CLI comandos.

## describe-resource-server

En el siguiente ejemplo de código, se muestra cómo usar `describe-resource-server`.

### AWS CLI

Para describir un servidor de recursos

En este ejemplo se describe el servidor de recursos `weather.example.com`.

Comando:

```
aws cognito-idp describe-resource-server --user-pool-id us-west-2_aaaaaaaaa --
identifier weather.example.com
```

Salida:

```

{
  "ResourceServer": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "Identifier": "weather.example.com",
    "Name": "Weather",
    "Scopes": [
      {
        "ScopeName": "weather.update",
        "ScopeDescription": "Update weather forecast"
      }
    ]
  }
}

```

```

    },
    {
      "ScopeName": "weather.read",
      "ScopeDescription": "Read weather forecasts"
    },
    {
      "ScopeName": "weather.delete",
      "ScopeDescription": "Delete a weather forecast"
    }
  ]
}
}

```

- Para API obtener más información, consulte la Referencia de [DescribeResourceServer](#) comandos AWS CLI .

## describe-risk-configuration

En el siguiente ejemplo de código, se muestra cómo usar `describe-risk-configuration`.

### AWS CLI

Para describir una configuración de riesgo

En este ejemplo se describe la configuración de riesgo asociada a la piscina `us-west-2_aaaaaaaaaa`.

Comando:

```
aws cognito-idp describe-risk-configuration --user-pool-id us-west-2_aaaaaaaaaa
```

Salida:

```

{
  "RiskConfiguration": {
    "UserPoolId": "us-west-2_aaaaaaaaaa",
    "CompromisedCredentialsRiskConfiguration": {
      "EventFilter": [
        "SIGN_IN",
        "SIGN_UP",
        "PASSWORD_CHANGE"
      ],

```

```

    "Actions": {
      "EventAction": "BLOCK"
    }
  },
  "AccountTakeoverRiskConfiguration": {
    "NotifyConfiguration": {
      "From": "diego@example.com",
      "ReplyTo": "diego@example.com",
      "SourceArn": "arn:aws:ses:us-east-1:111111111111:identity/
diego@example.com",
      "BlockEmail": {
        "Subject": "Blocked sign-in attempt",
        "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML
email context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We
blocked an unrecognized sign-in to your account with this information:\n<ul>
\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},
{country}</li>\n</ul>\nIf this sign-in was not by you, you should change your
password and notify us by clicking on <a href={one-click-link-invalid}>this link</
a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this
link</a> to let us know</pre>\n</body>\n</html>",
        "TextBody": "We blocked an unrecognized sign-in to your account
with this information:\nTime: {login-time}\nDevice: {device-name}\nLocation:
{city}, {country}\nIf this sign-in was not by you, you should change your password
and notify us by clicking on {one-click-link-invalid}\nIf this sign-in was by you,
you can follow {one-click-link-valid} to let us know"
      },
      "NoActionEmail": {
        "Subject": "New sign-in attempt",
        "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML
email context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We
observed an unrecognized sign-in to your account with this information:\n<ul>
\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},
{country}</li>\n</ul>\nIf this sign-in was not by you, you should change your
password and notify us by clicking on <a href={one-click-link-invalid}>this link</
a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this
link</a> to let us know</pre>\n</body>\n</html>",
        "TextBody": "We observed an unrecognized sign-in to your account
with this information:\nTime: {login-time}\nDevice: {device-name}\nLocation:
{city}, {country}\nIf this sign-in was not by you, you should change your password
and notify us by clicking on {one-click-link-invalid}\nIf this sign-in was by you,
you can follow {one-click-link-valid} to let us know"
      },
      "MfaEmail": {
        "Subject": "New sign-in attempt",

```

```
        "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML email\ncontext</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We required\nyou to use multi-factor authentication for the following sign-in attempt:\n<ul>\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},\n{country}</li>\n</ul>\nIf this sign-in was not by you, you should change your\npassword and notify us by clicking on <a href={one-click-link-invalid}>this link</\na>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this\nlink</a> to let us know</pre>\n</body>\n</html>",\n        "TextBody": "We required you to use multi-factor authentication\nfor the following sign-in attempt:\nTime: {login-time}\nDevice: {device-\nname}\nLocation: {city}, {country}\nIf this sign-in was not by you, you should\nchange your password and notify us by clicking on {one-click-link-invalid}\nIf this\nsign-in was by you, you can follow {one-click-link-valid} to let us know"\n    }\n},\n    "Actions": {\n        "LowAction": {\n            "Notify": true,\n            "EventAction": "NO_ACTION"\n        },\n        "MediumAction": {\n            "Notify": true,\n            "EventAction": "MFA_IF_CONFIGURED"\n        },\n        "HighAction": {\n            "Notify": true,\n            "EventAction": "MFA_IF_CONFIGURED"\n        }\n    }\n}\n}\n}
```

- Para API obtener más información, consulte la Referencia de comandos.

[DescribeRiskConfiguration](#) AWS CLI

## describe-user-import-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-user-import-job`.

AWS CLI

Para describir un trabajo de importación de usuarios

En este ejemplo se describe un trabajo de entrada de usuario.

Para obtener más información sobre la importación de usuarios, consulte [Importación de usuarios a grupos de usuarios desde un CSV archivo](#).

Comando:

```
aws cognito-idp describe-user-import-job --user-pool-id us-west-2_aaaaaaaaa --job-id import-TZqNQvDRnW
```

Salida:

```
{
  "UserImportJob": {
    "JobName": "import-Test1",
    "JobId": "import-TZqNQvDRnW",
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "PreSignedUrl": "PRE_SIGNED URL",
    "CreationDate": 1548271708.512,
    "Status": "Created",
    "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
    "ImportedUsers": 0,
    "SkippedUsers": 0,
    "FailedUsers": 0
  }
}
```

- Para API obtener más información, consulte [DescribeUserImportJob](#) la Referencia de AWS CLI comandos.

## describe-user-pool-client

En el siguiente ejemplo de código, se muestra cómo usar `describe-user-pool-client`.

AWS CLI

Para describir un cliente de grupo de usuarios

En este ejemplo se describe un cliente de grupo de usuarios.

Comando:

```
aws cognito-idp describe-user-pool-client --user-pool-id us-west-2_aaaaaaaaa --  
client-id 38fjsnc484p94kpbsnet7mpld0
```

Salida:

```
{  
  "UserPoolClient": {  
    "UserPoolId": "us-west-2_aaaaaaaaa",  
    "ClientName": "MyApp",  
    "ClientId": "38fjsnc484p94kpbsnet7mpld0",  
    "ClientSecret": "CLIENT_SECRET",  
    "LastModifiedDate": 1548108676.163,  
    "CreationDate": 1548108676.163,  
    "RefreshTokenValidity": 30,  
    "ReadAttributes": [  
      "address",  
      "birthdate",  
      "custom:CustomAttr1",  
      "custom:CustomAttr2",  
      "email",  
      "email_verified",  
      "family_name",  
      "gender",  
      "given_name",  
      "locale",  
      "middle_name",  
      "name",  
      "nickname",  
      "phone_number",  
      "phone_number_verified",  
      "picture",  
      "preferred_username",  
      "profile",  
      "updated_at",  
      "website",  
      "zoneinfo"  
    ],  
    "WriteAttributes": [  
      "address",  
      "birthdate",  
      "custom:CustomAttr1",  
      "custom:CustomAttr2",  
      "email",
```

```
    "family_name",
    "gender",
    "given_name",
    "locale",
    "middle_name",
    "name",
    "nickname",
    "phone_number",
    "picture",
    "preferred_username",
    "profile",
    "updated_at",
    "website",
    "zoneinfo"
  ],
  "ExplicitAuthFlows": [
    "ADMIN_NO_SRP_AUTH",
    "USER_PASSWORD_AUTH"
  ],
  "AllowedOAuthFlowsUserPoolClient": false
}
}
```

- Para API obtener más información, consulte [DescribeUserPoolClient](#) la Referencia de AWS CLI comandos.

## describe-user-pool-domain

En el siguiente ejemplo de código, se muestra cómo usar `describe-user-pool-domain`.

### AWS CLI

Para describir un cliente de grupo de usuarios

En este ejemplo, se describe un dominio de grupo de usuarios denominado `my-domain`.

Comando:

```
aws cognito-idp describe-user-pool-domain --domain my-domain
```

Salida:

```
{
```

```
"DomainDescription": {
  "UserPoolId": "us-west-2_aaaaaaaaa",
  "AWSAccountId": "111111111111",
  "Domain": "my-domain",
  "S3Bucket": "aws-cognito-prod-pdx-assets",
  "CloudFrontDistribution": "aaaaaaaaaaaaa.cloudfront.net",
  "Version": "20190128175402",
  "Status": "ACTIVE",
  "CustomDomainConfig": {}
}
```

- Para API obtener más información, consulte [DescribeUserPoolDomain](#) la Referencia de AWS CLI comandos.

## describe-user-pool

En el siguiente ejemplo de código, se muestra cómo usar `describe-user-pool`.

### AWS CLI

Para describir un grupo de usuarios

En este ejemplo se describe un grupo de usuarios con el identificador de grupo de usuarios `us-west-2_aaaaaaaaa`.

Comando:

```
aws cognito-idp describe-user-pool --user-pool-id us-west-2_aaaaaaaaa
```

Salida:

```
{
  "UserPool": {
    "SmsVerificationMessage": "Your verification code is {####}. ",
    "SchemaAttributes": [
      {
        "Name": "sub",
        "StringAttributeConstraints": {
          "MinLength": "1",
          "MaxLength": "2048"
        }
      }
    ],
  },
}
```



```
    "DeveloperOnlyAttribute": false,
    "Required": true,
    "AttributeDataType": "String",
    "Mutable": false
  },
  {
    "Name": "name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "given_name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "family_name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "middle_name",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
  },
```

```
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "nickname",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "preferred_username",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "profile",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "picture",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
  },
```

```
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "website",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "email",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": true,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "AttributeDataType": "Boolean",
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "Name": "email_verified",
    "Mutable": true
  },
  {
    "Name": "gender",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  }
}
```

```
  },
  {
    "Name": "birthdate",
    "StringAttributeConstraints": {
      "MinLength": "10",
      "MaxLength": "10"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "zoneinfo",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "locale",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  },
  {
    "Name": "phone_number",
    "StringAttributeConstraints": {
      "MinLength": "0",
      "MaxLength": "2048"
    },
    "DeveloperOnlyAttribute": false,
    "Required": false,
    "AttributeDataType": "String",
    "Mutable": true
  }
}
```

```

    },
    {
      "AttributeDataType": "Boolean",
      "DeveloperOnlyAttribute": false,
      "Required": false,
      "Name": "phone_number_verified",
      "Mutable": true
    },
    {
      "Name": "address",
      "StringAttributeConstraints": {
        "MinLength": "0",
        "MaxLength": "2048"
      },
      "DeveloperOnlyAttribute": false,
      "Required": false,
      "AttributeDataType": "String",
      "Mutable": true
    },
    {
      "Name": "updated_at",
      "NumberAttributeConstraints": {
        "MinValue": "0"
      },
      "DeveloperOnlyAttribute": false,
      "Required": false,
      "AttributeDataType": "Number",
      "Mutable": true
    }
  ],
  "EmailVerificationSubject": "Your verification code",
  "MfaConfiguration": "OFF",
  "Name": "MyUserPool",
  "EmailVerificationMessage": "Your verification code is {#####}. ",
  "SmsAuthenticationMessage": "Your authentication code is {#####}. ",
  "LastModifiedDate": 1547763720.822,
  "AdminCreateUserConfig": {
    "InviteMessageTemplate": {
      "EmailMessage": "Your username is {username} and temporary password is
#####. ",
      "EmailSubject": "Your temporary password",
      "SMSMessage": "Your username is {username} and temporary password is
#####. "
    },

```

```
    "UnusedAccountValidityDays": 7,
    "AllowAdminCreateUserOnly": false
  },
  "EmailConfiguration": {
    "ReplyToEmailAddress": "myemail@mydomain.com"
    "SourceArn": "arn:aws:ses:us-east-1:000000000000:identity/
myemail@mydomain.com"
  },
  "AutoVerifiedAttributes": [
    "email"
  ],
  "Policies": {
    "PasswordPolicy": {
      "RequireLowercase": true,
      "RequireSymbols": true,
      "RequireNumbers": true,
      "MinimumLength": 8,
      "RequireUppercase": true
    }
  },
  "UserPoolTags": {},
  "UsernameAttributes": [
    "email"
  ],
  "CreationDate": 1547763720.822,
  "EstimatedNumberOfUsers": 1,
  "Id": "us-west-2_aaaaaaaaaa",
  "LambdaConfig": {}
}
}
```

- Para API obtener más información, consulte la Referencia de comandos.

[DescribeUserPool](#)AWS CLI

## forget-device

En el siguiente ejemplo de código, se muestra cómo usar forget-device.

AWS CLI

Para olvidar un dispositivo

En este ejemplo se olvida de un dispositivo por otro.

Comando:

```
aws cognito-idp forget-device --device-key us-west-2_abcd_1234-5678
```

- Para API obtener más información, consulte [ForgetDevice](#) la Referencia de AWS CLI comandos.

## forgot-password

En el siguiente ejemplo de código, se muestra cómo usar forgot-password.

AWS CLI

Para forzar un cambio de contraseña

En el siguiente forgot-password ejemplo, se envía un mensaje a jane@example.com para cambiar su contraseña.

```
aws cognito-idp forgot-password --client-id 38fjsnc484p94kpbsnet7mpld0 --  
username jane@example.com
```

Salida:

```
{  
  "CodeDeliveryDetails": {  
    "Destination": "j***@e***.com",  
    "DeliveryMedium": "EMAIL",  
    "AttributeName": "email"  
  }  
}
```

- Para API obtener más información, consulte [ForgotPassword](#) la Referencia de AWS CLI comandos.

## get-csv-header

En el siguiente ejemplo de código, se muestra cómo usar get-csv-header.

AWS CLI

Para crear un encabezado csv

En este ejemplo, se crea un encabezado csv.

Para obtener más información sobre la importación de usuarios, consulte [Importación de usuarios a grupos de usuarios desde un CSV archivo](#).

Comando:

```
aws cognito-idp get-csv-header --user-pool-id us-west-2_aaaaaaaa
```

Salida:

```
{
  "UserPoolId": "us-west-2_aaaaaaaa",
  "CSVHeader": [
    "name",
    "given_name",
    "family_name",
    "middle_name",
    "nickname",
    "preferred_username",
    "profile",
    "picture",
    "website",
    "email",
    "email_verified",
    "gender",
    "birthdate",
    "zoneinfo",
    "locale",
    "phone_number",
    "phone_number_verified",
    "address",
    "updated_at",
    "cognito:mfa_enabled",
    "cognito:username"
  ]
}
```

... Importación de usuarios a grupos de usuarios desde un CSV archivo: <https://docs.aws.amazon.com/cognito/cognito-user-pools-using-latest/developerguide/-import-tool.html>

- Para obtener API más información, consulte la Referencia de comandos. [GetCsvHeaderAWS CLI](#)



## get-group

En el siguiente ejemplo de código, se muestra cómo usar `get-group`.

### AWS CLI

Para obtener información sobre un grupo

En este ejemplo se obtiene información sobre un grupo denominado `MyGroup`.

Comando:

```
aws cognito-idp get-group --user-pool-id us-west-2_aaaaaaaaa --group-name MyGroup
```

Salida:

```
{
  "Group": {
    "GroupName": "MyGroup",
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "Description": "A sample group.",
    "LastModifiedDate": 1548270073.795,
    "CreationDate": 1548270073.795
  }
}
```

- Para API obtener más información, consulte [GetGroup](#) la Referencia de AWS CLI comandos.

## get-signing-certificate

En el siguiente ejemplo de código, se muestra cómo usar `get-signing-certificate`.

### AWS CLI

Para obtener un certificado de firma

En este ejemplo, se obtiene un certificado de firma para un grupo de usuarios.

Comando:

```
aws cognito-idp get-signing-certificate --user-pool-id us-west-2_aaaaaaaaa
```

Salida:

```
{
  "Certificate": "CERTIFICATE_DATA"
}
```

- Para API obtener más información, consulte [GetSigningCertificate](#) la Referencia de AWS CLI comandos.

## get-ui-customization

En el siguiente ejemplo de código, se muestra cómo usar `get-ui-customization`.

### AWS CLI

Para obtener información sobre la personalización de la interfaz de usuario

En este ejemplo, se obtiene información de personalización de la interfaz de usuario de un grupo de usuarios.

Comando:

```
aws cognito-idp get-ui-customization --user-pool-id us-west-2_aaaaaaaaa
```

Salida:

```
{
  "UICustomization": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "ClientId": "ALL",
    "ImageUrl": "https://aaaaaaaaaaaaa.cloudfront.net/us-west-2_aaaaaaaaa/
ALL/20190128231240/assets/images/image.jpg",
    "CSS": ".logo-customizable {\n\tmax-width: 60%;\n\tmax-height: 30%;
\n}\n\n.banner-customizable {\n\tpadding: 25px 0px 25px 10px;\n\tbackground-color:
lightgray;\n}\n\n.label-customizable {\n\tfont-weight: 300;\n}\n\n.textDescription-
customizable {\n\tpadding-top: 10px;\n\tpadding-bottom: 10px;\n\tdisplay: block;
\n\tfont-size: 16px;\n}\n\n.idpDescription-customizable {\n\tpadding-top: 10px;\n
\tpadding-bottom: 10px;\n\tdisplay: block;\n\tfont-size: 16px;\n}\n\n.legalText-
customizable {\n\tcolor: #747474;\n\tfont-size: 11px;\n}\n\n.submitButton-customizable
{\n\tfont-size: 14px;\n\tfont-weight: bold;\n\tmargin: 20px 0px 10px 0px;\n
\theight: 40px;\n\twidth: 100%;\n\tcolor: #fff;\n\tbackground-color: #337ab7;
\n}\n\n.submitButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color:
#286090;\n}\n\n.errorMessage-customizable {\n\tpadding: 5px;\n\tfont-size: 14px;
\n\twidth: 100%;\n\tbackground: #F5F5F5;\n\tborder: 2px solid #D64958;\n\tcolor:
```

```
#D64958;\n}\n.inputField-customizable {\n\twidth: 100%;\n\theight: 34px;\n\tcolor:
#555;\n\tbackground-color: #fff;\n\tborder: 1px solid #ccc;\n}\n.inputField-
customizable:focus {\n\tborder-color: #66afe9;\n\toutline: 0;\n}\n.idpButton-
customizable {\n\theight: 40px;\n\twidth: 100%;\n\ttext-align: center;\n\tmargin-
bottom: 15px;\n\tcolor: #fff;\n\tbackground-color: #5bc0de;\n\tborder-color:
#46b8da;\n}\n.idpButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color:
#31b0d5;\n}\n.socialButton-customizable {\n\theight: 40px;\n\ttext-align: left;
\n\twidth: 100%;\n\tmargin-bottom: 15px;\n}\n.redirect-customizable {\n\ttext-
align: center;\n}\n.passwordCheck-notValid-customizable {\n\tcolor: #DF3312;
\n}\n.passwordCheck-valid-customizable {\n\tcolor: #19BF00;\n}\n.background-
customizable {\n\tbackground-color: #faf;\n}\n",
  "CSSVersion": "20190128231240"
}
}
```

- Para API obtener más información, consulte [GetUiCustomization](#) la Referencia de AWS CLI comandos.

## list-user-import-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-user-import-jobs`.

### AWS CLI

Para enumerar los trabajos de importación de usuarios

En este ejemplo se enumeran los trabajos de importación de usuarios.

Para obtener más información sobre la importación de usuarios, consulte Importación de usuarios a grupos de usuarios desde un CSV archivo.

Comando:

```
aws cognito-idp list-user-import-jobs --user-pool-id us-west-2_aaaaaaaaaa --max-
results 20
```

Salida:

```
{
  "UserImportJobs": [
    {
      "JobName": "Test2",
      "JobId": "import-d00nwGA3mV",
```

```

    "UserPoolId": "us-west-2_aaaaaaaaa",
    "PreSignedUrl": "PRE_SIGNED_URL",
    "CreationDate": 1548272793.069,
    "Status": "Created",
    "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
    "ImportedUsers": 0,
    "SkippedUsers": 0,
    "FailedUsers": 0
  },
  {
    "JobName": "Test1",
    "JobId": "import-qQ0DCt2fRh",
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "PreSignedUrl": "PRE_SIGNED_URL",
    "CreationDate": 1548271795.471,
    "Status": "Created",
    "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
    "ImportedUsers": 0,
    "SkippedUsers": 0,
    "FailedUsers": 0
  },
  {
    "JobName": "import-Test1",
    "JobId": "import-TZqNQvDRnW",
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "PreSignedUrl": "PRE_SIGNED_URL",
    "CreationDate": 1548271708.512,
    "StartDate": 1548277247.962,
    "CompletionDate": 1548277248.912,
    "Status": "Failed",
    "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
    "ImportedUsers": 0,
    "SkippedUsers": 0,
    "FailedUsers": 1,
    "CompletionMessage": "Too many users have failed or been skipped during
the import."
  }
]
}

```

- Para API obtener más información, consulte [ListUserImportJobs](#) la Referencia de AWS CLI comandos.

## list-user-pools

En el siguiente ejemplo de código, se muestra cómo usar `list-user-pools`.

AWS CLI

Para mostrar los grupos de usuarios

En este ejemplo se muestran hasta 20 grupos de usuarios.

Comando:

```
aws cognito-idp list-user-pools --max-results 20
```

Salida:

```
{
  "UserPools": [
    {
      "CreationDate": 1547763720.822,
      "LastModifiedDate": 1547763720.822,
      "LambdaConfig": {},
      "Id": "us-west-2_aaaaaaaaa",
      "Name": "MyUserPool"
    }
  ]
}
```

- Para API obtener más información, consulte [ListUserPools](#) la Referencia de AWS CLI comandos.

## list-users-in-group

En el siguiente ejemplo de código, se muestra cómo usar `list-users-in-group`.

AWS CLI

Para enumerar los usuarios de un grupo

En este ejemplo se enumeran los usuarios del grupo MyGroup.

Comando:

```
aws cognito-idp list-users-in-group --user-pool-id us-west-2_aaaaaaaaa --group-name MyGroup
```

Salida:

```
{
  "Users": [
    {
      "Username": "acf10624-80bb-401a-ac61-607bee2110ec",
      "Attributes": [
        {
          "Name": "sub",
          "Value": "acf10624-80bb-401a-ac61-607bee2110ec"
        },
        {
          "Name": "custom:CustomAttr1",
          "Value": "New Value!"
        },
        {
          "Name": "email",
          "Value": "jane@example.com"
        }
      ],
      "UserCreateDate": 1548102770.284,
      "UserLastModifiedDate": 1548103204.893,
      "Enabled": true,
      "UserStatus": "CONFIRMED"
    },
    {
      "Username": "22704aa3-fc10-479a-97eb-2af5806bd327",
      "Attributes": [
        {
          "Name": "sub",
          "Value": "22704aa3-fc10-479a-97eb-2af5806bd327"
        },
        {
          "Name": "email_verified",
          "Value": "true"
        }
      ],
    }
  ]
}
```

```
        {
            "Name": "email",
            "Value": "diego@example.com"
        }
    ],
    "UserCreateDate": 1548089817.683,
    "UserLastModifiedDate": 1548089817.683,
    "Enabled": true,
    "UserStatus": "FORCE_CHANGE_PASSWORD"
}
]
```

- Para API obtener más información, consulte [ListUsersInGroup](#) la Referencia de AWS CLI comandos.

## list-users

En el siguiente ejemplo de código, se muestra cómo usar `list-users`.

### AWS CLI

Para mostrar los usuarios

En este ejemplo se muestran hasta 20 usuarios.

Comando:

```
aws cognito-idp list-users --user-pool-id us-west-2_aaaaaaaaa --limit 20
```

Salida:

```
{
  "Users": [
    {
      "Username": "22704aa3-fc10-479a-97eb-2af5806bd327",
      "Enabled": true,
      "UserStatus": "FORCE_CHANGE_PASSWORD",
      "UserCreateDate": 1548089817.683,
      "UserLastModifiedDate": 1548089817.683,
      "Attributes": [
```

```

    {
      "Name": "sub",
      "Value": "22704aa3-fc10-479a-97eb-2af5806bd327"
    },
    {
      "Name": "email_verified",
      "Value": "true"
    },
    {
      "Name": "email",
      "Value": "mary@example.com"
    }
  ]
}

```

- Para API obtener más información, consulte [ListUsers](#) la Referencia de AWS CLI comandos.

## resend-confirmation-code

En el siguiente ejemplo de código, se muestra cómo usar `resend-confirmation-code`.

### AWS CLI

Para reenviar un código de confirmación

En el siguiente ejemplo `resend-confirmation-code`, se envía un código de confirmación al usuario `jane`.

```

aws cognito-idp resend-confirmation-code \
  --client-id 12a3b456c7de890f11g123hijk \
  --username jane

```

Salida:

```

{
  "CodeDeliveryDetails": {
    "Destination": "j***@e***.com",
    "DeliveryMedium": "EMAIL",
    "AttributeName": "email"
  }
}

```



```
}
```

Para obtener más información, consulte [Inscripción y confirmación de cuentas de usuarios](#) en la Guía para desarrolladores de Amazon Cognito.

- Para API obtener más información, consulte [ResendConfirmationCode](#) la Referencia de AWS CLI comandos.

## respond-to-auth-challenge

En el siguiente ejemplo de código, se muestra cómo usar `respond-to-auth-challenge`.

### AWS CLI

Para responder a un desafío de autorización

Este ejemplo responde a un desafío de autorización iniciado con `initiate-auth`. Es una respuesta al REQUIRED desafío `NEW_PASSWORD_`. Establece una contraseña para el usuario `jane@example.com`.

Comando:

```
aws cognito-idp respond-to-auth-challenge --client-id 3n4b5urk1ft4fL3mg5e62d9ado  
--challenge-name NEW_PASSWORD_REQUIRED --challenge-responses  
USERNAME=jane@example.com,NEW_PASSWORD="password" --session "SESSION_TOKEN"
```

Salida:

```
{  
  "ChallengeParameters": {},  
  "AuthenticationResult": {  
    "AccessToken": "ACCESS_TOKEN",  
    "ExpiresIn": 3600,  
    "TokenType": "Bearer",  
    "RefreshToken": "REFRESH_TOKEN",  
    "IdToken": "ID_TOKEN",  
    "NewDeviceMetadata": {  
      "DeviceKey": "us-west-2_fec070d2-fa88-424a-8ec8-b26d7198eb23",  
      "DeviceGroupKey": "-wt2ha1Zd"  
    }  
  }  
}
```

```
}
```

- Para API obtener más información, consulte [RespondToAuthChallenge](#) la Referencia de AWS CLI comandos.

## set-risk-configuration

En el siguiente ejemplo de código, se muestra cómo usar `set-risk-configuration`.

### AWS CLI

Para establecer la configuración de riesgo

En este ejemplo, se establece la configuración de riesgo de un grupo de usuarios. Establece la acción del evento de registro en ACTION NO\_.

Comando:

```
aws cognito-idp set-risk-configuration --user-pool-id us-west-2_aaaaaaaaa --  
compromised-credentials-risk-  
configuration EventFilter=SIGN_UP,Actions={EventAction=NO_ACTION}
```

Salida:

```
{  
  "RiskConfiguration": {  
    "UserPoolId": "us-west-2_aaaaaaaaa",  
    "CompromisedCredentialsRiskConfiguration": {  
      "EventFilter": [  
        "SIGN_UP"  
      ],  
      "Actions": {  
        "EventAction": "NO_ACTION"  
      }  
    }  
  }  
}
```

- Para API obtener más información, consulte la Referencia [SetRiskConfiguration](#) de AWS CLI comandos.

## set-ui-customization

En el siguiente ejemplo de código, se muestra cómo usar `set-ui-customization`.

### AWS CLI

Para configurar la personalización de la interfaz de usuario

En este ejemplo, se personaliza la CSS configuración de un grupo de usuarios.

Comando:

```
aws cognito-idp set-ui-customization --user-pool-id us-west-2_aaaaaaaaa --
css ".logo-customizable {\n\tmax-width: 60%;\n\tmax-height: 30%;\n}\n.banner-
customizable {\n\tpadding: 25px 0px 25px 10px;\n\tbackground-color: lightgray;
\n}\n.label-customizable {\n\tfont-weight: 300;\n}\n.textDescription-customizable
{\n\tpadding-top: 10px;\n\tpadding-bottom: 10px;\n\tdisplay: block;\n\tfont-
size: 16px;\n}\n.idpDescription-customizable {\n\tpadding-top: 10px;\n\tpadding-
bottom: 10px;\n\tdisplay: block;\n\tfont-size: 16px;\n}\n.legalText-customizable
{\n\tcolor: #747474;\n\tfont-size: 11px;\n}\n.submitButton-customizable
{\n\tfont-size: 14px;\n\tfont-weight: bold;\n\tmargin: 20px 0px 10px 0px;\n
\theight: 40px;\n\twidth: 100%;\n\tcolor: #fff;\n\tbackground-color: #337ab7;
\n}\n.submitButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color:
#286090;\n}\n.errorMessage-customizable {\n\tpadding: 5px;\n\tfont-size: 14px;
\n\twidth: 100%;\n\tbackground: #F5F5F5;\n\tborder: 2px solid #D64958;\n\tcolor:
#D64958;\n}\n.inputField-customizable {\n\twidth: 100%;\n\theight: 34px;\n\tcolor:
#555;\n\tbackground-color: #fff;\n\tborder: 1px solid #ccc;\n}\n.inputField-
customizable:focus {\n\tborder-color: #66afe9;\n\toutline: 0;\n}\n.idpButton-
customizable {\n\theight: 40px;\n\twidth: 100%;\n\ttext-align: center;\n\tmargin-
bottom: 15px;\n\tcolor: #fff;\n\tbackground-color: #5bc0de;\n\tborder-color:
#46b8da;\n}\n.idpButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color:
#31b0d5;\n}\n.socialButton-customizable {\n\theight: 40px;\n\ttext-align: left;
\n\twidth: 100%;\n\tmargin-bottom: 15px;\n}\n.redirect-customizable {\n\ttext-
align: center;\n}\n.passwordCheck-notValid-customizable {\n\tcolor: #DF3312;
\n}\n.passwordCheck-valid-customizable {\n\tcolor: #19BF00;\n}\n.background-
customizable {\n\tbackground-color: #faf;\n}\n"
```

Salida:

```
{
  "UICustomization": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "ClientId": "ALL",
```

```

    "CSS": ".logo-customizable {\n\tmax-width: 60%;\n\tmax-height: 30%;
\n}\n.banner-customizable {\n\tpadding: 25px 0px 25px 10px;\n\tbackground-color:
lightgray;\n}\n.label-customizable {\n\tfont-weight: 300;\n}\n.textDescription-
customizable {\n\tpadding-top: 10px;\n\tpadding-bottom: 10px;\n\tdisplay: block;
\n\tfont-size: 16px;\n}\n.idpDescription-customizable {\n\tpadding-top: 10px;\n
\tpadding-bottom: 10px;\n\tdisplay: block;\n\tfont-size: 16px;\n}\n.legalText-
customizable {\n\tcolor: #747474;\n\tfont-size: 11px;\n}\n.submitButton-customizable
{\n\tfont-size: 14px;\n\tfont-weight: bold;\n\tmargin: 20px 0px 10px 0px;\n
\theight: 40px;\n\twidth: 100%;\n\tcolor: #fff;\n\tbackground-color: #337ab7;
\n}\n.submitButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color:
#286090;\n}\n.errorMessage-customizable {\n\tpadding: 5px;\n\tfont-size: 14px;
\n\twidth: 100%;\n\tbackground: #F5F5F5;\n\tborder: 2px solid #D64958;\n\tcolor:
#D64958;\n}\n.inputField-customizable {\n\twidth: 100%;\n\theight: 34px;\n\tcolor:
#555;\n\tbackground-color: #fff;\n\tborder: 1px solid #ccc;\n}\n.inputField-
customizable:focus {\n\tborder-color: #66afe9;\n\toutline: 0;\n}\n.idpButton-
customizable {\n\theight: 40px;\n\twidth: 100%;\n\ttext-align: center;\n\tmargin-
bottom: 15px;\n\tcolor: #fff;\n\tbackground-color: #5bc0de;\n\tborder-color:
#46b8da;\n}\n.idpButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color:
#31b0d5;\n}\n.socialButton-customizable {\n\theight: 40px;\n\ttext-align: left;
\n\twidth: 100%;\n\tmargin-bottom: 15px;\n}\n.redirect-customizable {\n\ttext-
align: center;\n}\n.passwordCheck-notValid-customizable {\n\tcolor: #DF3312;
\n}\n.passwordCheck-valid-customizable {\n\tcolor: #19BF00;\n}\n.background-
customizable {\n\tbackground-color: #faf;\n}\n",
    "CSSVersion": "20190129172214"
  }
}

```

- Para API obtener más información, consulte [SetUiCustomization](#) la Referencia de AWS CLI comandos.

## set-user-mfa-preference

En el siguiente ejemplo de código, se muestra cómo usar `set-user-mfa-preference`.

### AWS CLI

Para establecer la MFA configuración de usuario

En el siguiente `set-user-mfa-preference` ejemplo, se modifican las opciones MFA de entrega. Cambia el medio de MFA entrega a SMS.

```
aws cognito-idp set-user-mfa-preference \
```

```
--access-token "eyJra12345EXAMPLE" \  
--software-token-mfa-settings Enabled=true,PreferredMfa=true \  
--sms-mfa-settings Enabled=false,PreferredMfa=false
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir MFA a un grupo de usuarios](#) en la Guía para desarrolladores de Amazon Cognito.

- Para API obtener más información, consulte [SetUserMfaPreference](#) la Referencia de AWS CLI comandos.

## set-user-settings

En el siguiente ejemplo de código, se muestra cómo usar `set-user-settings`.

AWS CLI

Para establecer la configuración de usuario

En este ejemplo, se establece la preferencia de MFA entrega en EMAIL.

Comando:

```
aws cognito-idp set-user-settings --access-token ACCESS_TOKEN --mfa-  
options DeliveryMedium=EMAIL
```

- Para API obtener más información, consulte [SetUserSettings](#) la Referencia de AWS CLI comandos.

## sign-up

En el siguiente ejemplo de código, se muestra cómo usar `sign-up`.

AWS CLI

Para inscribir a un usuario

En este ejemplo, se registra `jane@example.com`.

Comando:

```
aws cognito-idp sign-up --client-id 3n4b5urk1ft4fL3mg5e62d9ado --  
username jane@example.com --password PASSWORD --user-attributes  
Name="email",Value="jane@example.com" Name="name",Value="Jane"
```

Salida:

```
{  
  "UserConfirmed": false,  
  "UserSub": "e04d60a6-45dc-441c-a40b-e25a787d4862"  
}
```

- Para API obtener más información, consulte [SignUp](#) la Referencia de AWS CLI comandos.

## start-user-import-job

En el siguiente ejemplo de código, se muestra cómo usar `start-user-import-job`.

### AWS CLI

Para iniciar un trabajo de importación de usuarios

En este ejemplo se inicia un trabajo de entrada de usuario.

Para obtener más información sobre la importación de usuarios, consulte [Importación de usuarios a grupos de usuarios desde un CSV archivo](#).

Comando:

```
aws cognito-idp start-user-import-job --user-pool-id us-west-2_aaaaaaaaa --job-  
id import-TZqNQvDRnW
```

Salida:

```
{  
  "UserImportJob": {  
    "JobName": "import-Test10",  
    "JobId": "import-lmpxS0uIzH",  
    "UserPoolId": "us-west-2_aaaaaaaaa",  
    "PreSignedUrl": "PRE_SIGNED_URL",  
    "CreationDate": 1548278378.928,  
  }  
}
```

```
"StartDate": 1548278397.334,  
"Status": "Pending",  
"CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/  
CognitoCloudWatchLogsRole",  
"ImportedUsers": 0,  
"SkippedUsers": 0,  
"FailedUsers": 0  
}  
}
```

- Para API obtener más información, consulte [StartUserImportJob](#) la Referencia de AWS CLI comandos.

## stop-user-import-job

En el siguiente ejemplo de código, se muestra cómo usar `stop-user-import-job`.

### AWS CLI

Para detener un trabajo de importación de usuarios

En este ejemplo se detiene un trabajo de entrada de un usuario.

Para obtener más información sobre la importación de usuarios, consulte [Importación de usuarios a grupos de usuarios desde un CSV archivo](#).

Comando:

```
aws cognito-idp stop-user-import-job --user-pool-id us-west-2_aaaaaaaaa --job-  
id import-TZqNQvDRnW
```

Salida:

```
{  
  "UserImportJob": {  
    "JobName": "import-Test5",  
    "JobId": "import-Fx0kARISFL",  
    "UserPoolId": "us-west-2_aaaaaaaaa",  
    "PreSignedUrl": "PRE_SIGNED_URL",  
    "CreationDate": 1548278576.259,  
    "StartDate": 1548278623.366,  
  }  
}
```

```
"CompletionDate": 1548278626.741,  
"Status": "Stopped",  
"CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/  
CognitoCloudWatchLogsRole",  
"ImportedUsers": 0,  
"SkippedUsers": 0,  
"FailedUsers": 0,  
"CompletionMessage": "The Import Job was stopped by the developer."  
}  
}
```

- Para API obtener más información, consulte [StopUserImportJob](#) la Referencia de AWS CLI comandos.

## update-auth-event-feedback

En el siguiente ejemplo de código, se muestra cómo usar `update-auth-event-feedback`.

### AWS CLI

Para actualizar los comentarios sobre los eventos de autenticación

En este ejemplo, se actualizan los comentarios sobre los eventos de autorización. Marca el evento como «Válido».

Comando:

```
aws cognito-idp update-auth-event-feedback --user-pool-id us-west-2_aaaaaaaaa --  
username diego@example.com --event-id EVENT_ID --feedback-token FEEDBACK_TOKEN --  
feedback-value "Valid"
```

- Para API obtener más información, consulte [UpdateAuthEventFeedback](#) la Referencia de AWS CLI comandos.

## update-device-status

En el siguiente ejemplo de código, se muestra cómo usar `update-device-status`.

### AWS CLI

Para actualizar el estado del dispositivo



En este ejemplo, se actualiza el estado de un dispositivo a «not\_remember».

Comando:

```
aws cognito-idp update-device-status --access-token ACCESS_TOKEN --device-  
key DEVICE_KEY --device-remembered-status "not_remembered"
```

- Para API obtener más información, consulte la Referencia [UpdateDeviceStatus](#) de AWS CLI comandos.

## update-group

En el siguiente ejemplo de código, se muestra cómo usar update-group.

AWS CLI

Para actualizar un grupo

En este ejemplo se actualizan la descripción y la prioridad de MyGroup.

Comando:

```
aws cognito-idp update-group --user-pool-id us-west-2_aaaaaaaaa --group-name MyGroup  
--description "New description" --precedence 2
```

Salida:

```
{  
  "Group": {  
    "GroupName": "MyGroup",  
    "UserPoolId": "us-west-2_aaaaaaaaa",  
    "Description": "New description",  
    "RoleArn": "arn:aws:iam::111111111111:role/MyRole",  
    "Precedence": 2,  
    "LastModifiedDate": 1548800862.812,  
    "CreationDate": 1548097827.125  
  }  
}
```

- Para API obtener más información, consulte [UpdateGroup](#) la Referencia de AWS CLI comandos.

## update-resource-server

En el siguiente ejemplo de código, se muestra cómo usar `update-resource-server`.

### AWS CLI

Para actualizar un servidor de recursos

En este ejemplo, se actualiza el clima del servidor de recursos. Agrega un nuevo ámbito.

Comando:

```
aws cognito-idp update-resource-server --user-pool-id us-west-2_aaaaaaaaa
--identifier weather.example.com --name Weather --scopes
ScopeName=NewScope,ScopeDescription="New scope description"
```

Salida:

```
{
  "ResourceServer": {
    "UserPoolId": "us-west-2_aaaaaaaaa",
    "Identifier": "weather.example.com",
    "Name": "Happy",
    "Scopes": [
      {
        "ScopeName": "NewScope",
        "ScopeDescription": "New scope description"
      }
    ]
  }
}
```

- Para API obtener más información, consulte [UpdateResourceServer](#) la Referencia de AWS CLI comandos.

## update-user-attributes

En el siguiente ejemplo de código, se muestra cómo usar `update-user-attributes`.

### AWS CLI

Para actualizar los atributos de usuario

En este ejemplo se actualiza el atributo de usuario «apodo».

Comando:

```
aws cognito-idp update-user-attributes --access-token ACCESS_TOKEN --user-attributes  
Name="nickname",Value="Dan"
```

- Para API obtener más información, consulte [UpdateUserAttributes](#) la Referencia de AWS CLI comandos.

## update-user-pool-client

En el siguiente ejemplo de código, se muestra cómo usar `update-user-pool-client`.

AWS CLI

Para actualizar un cliente de grupo de usuarios

En este ejemplo se actualiza el nombre de un cliente de grupo de usuarios. También añade un atributo «apodo» que se puede escribir.

Comando:

```
aws cognito-idp update-user-pool-client --user-pool-id us-west-2_aaaaaaaaa --  
client-id 3n4b5urk1ft4fl3mg5e62d9ado --client-name "NewClientName" --write-  
attributes "nickname"
```

Salida:

```
{  
  "UserPoolClient": {  
    "UserPoolId": "us-west-2_aaaaaaaaa",  
    "ClientName": "NewClientName",  
    "ClientId": "3n4b5urk1ft4fl3mg5e62d9ado",  
    "LastModifiedDate": 1548802761.334,  
    "CreationDate": 1548178931.258,  
    "RefreshTokenValidity": 30,  
    "WriteAttributes": [  
      "nickname"  
    ],  
  },  
}
```

```
    "AllowedOAuthFlowsUserPoolClient": false
  }
}
```

- Para API obtener más información, consulte la Referencia [UpdateUserPoolClient](#) de AWS CLI comandos.

## update-user-pool

En el siguiente ejemplo de código, se muestra cómo usar `update-user-pool`.

### AWS CLI

Para actualizar un grupo de usuarios

En este ejemplo, se agregan etiquetas a un grupo de usuarios.

Comando:

```
aws cognito-idp update-user-pool --user-pool-id us-west-2_aaaaaaaaa --user-pool-  
tags Team=Blue,Area=West
```

- Para API obtener más información, consulte [UpdateUserPool](#) la Referencia de AWS CLI comandos.

## Amazon Comprehend ejemplos utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Amazon Comprehend.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### batch-detect-dominant-language

En el siguiente ejemplo de código, se muestra cómo usar `batch-detect-dominant-language`.

#### AWS CLI

Para detectar el idioma dominante de los textos de entrada múltiple

El siguiente `batch-detect-dominant-language` ejemplo analiza varios textos de entrada y devuelve el idioma dominante de cada uno. La puntuación de confianza de los modelos previamente entrenados también se muestra para cada predicción.

```
aws comprehend batch-detect-dominant-language \  
  --text-list "Physics is the natural science that involves the study of matter  
and its motion and behavior through space and time, along with related concepts  
such as energy and force."
```

Salida:

```
{  
  "ResultList": [  
    {  
      "Index": 0,  
      "Languages": [  
        {  
          "LanguageCode": "en",  
          "Score": 0.9986501932144165  
        }  
      ]  
    }  
  ],  
  "ErrorList": []  
}
```

Para obtener más información, consulte [Idioma dominante](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [BatchDetectDominantLanguage](#) la Referencia de AWS CLI comandos.

## batch-detect-entities

En el siguiente ejemplo de código, se muestra cómo usar `batch-detect-entities`.

### AWS CLI

Para detectar entidades a partir de varios textos de entrada

El siguiente `batch-detect-entities` ejemplo analiza varios textos de entrada y devuelve las entidades nombradas de cada uno. La puntuación de confianza del modelo previamente entrenado también se muestra para cada predicción.

```
aws comprehend batch-detect-entities \  
  --language-code en \  
  --text-list "Dear Jane, Your AnyCompany Financial Services LLC credit card  
account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July  
31st." "Please send customer feedback to Sunshine Spa, 123 Main St, Anywhere or to  
Alice at AnySpa@example.com."
```

Salida:

```
{  
  "ResultList": [  
    {  
      "Index": 0,  
      "Entities": [  
        {  
          "Score": 0.9985517859458923,  
          "Type": "PERSON",  
          "Text": "Jane",  
          "BeginOffset": 5,  
          "EndOffset": 9  
        },  
        {  
          "Score": 0.9767839312553406,  
          "Type": "ORGANIZATION",  
          "Text": "AnyCompany Financial Services, LLC",  
          "BeginOffset": 16,  
          "EndOffset": 50  
        },  
        {  
          "Score": 0.9856694936752319,  
          "Type": "OTHER",
```

```
        "Text": "1111-XXXX-1111-XXXX",
        "BeginOffset": 71,
        "EndOffset": 90
    },
    {
        "Score": 0.9652159810066223,
        "Type": "QUANTITY",
        "Text": ".53",
        "BeginOffset": 116,
        "EndOffset": 119
    },
    {
        "Score": 0.9986667037010193,
        "Type": "DATE",
        "Text": "July 31st",
        "BeginOffset": 135,
        "EndOffset": 144
    }
]
},
{
    "Index": 1,
    "Entities": [
        {
            "Score": 0.720084547996521,
            "Type": "ORGANIZATION",
            "Text": "Sunshine Spa",
            "BeginOffset": 33,
            "EndOffset": 45
        },
        {
            "Score": 0.9865870475769043,
            "Type": "LOCATION",
            "Text": "123 Main St",
            "BeginOffset": 47,
            "EndOffset": 58
        },
        {
            "Score": 0.5895616412162781,
            "Type": "LOCATION",
            "Text": "Anywhere",
            "BeginOffset": 60,
            "EndOffset": 68
        }
    ]
},
```

```

        {
            "Score": 0.6809214353561401,
            "Type": "PERSON",
            "Text": "Alice",
            "BeginOffset": 75,
            "EndOffset": 80
        },
        {
            "Score": 0.9979087114334106,
            "Type": "OTHER",
            "Text": "AnySpa@example.com",
            "BeginOffset": 84,
            "EndOffset": 99
        }
    ]
},
"ErrorList": []
}

```

Para obtener más información, consulte [Entidades](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [BatchDetectEntities](#) la Referencia de AWS CLI comandos.

## batch-detect-key-phrases

En el siguiente ejemplo de código, se muestra cómo usar batch-detect-key-phrases.

AWS CLI

Para detectar frases clave de varias entradas de texto

El siguiente batch-detect-key-phrases ejemplo analiza varios textos introducidos y devuelve las frases nominales clave de cada uno de ellos. También se muestra la puntuación de confianza del modelo previamente entrenado para cada predicción.

```

aws comprehend batch-detect-key-phrases \
  --language-code en \
  --text-list "Hello Zhang Wei, I am John, writing to you about the trip for
next Saturday." "Dear Jane, Your AnyCompany Financial Services LLC credit card

```



***account 1111-XXXX-1111-XXXX has a minimum payment of \$24.53 that is due by July 31st." "Please send customer feedback to Sunshine Spa, 123 Main St, Anywhere or to Alice at AnySpa@example.com."***

Salida:

```
{
  "ResultList": [
    {
      "Index": 0,
      "KeyPhrases": [
        {
          "Score": 0.99700927734375,
          "Text": "Zhang Wei",
          "BeginOffset": 6,
          "EndOffset": 15
        },
        {
          "Score": 0.9929308891296387,
          "Text": "John",
          "BeginOffset": 22,
          "EndOffset": 26
        },
        {
          "Score": 0.9997230172157288,
          "Text": "the trip",
          "BeginOffset": 49,
          "EndOffset": 57
        },
        {
          "Score": 0.9999470114707947,
          "Text": "next Saturday",
          "BeginOffset": 62,
          "EndOffset": 75
        }
      ]
    },
    {
      "Index": 1,
      "KeyPhrases": [
        {
          "Score": 0.8358274102210999,
          "Text": "Dear Jane",

```

```
        "BeginOffset": 0,
        "EndOffset": 9
    },
    {
        "Score": 0.989359974861145,
        "Text": "Your AnyCompany Financial Services",
        "BeginOffset": 11,
        "EndOffset": 45
    },
    {
        "Score": 0.8812323808670044,
        "Text": "LLC credit card account 1111-XXXX-1111-XXXX",
        "BeginOffset": 47,
        "EndOffset": 90
    },
    {
        "Score": 0.9999381899833679,
        "Text": "a minimum payment",
        "BeginOffset": 95,
        "EndOffset": 112
    },
    {
        "Score": 0.9997439980506897,
        "Text": ".53",
        "BeginOffset": 116,
        "EndOffset": 119
    },
    {
        "Score": 0.996875524520874,
        "Text": "July 31st",
        "BeginOffset": 135,
        "EndOffset": 144
    }
]
},
{
    "Index": 2,
    "KeyPhrases": [
        {
            "Score": 0.9990295767784119,
            "Text": "customer feedback",
            "BeginOffset": 12,
            "EndOffset": 29
        }
    ]
},
```

```
    {
      "Score": 0.9994127750396729,
      "Text": "Sunshine Spa",
      "BeginOffset": 33,
      "EndOffset": 45
    },
    {
      "Score": 0.9892991185188293,
      "Text": "123 Main St",
      "BeginOffset": 47,
      "EndOffset": 58
    },
    {
      "Score": 0.9969810843467712,
      "Text": "Alice",
      "BeginOffset": 75,
      "EndOffset": 80
    },
    {
      "Score": 0.9703696370124817,
      "Text": "AnySpa@example.com",
      "BeginOffset": 84,
      "EndOffset": 99
    }
  ]
},
"ErrorList": []
}
```

Para obtener más información, consulte [Frases clave](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [BatchDetectKeyPhrases](#) la Referencia de AWS CLI comandos.

## batch-detect-sentiment

En el siguiente ejemplo de código, se muestra cómo usar batch-detect-sentiment.

### AWS CLI

Para detectar la opinión predominante en los textos de entrada múltiple

El siguiente batch-detect-sentiment ejemplo analiza varios textos de entrada y devuelve el sentimiento predominante (POSITIVENEUTRAL,MIXED, oNEGATIVE, de cada uno).

```
aws comprehend batch-detect-sentiment \  
  --text-list "That movie was very boring, I can't believe it was over four hours long." "It is a beautiful day for hiking today." "My meal was okay, I'm excited to try other restaurants." \  
  --language-code en
```

Salida:

```
{  
  "ResultList": [  
    {  
      "Index": 0,  
      "Sentiment": "NEGATIVE",  
      "SentimentScore": {  
        "Positive": 0.00011316669406369328,  
        "Negative": 0.9995445609092712,  
        "Neutral": 0.00014722718333359808,  
        "Mixed": 0.00019498742767609656  
      }  
    },  
    {  
      "Index": 1,  
      "Sentiment": "POSITIVE",  
      "SentimentScore": {  
        "Positive": 0.9981263279914856,  
        "Negative": 0.00015240783977787942,  
        "Neutral": 0.0013876151060685515,  
        "Mixed": 0.00033366199932061136  
      }  
    },  
    {  
      "Index": 2,  
      "Sentiment": "MIXED",  
      "SentimentScore": {  
        "Positive": 0.15930435061454773,  
        "Negative": 0.11471917480230331,  
        "Neutral": 0.26897063851356506,  
        "Mixed": 0.45700588822364807  
      }  
    }  
  ]  
}
```

```

    ],
    "ErrorList": []
  }

```

Para obtener más información, consulte [Sentiment](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [BatchDetectSentiment](#) la Referencia de AWS CLI comandos.

## batch-detect-syntax

En el siguiente ejemplo de código, se muestra cómo usar batch-detect-syntax.

### AWS CLI

Para inspeccionar la sintaxis y las partes de la oración de las palabras en textos de entrada múltiple

El siguiente batch-detect-syntax ejemplo analiza la sintaxis de varios textos de entrada y devuelve las distintas partes de la oración. La puntuación de confianza del modelo previamente entrenado también se muestra para cada predicción.

```

aws comprehend batch-detect-syntax \
  --text-list "It is a beautiful day." "Can you please pass the salt?" "Please pay the bill before the 31st." \
  --language-code en

```

Salida:

```

{
  "ResultList": [
    {
      "Index": 0,
      "SyntaxTokens": [
        {
          "TokenId": 1,
          "Text": "It",
          "BeginOffset": 0,
          "EndOffset": 2,
          "PartOfSpeech": {
            "Tag": "PRON",

```

```
        "Score": 0.9999740719795227
      }
    },
    {
      "TokenId": 2,
      "Text": "is",
      "BeginOffset": 3,
      "EndOffset": 5,
      "PartOfSpeech": {
        "Tag": "VERB",
        "Score": 0.999937117099762
      }
    },
    {
      "TokenId": 3,
      "Text": "a",
      "BeginOffset": 6,
      "EndOffset": 7,
      "PartOfSpeech": {
        "Tag": "DET",
        "Score": 0.9999926686286926
      }
    },
    {
      "TokenId": 4,
      "Text": "beautiful",
      "BeginOffset": 8,
      "EndOffset": 17,
      "PartOfSpeech": {
        "Tag": "ADJ",
        "Score": 0.9987891912460327
      }
    },
    {
      "TokenId": 5,
      "Text": "day",
      "BeginOffset": 18,
      "EndOffset": 21,
      "PartOfSpeech": {
        "Tag": "NOUN",
        "Score": 0.9999778866767883
      }
    }
  ],
  {
```

```
        "TokenId": 6,
        "Text": ".",
        "BeginOffset": 21,
        "EndOffset": 22,
        "PartOfSpeech": {
            "Tag": "PUNCT",
            "Score": 0.9999974966049194
        }
    }
]
},
{
    "Index": 1,
    "SyntaxTokens": [
        {
            "TokenId": 1,
            "Text": "Can",
            "BeginOffset": 0,
            "EndOffset": 3,
            "PartOfSpeech": {
                "Tag": "AUX",
                "Score": 0.9999770522117615
            }
        },
        {
            "TokenId": 2,
            "Text": "you",
            "BeginOffset": 4,
            "EndOffset": 7,
            "PartOfSpeech": {
                "Tag": "PRON",
                "Score": 0.9999986886978149
            }
        },
        {
            "TokenId": 3,
            "Text": "please",
            "BeginOffset": 8,
            "EndOffset": 14,
            "PartOfSpeech": {
                "Tag": "INTJ",
                "Score": 0.9681622385978699
            }
        }
    ],
}
```

```
    {
      "TokenId": 4,
      "Text": "pass",
      "BeginOffset": 15,
      "EndOffset": 19,
      "PartOfSpeech": {
        "Tag": "VERB",
        "Score": 0.9999874830245972
      }
    },
    {
      "TokenId": 5,
      "Text": "the",
      "BeginOffset": 20,
      "EndOffset": 23,
      "PartOfSpeech": {
        "Tag": "DET",
        "Score": 0.9999827146530151
      }
    },
    {
      "TokenId": 6,
      "Text": "salt",
      "BeginOffset": 24,
      "EndOffset": 28,
      "PartOfSpeech": {
        "Tag": "NOUN",
        "Score": 0.9995040893554688
      }
    },
    {
      "TokenId": 7,
      "Text": "?",
      "BeginOffset": 28,
      "EndOffset": 29,
      "PartOfSpeech": {
        "Tag": "PUNCT",
        "Score": 0.999998152256012
      }
    }
  ]
},
{
  "Index": 2,
```



```
"SyntaxTokens": [  
  {  
    "TokenId": 1,  
    "Text": "Please",  
    "BeginOffset": 0,  
    "EndOffset": 6,  
    "PartOfSpeech": {  
      "Tag": "INTJ",  
      "Score": 0.9997857809066772  
    }  
  },  
  {  
    "TokenId": 2,  
    "Text": "pay",  
    "BeginOffset": 7,  
    "EndOffset": 10,  
    "PartOfSpeech": {  
      "Tag": "VERB",  
      "Score": 0.9999252557754517  
    }  
  },  
  {  
    "TokenId": 3,  
    "Text": "the",  
    "BeginOffset": 11,  
    "EndOffset": 14,  
    "PartOfSpeech": {  
      "Tag": "DET",  
      "Score": 0.9999842643737793  
    }  
  },  
  {  
    "TokenId": 4,  
    "Text": "bill",  
    "BeginOffset": 15,  
    "EndOffset": 19,  
    "PartOfSpeech": {  
      "Tag": "NOUN",  
      "Score": 0.9999588131904602  
    }  
  },  
  {  
    "TokenId": 5,  
    "Text": "before",
```

```
        "BeginOffset": 20,
        "EndOffset": 26,
        "PartOfSpeech": {
            "Tag": "ADP",
            "Score": 0.9958304762840271
        }
    },
    {
        "TokenId": 6,
        "Text": "the",
        "BeginOffset": 27,
        "EndOffset": 30,
        "PartOfSpeech": {
            "Tag": "DET",
            "Score": 0.9999947547912598
        }
    },
    {
        "TokenId": 7,
        "Text": "31st",
        "BeginOffset": 31,
        "EndOffset": 35,
        "PartOfSpeech": {
            "Tag": "NOUN",
            "Score": 0.9924124479293823
        }
    },
    {
        "TokenId": 8,
        "Text": ".",
        "BeginOffset": 35,
        "EndOffset": 36,
        "PartOfSpeech": {
            "Tag": "PUNCT",
            "Score": 0.9999955892562866
        }
    }
]
}
],
"ErrorList": []
}
```

Para obtener más información, consulte los [Análisis de sintaxis](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [BatchDetectSyntax](#) la Referencia de AWS CLI comandos.

## batch-detect-targeted-sentiment

En el siguiente ejemplo de código, se muestra cómo usar `batch-detect-targeted-sentiment`.

### AWS CLI

Para detectar el sentimiento y cada entidad nombrada en el caso de varios textos de entrada

El siguiente `batch-detect-targeted-sentiment` ejemplo analiza varios textos de entrada y devuelve las entidades nombradas junto con el sentimiento predominante asociado a cada entidad. La puntuación de confianza del modelo previamente entrenado también se muestra para cada predicción.

```
aws comprehend batch-detect-targeted-sentiment \  
  --language-code en \  
  --text-list "That movie was really boring, the original was way more  
entertaining" "The trail is extra beautiful today." "My meal was just okay."
```

Salida:

```
{  
  "ResultList": [  
    {  
      "Index": 0,  
      "Entities": [  
        {  
          "DescriptiveMentionIndex": [  
            0  
          ],  
          "Mentions": [  
            {  
              "Score": 0.9999009966850281,  
              "GroupScore": 1.0,  
              "Text": "movie",  
              "Type": "MOVIE",  
              "MentionSentiment": {
```

```

        "Sentiment": "NEGATIVE",
        "SentimentScore": {
            "Positive": 0.13887299597263336,
            "Negative": 0.8057460188865662,
            "Neutral": 0.05525200068950653,
            "Mixed": 0.00012799999967683107
        }
    },
    "BeginOffset": 5,
    "EndOffset": 10
}
]
},
{
    "DescriptiveMentionIndex": [
        0
    ],
    "Mentions": [
        {
            "Score": 0.9921110272407532,
            "GroupScore": 1.0,
            "Text": "original",
            "Type": "MOVIE",
            "MentionSentiment": {
                "Sentiment": "POSITIVE",
                "SentimentScore": {
                    "Positive": 0.9999989867210388,
                    "Negative": 9.999999974752427e-07,
                    "Neutral": 0.0,
                    "Mixed": 0.0
                }
            }
        },
        {
            "BeginOffset": 34,
            "EndOffset": 42
        }
    ]
}
]
},
{
    "Index": 1,
    "Entities": [
        {
            "DescriptiveMentionIndex": [

```

```
    0
  ],
  "Mentions": [
    {
      "Score": 0.7545599937438965,
      "GroupScore": 1.0,
      "Text": "trail",
      "Type": "OTHER",
      "MentionSentiment": {
        "Sentiment": "POSITIVE",
        "SentimentScore": {
          "Positive": 1.0,
          "Negative": 0.0,
          "Neutral": 0.0,
          "Mixed": 0.0
        }
      }
    },
    {
      "BeginOffset": 4,
      "EndOffset": 9
    }
  ]
},
{
  "DescriptiveMentionIndex": [
    0
  ],
  "Mentions": [
    {
      "Score": 0.9999960064888,
      "GroupScore": 1.0,
      "Text": "today",
      "Type": "DATE",
      "MentionSentiment": {
        "Sentiment": "NEUTRAL",
        "SentimentScore": {
          "Positive": 9.000000318337698e-06,
          "Negative": 1.999999949504854e-06,
          "Neutral": 0.9999859929084778,
          "Mixed": 3.99999989900971e-06
        }
      }
    },
    {
      "BeginOffset": 29,
      "EndOffset": 34
    }
  ]
}
```

```
    ]
  }
]
},
{
  "Index": 2,
  "Entities": [
    {
      "DescriptiveMentionIndex": [
        0
      ],
      "Mentions": [
        {
          "Score": 0.9999880194664001,
          "GroupScore": 1.0,
          "Text": "My",
          "Type": "PERSON",
          "MentionSentiment": {
            "Sentiment": "NEUTRAL",
            "SentimentScore": {
              "Positive": 0.0,
              "Negative": 0.0,
              "Neutral": 1.0,
              "Mixed": 0.0
            }
          }
        },
        {
          "BeginOffset": 0,
          "EndOffset": 2
        }
      ]
    }
  ],
  {
    "DescriptiveMentionIndex": [
      0
    ],
    "Mentions": [
      {
        "Score": 0.9995260238647461,
        "GroupScore": 1.0,
        "Text": "meal",
        "Type": "OTHER",
        "MentionSentiment": {
          "Sentiment": "NEUTRAL",
          "SentimentScore": {
```

```

        "Positive": 0.04695599898695946,
        "Negative": 0.003226999891921878,
        "Neutral": 0.6091709733009338,
        "Mixed": 0.34064599871635437
    }
},
"BeginOffset": 3,
"EndOffset": 7
}
]
}
],
"ErrorList": []
}

```

Para obtener más información, consulte [Targeted Sentiment](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [BatchDetectTargetedSentiment](#) la Referencia de AWS CLI comandos.

## classify-document

En el siguiente ejemplo de código, se muestra cómo usar `classify-document`.

### AWS CLI

Para clasificar el documento con un punto final específico del modelo

El siguiente `classify-document` ejemplo clasifica un documento con un punto final de un modelo personalizado. El modelo de este ejemplo se entrenó con un conjunto de datos que contenía mensajes SMS etiquetados como spam o no spam o «ham».

```

aws comprehend classify-document \
  --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-
  endpoint/example-classifier-endpoint \
  --text "CONGRATULATIONS! TXT 1235550100 to win $5000"

```

Salida:

```
{
  "Classes": [
    {
      "Name": "spam",
      "Score": 0.9998599290847778
    },
    {
      "Name": "ham",
      "Score": 0.00014001205272506922
    }
  ]
}
```

Para obtener más información, consulte [Clasificación personalizada](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [ClassifyDocument](#) la Referencia de AWS CLI comandos.

## **contains-pii-entities**

En el siguiente ejemplo de código, se muestra cómo usar `contains-pii-entities`.

### AWS CLI

Para analizar el texto introducido para detectar la presencia de PII información

El siguiente `contains-pii-entities` ejemplo analiza el texto introducido para detectar la presencia de información de identificación personal (PII) y devuelve las etiquetas de los tipos de PII entidades identificadas, como el nombre, la dirección, el número de cuenta bancaria o el número de teléfono.

```
aws comprehend contains-pii-entities \
  --language-code en \
  --text "Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC
  credit card
  account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by
  July 31st. Based on your autopay settings,
  we will withdraw your payment on the due date from your bank account number
  XXXXXX1111 with the routing number XXXXXX0000."
```



*Customer feedback for Sunshine Spa, 100 Main St, Anywhere. Send comments to Alice at AnySpa@example.com."*

Salida:

```
{
  "Labels": [
    {
      "Name": "NAME",
      "Score": 1.0
    },
    {
      "Name": "EMAIL",
      "Score": 1.0
    },
    {
      "Name": "BANK_ACCOUNT_NUMBER",
      "Score": 0.9995794296264648
    },
    {
      "Name": "BANK_ROUTING",
      "Score": 0.9173126816749573
    },
    {
      "Name": "CREDIT_DEBIT_NUMBER",
      "Score": 1.0
    }
  ]
}
```

Para obtener más información, consulte [Información de identificación personal \(PII\)](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [ContainsPiiEntities](#) la Referencia de AWS CLI comandos.

## create-dataset

En el siguiente ejemplo de código, se muestra cómo usar create-dataset.

### AWS CLI

Para crear un conjunto de datos de volantes

En el siguiente `create-dataset` ejemplo, se crea un conjunto de datos para un volante. Este conjunto de datos se utilizará como datos de entrenamiento adicionales según lo especifique la `--dataset-type` etiqueta.

```
aws comprehend create-dataset \  
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-  
entity \  
  --dataset-name example-dataset \  
  --dataset-type "TRAIN" \  
  --input-data-config file://inputConfig.json
```

Contenidos de `file://inputConfig.json`:

```
{  
  "DataFormat": "COMPREHEND_CSV",  
  "DocumentClassifierInputDataConfig": {  
    "S3Uri": "s3://DOC-EXAMPLE-BUCKET/training-data.csv"  
  }  
}
```

Salida:

```
{  
  "DatasetArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-  
entity/dataset/example-dataset"  
}
```

Para obtener más información, consulte [Descripción general de Flywheel](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [CreateDataset](#) de AWS CLI comandos.

## **create-document-classifier**

En el siguiente ejemplo de código, se muestra cómo usar `create-document-classifier`.

### AWS CLI

Para crear un clasificador de documentos para clasificar los documentos

El siguiente ejemplo de `create-document-classifier` inicia el proceso de entrenamiento de un modelo de clasificador de documentos. El archivo de datos de entrenamiento, `training.csv`, se encuentra en la etiqueta `--input-data-config`. `training.csv` es un documento de dos columnas donde las etiquetas o clasificaciones se proporcionan en la primera columna y los documentos en la segunda columna.

```
aws comprehend create-document-classifier \  
  --document-classifier-name example-classifier \  
  --data-access-arn arn:aws:comprehend:us-west-2:111122223333:pii-entities-  
detection-job/123456abcdeb0e11022f22a11EXAMPLE \  
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/" \  
  --language-code en
```

Salida:

```
{  
  "DocumentClassifierArn": "arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-classifier"  
}
```

Para obtener más información, consulte [Clasificación personalizada](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [CreateDocumentClassifier](#) la Referencia de AWS CLI comandos.

## create-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `create-endpoint`.

### AWS CLI

Para crear un punto final para un modelo personalizado

El siguiente `create-endpoint` ejemplo crea un punto final para la inferencia sincrónica de un modelo personalizado previamente entrenado.

```
aws comprehend create-endpoint \  
  --endpoint-name example-classifier-endpoint-1 \  
  --model-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier \  
  --language-code en
```

```
--desired-inference-units 1
```

Salida:

```
{
  "EndpointArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier-
endpoint/example-classifier-endpoint-1"
}
```

Para obtener más información, consulte [Administración de puntos de conexión de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [CreateEndpoint](#) de AWS CLI comandos.

## create-entity-recognizer

En el siguiente ejemplo de código, se muestra cómo usar `create-entity-recognizer`.

AWS CLI

Para crear un reconocedor de entidades personalizado

El siguiente `create-entity-recognizer` ejemplo inicia el proceso de entrenamiento de un modelo de reconocedor de entidades personalizado. En este ejemplo, `raw_text.csv` se utiliza un CSV archivo que contiene documentos de formación y una lista de CSV entidades `entity_list.csv` para entrenar el modelo. `entity-list.csv` contiene las siguientes columnas: texto y texto.

```
aws comprehend create-entity-recognizer \
  --recognizer-name example-entity-recognizer
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
  --input-data-config "EntityType=[{Type=DEVICE}], Documents={S3Uri=s3://DOC-
EXAMPLE-BUCKET/trainingdata/raw_text.csv}, EntityList={S3Uri=s3://DOC-EXAMPLE-BUCKET/
trainingdata/entity_list.csv}"
  --language-code en
```

Salida:

```
{
```

```
"EntityRecognizerArn": "arn:aws:comprehend:us-west-2:111122223333:example-  
entity-recognizer/entityrecognizer1"  
}
```

Para obtener más información, consulte [Reconocimiento de entidades personalizado](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [CreateEntityRecognizer](#) la Referencia de AWS CLI comandos.

## create-flywheel

En el siguiente ejemplo de código, se muestra cómo usar `create-flywheel`.

### AWS CLI

Para crear un volante

El siguiente `create-flywheel` ejemplo crea un volante para organizar el entrenamiento continuo de un modelo de clasificación de documentos o de reconocimiento de entidades. El volante de este ejemplo se creó para gestionar un modelo entrenado existente especificado por la etiqueta. `--active-model-arn` Cuando se crea el volante, se crea un lago de datos en la `--input-data-lake` etiqueta.

```
aws comprehend create-flywheel \  
  --flywheel-name example-flywheel \  
  --active-model-arn arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-model/version/1 \  
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role \  
  --data-lake-s3-uri "s3://DOC-EXAMPLE-BUCKET"
```

Salida:

```
{  
  "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-  
flywheel"  
}
```

Para obtener más información, consulte [Descripción general de Flywheel](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [CreateFlywheel](#) de AWS CLI comandos.

## **delete-document-classifier**

En el siguiente ejemplo de código, se muestra cómo usar `delete-document-classifier`.

AWS CLI

Para eliminar un clasificador de documentos personalizado

En el siguiente ejemplo de `delete-document-classifier`, se elimina un modelo de clasificador de documentos personalizado.

```
aws comprehend delete-document-classifier \  
  --document-classifier-arn arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-classifier-1
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de puntos de conexión de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DeleteDocumentClassifier](#) la Referencia de AWS CLI comandos.

## **delete-endpoint**

En el siguiente ejemplo de código, se muestra cómo usar `delete-endpoint`.

AWS CLI

Para eliminar un punto final de un modelo personalizado

En el siguiente `delete-endpoint` ejemplo, se elimina un punto final específico del modelo. Para poder eliminar el modelo, se deben borrar todos los puntos de enlace.

```
aws comprehend delete-endpoint \  
  --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-  
endpoint/example-classifier-endpoint-1
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de puntos de conexión de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia de [DeleteEndpoint](#) comandos AWS CLI .

## **delete-entity-recognizer**

En el siguiente ejemplo de código, se muestra cómo usar `delete-entity-recognizer`.

AWS CLI

Para eliminar un modelo de reconocedor de entidades personalizado

En el siguiente `delete-entity-recognizer` ejemplo, se elimina un modelo de reconocedor de entidades personalizado.

```
aws comprehend delete-entity-recognizer \  
  --entity-recognizer-arn arn:aws:comprehend:us-west-2:111122223333:entity-recognizer/example-entity-recognizer-1
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de puntos de conexión de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [DeleteEntityRecognizer](#) de AWS CLI comandos.

## **delete-flywheel**

En el siguiente ejemplo de código, se muestra cómo usar `delete-flywheel`.

AWS CLI

Para eliminar un volante

En el `delete-flywheel` ejemplo siguiente se elimina un volante. El lago de datos o el modelo asociado al volante no se eliminan.

```
aws comprehend delete-flywheel \  
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-  
flywheel-1
```

Este comando no genera ninguna salida.

Para obtener más información, consulte la [descripción general de Flywheel](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [DeleteFlywheel](#) de AWS CLI comandos.

## delete-resource-policy

En el siguiente ejemplo de código, se muestra cómo usar delete-resource-policy.

AWS CLI

Para eliminar una política basada en recursos

El siguiente delete-resource-policy ejemplo elimina una política basada en recursos de un recurso de Amazon Comprehend.

```
aws comprehend delete-resource-policy \  
  --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier-1/version/1
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Copiar modelos personalizados entre AWS cuentas](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DeleteResourcePolicy](#) la Referencia de AWS CLI comandos.

## describe-dataset

En el siguiente ejemplo de código, se muestra cómo usar describe-dataset.

AWS CLI

Para describir un conjunto de datos sobre volantes



En el siguiente describe-dataset ejemplo, se obtienen las propiedades de un conjunto de datos de volante.

```
aws comprehend describe-dataset \  
  --dataset-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-  
entity/dataset/example-dataset
```

Salida:

```
{  
  "DatasetProperties": {  
    "DatasetArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-  
entity/dataset/example-dataset",  
    "DatasetName": "example-dataset",  
    "DatasetType": "TRAIN",  
    "DatasetS3Uri": "s3://DOC-EXAMPLE-BUCKET/flywheel-entity/  
schemaVersion=1/12345678A123456Z/datasets/example-dataset/20230616T203710Z/",  
    "Status": "CREATING",  
    "CreationTime": "2023-06-16T20:37:10.400000+00:00"  
  }  
}
```

Para obtener más información, consulte [Descripción general de Flywheel](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [DescribeDataset](#) de AWS CLI comandos.

## describe-document-classification-job

En el siguiente ejemplo de código, se muestra cómo usar describe-document-classification-job.

AWS CLI

Para describir un trabajo de clasificación de documentos

En el siguiente ejemplo de describe-document-classification-job, se obtienen las propiedades de un trabajo de clasificación de documentos asíncrono.

```
aws comprehend describe-document-classification-job \  
  --dataset-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-  
entity/dataset/example-dataset
```

```
--job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{
  "DocumentClassificationJobProperties": {
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classification-job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobName": "exampleclassificationjob",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2023-06-14T17:09:51.788000+00:00",
    "EndTime": "2023-06-14T17:15:58.582000+00:00",
    "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/mymodel/version/1",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/jobdata/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/111122223333-
CLN-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
    },
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-servicerole"
  }
}
```

Para obtener más información, consulte [Clasificación personalizada](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DescribeDocumentClassificationJob](#) la Referencia de AWS CLI comandos.

## describe-document-classifier

En el siguiente ejemplo de código, se muestra cómo usar `describe-document-classifier`.

AWS CLI

Para describir un clasificador de documentos

En el siguiente ejemplo de `describe-document-classifier`, se obtienen las propiedades de un modelo de clasificador de documentos personalizado.

```
aws comprehend describe-document-classifier \  
  --document-classifier-arn arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-classifier-1
```

Salida:

```
{  
  "DocumentClassifierProperties": {  
    "DocumentClassifierArn": "arn:aws:comprehend:us-  
west-2:111122223333:document-classifier/example-classifier-1",  
    "LanguageCode": "en",  
    "Status": "TRAINED",  
    "SubmitTime": "2023-06-13T19:04:15.735000+00:00",  
    "EndTime": "2023-06-13T19:42:31.752000+00:00",  
    "TrainingStartTime": "2023-06-13T19:08:20.114000+00:00",  
    "TrainingEndTime": "2023-06-13T19:41:35.080000+00:00",  
    "InputDataConfig": {  
      "DataFormat": "COMPREHEND_CSV",  
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata"  
    },  
    "OutputDataConfig": {},  
    "ClassifierMetadata": {  
      "NumberOfLabels": 3,  
      "NumberOfTrainedDocuments": 5016,  
      "NumberOfTestDocuments": 557,  
      "EvaluationMetrics": {  
        "Accuracy": 0.9856,  
        "Precision": 0.9919,  
        "Recall": 0.9459,  
        "F1Score": 0.9673,  
        "MicroPrecision": 0.9856,  
        "MicroRecall": 0.9856,  
        "MicroF1Score": 0.9856,  
        "HammingLoss": 0.0144  
      }  
    },  
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role",  
    "Mode": "MULTI_CLASS"  
  }  
}
```

```
}
```

Para obtener más información, consulte [Creación y administración de modelos personalizados](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DescribeDocumentClassifier](#) la Referencia de AWS CLI comandos.

## describe-dominant-language-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-dominant-language-detection-job`.

### AWS CLI

Para describir un trabajo de detección del lenguaje dominante.

En el siguiente `describe-dominant-language-detection-job` ejemplo, se obtienen las propiedades de un trabajo de detección de lenguaje dominante asíncrono.

```
aws comprehend describe-dominant-language-detection-job \  
--job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{  
  "DominantLanguageDetectionJobProperties": {  
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:dominant-language-  
detection-job/123456abcdeb0e11022f22a11EXAMPLE",  
    "JobName": "languageanalysis1",  
    "JobStatus": "IN_PROGRESS",  
    "SubmitTime": "2023-06-09T18:10:38.037000+00:00",  
    "InputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET",  
      "InputFormat": "ONE_DOC_PER_LINE"  
    },  
    "OutputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/111122223333-  
LANGUAGE-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"  
    },  
  },  
}
```

```

    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
}

```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia de [DescribeDominantLanguageDetectionJob](#) comandos AWS CLI .

## describe-endpoint

En el siguiente ejemplo de código, se muestra cómo usar describe-endpoint.

### AWS CLI

Para describir un punto final específico

En el siguiente describe-endpoint ejemplo, se obtienen las propiedades de un punto final específico del modelo.

```

aws comprehend describe-endpoint \
  --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-
endpoint/example-classifier-endpoint

```

Salida:

```

{
  "EndpointProperties": {
    "EndpointArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier-endpoint/example-classifier-endpoint",
    "Status": "IN_SERVICE",
    "ModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/
exampleclassifier1",
    "DesiredModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier1",
    "DesiredInferenceUnits": 1,
    "CurrentInferenceUnits": 1,
    "CreationTime": "2023-06-13T20:32:54.526000+00:00",
    "LastModifiedTime": "2023-06-13T20:32:54.526000+00:00"
  }
}

```

```
}  
}
```

Para obtener más información, consulte [Administración de puntos de conexión de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [DescribeEndpoint](#) de AWS CLI comandos.

## describe-entities-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-entities-detection-job`.

### AWS CLI

Para describir un trabajo de detección de entidades

El siguiente `describe-entities-detection-job` ejemplo obtiene las propiedades de un trabajo de detección de entidades asincrónico.

```
aws comprehend describe-entities-detection-job \  
--job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{  
  "EntitiesDetectionJobProperties": {  
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-  
job/123456abcdeb0e11022f22a11EXAMPLE",  
    "JobName": "example-entity-detector",  
    "JobStatus": "COMPLETED",  
    "SubmitTime": "2023-06-08T21:30:15.323000+00:00",  
    "EndTime": "2023-06-08T21:40:23.509000+00:00",  
    "InputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",  
      "InputFormat": "ONE_DOC_PER_LINE"  
    },  
    "OutputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/thefolder/111122223333-  
NER-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"  
    }  
  }  
}
```

```
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::12345678012:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia de [DescribeEntitiesDetectionJob](#) comandos AWS CLI .

## describe-entity-recognizer

En el siguiente ejemplo de código, se muestra cómo usar `describe-entity-recognizer`.

### AWS CLI

Para describir un reconocedor de entidades

El siguiente `describe-entity-recognizer` ejemplo obtiene las propiedades de un modelo de reconocedor de entidades personalizado.

```
aws comprehend describe-entity-recognizer \
    entity-recognizer-arn arn:aws:comprehend:us-west-2:111122223333:entity-recognizer/business-recongizer-1/version/1
```

Salida:

```
{
  "EntityRecognizerProperties": {
    "EntityRecognizerArn": "arn:aws:comprehend:us-west-2:111122223333:entity-recognizer/business-recongizer-1/version/1",
    "LanguageCode": "en",
    "Status": "TRAINED",
    "SubmitTime": "2023-06-14T20:44:59.631000+00:00",
    "EndTime": "2023-06-14T20:59:19.532000+00:00",
    "TrainingStartTime": "2023-06-14T20:48:52.811000+00:00",
    "TrainingEndTime": "2023-06-14T20:58:11.473000+00:00",
    "InputDataConfig": {
      "DataFormat": "COMPREHEND_CSV",
```

```

    "EntityTypes": [
      {
        "Type": "BUSINESS"
      }
    ],
    "Documents": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata/dataset/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "EntityList": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata/entity.csv"
    }
  },
  "RecognizerMetadata": {
    "NumberOfTrainedDocuments": 1814,
    "NumberOfTestDocuments": 486,
    "EvaluationMetrics": {
      "Precision": 100.0,
      "Recall": 100.0,
      "F1Score": 100.0
    }
  },
  "EntityTypes": [
    {
      "Type": "BUSINESS",
      "EvaluationMetrics": {
        "Precision": 100.0,
        "Recall": 100.0,
        "F1Score": 100.0
      },
      "NumberOfTrainMentions": 1520
    }
  ]
},
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-example-role",
  "VersionName": "1"
}
}

```

Para obtener más información, consulte [Reconocimiento de entidades personalizado](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DescribeEntityRecognizer](#) la Referencia de AWS CLI comandos.



## describe-events-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-events-detection-job`.

### AWS CLI

Para describir un trabajo de detección de eventos.

El siguiente `describe-events-detection-job` ejemplo obtiene las propiedades de un trabajo de detección de eventos asíncrono.

```
aws comprehend describe-events-detection-job \  
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{  
  "EventsDetectionJobProperties": {  
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:events-detection-  
job/123456abcdeb0e11022f22a11EXAMPLE",  
    "JobName": "events_job_1",  
    "JobStatus": "IN_PROGRESS",  
    "SubmitTime": "2023-06-12T18:45:56.054000+00:00",  
    "InputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/EventsData",  
      "InputFormat": "ONE_DOC_PER_LINE"  
    },  
    "OutputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/111122223333-  
EVENTS-123456abcdeb0e11022f22a11EXAMPLE/output/"  
    },  
    "LanguageCode": "en",  
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role",  
    "TargetEventTypes": [  
      "BANKRUPTCY",  
      "EMPLOYMENT",  
      "CORPORATE_ACQUISITION",  
      "CORPORATE_MERGER",  
      "INVESTMENT_GENERAL"  
    ]  
  }  
}
```

```
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia de [DescribeEventsDetectionJob](#) comandos AWS CLI .

## describe-flywheel-iteration

En el siguiente ejemplo de código, se muestra cómo usar `describe-flywheel-iteration`.

### AWS CLI

Para describir una iteración del volante

En el siguiente `describe-flywheel-iteration` ejemplo se obtienen las propiedades de una iteración de volante.

```
aws comprehend describe-flywheel-iteration \
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-  
flywheel \
  --flywheel-iteration-id 20232222AEXAMPLE
```

Salida:

```
{
  "FlywheelIterationProperties": {
    "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-  
entity",
    "FlywheelIterationId": "20232222AEXAMPLE",
    "CreationTime": "2023-06-16T21:10:26.385000+00:00",
    "EndTime": "2023-06-16T23:33:16.827000+00:00",
    "Status": "COMPLETED",
    "Message": "FULL_ITERATION: Flywheel iteration performed all functions  
successfully.",
    "EvaluatedModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-classifier/version/1",
    "EvaluatedModelMetrics": {
      "AverageF1Score": 0.7742663922375772,
      "AveragePrecision": 0.8287636394041166,
      "AverageRecall": 0.7427084833645399,
      "AverageAccuracy": 0.8795394154118689
    }
  }
}
```

```

    },
    "TrainedModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/example-classifier/version/Comprehend-Generated-v1-bb52d585",
    "TrainedModelMetrics": {
      "AverageF1Score": 0.9767700253081214,
      "AveragePrecision": 0.9767700253081214,
      "AverageRecall": 0.9767700253081214,
      "AverageAccuracy": 0.9858281665190434
    },
    "EvaluationManifestS3Prefix": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/flywheel-
entity/schemaVersion=1/20230616T200543Z/evaluation/20230616T211026Z/"
  }
}

```

Para obtener más información, consulte la [descripción general de Flywheel](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [DescribeFlywheelIteration](#) de AWS CLI comandos.

## describe-flywheel

En el siguiente ejemplo de código, se muestra cómo usar `describe-flywheel`.

### AWS CLI

Para describir un volante

En el siguiente `describe-flywheel` ejemplo se obtienen las propiedades de un volante. En este ejemplo, el modelo asociado al volante de inercia es un modelo de clasificador personalizado que está entrenado para clasificar los documentos como spam o no spam, o como «jamón».

```

aws comprehend describe-flywheel \
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-
flywheel

```

Salida:

```

{
  "FlywheelProperties": {
    "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-
flywheel",

```

```

    "ActiveModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/example-model/version/1",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
    "TaskConfig": {
      "LanguageCode": "en",
      "DocumentClassificationConfig": {
        "Mode": "MULTI_CLASS",
        "Labels": [
          "ham",
          "spam"
        ]
      }
    },
    "DataLakeS3Uri": "s3://DOC-EXAMPLE-BUCKET/example-flywheel/
schemaVersion=1/20230616T200543Z/",
    "DataSecurityConfig": {},
    "Status": "ACTIVE",
    "ModelType": "DOCUMENT_CLASSIFIER",
    "CreationTime": "2023-06-16T20:05:43.242000+00:00",
    "LastModifiedTime": "2023-06-16T20:21:43.567000+00:00"
  }
}

```

Para obtener más información, consulte [Descripción general de Flywheel](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [DescribeFlywheel](#) de AWS CLI comandos.

## describe-key-phrases-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-key-phrases-detection-job`.

### AWS CLI

Para describir un trabajo de detección de frases clave

En el siguiente `describe-key-phrases-detection-job` ejemplo se obtienen las propiedades de un trabajo de detección de frases clave asíncrono.

```
aws comprehend describe-key-phrases-detection-job \
```

```
--job-id 123456abcdef0e11022f22a11EXAMPLE
```

Salida:

```
{
  "KeyPhrasesDetectionJobProperties": {
    "JobId": "69aa080c00fc68934a6a98f10EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-detection-
job/69aa080c00fc68934a6a98f10EXAMPLE",
    "JobName": "example-key-phrases-detection-job",
    "JobStatus": "COMPLETED",
    "SubmitTime": 1686606439.177,
    "EndTime": 1686606806.157,
    "InputDataConfig": {
      "S3Uri": "s3://dereksbucket1001/EventsData/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://dereksbucket1002/testfolder/111122223333-
KP-69aa080c00fc68934a6a98f10EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-testrole"
  }
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia de [DescribeKeyPhrasesDetectionJob](#) comandos AWS CLI .

## **describe-pii-entities-detection-job**

En el siguiente ejemplo de código, se muestra cómo usar `describe-pii-entities-detection-job`.

AWS CLI

Para describir un trabajo de detección de PII entidades

El siguiente `describe-pii-entities-detection-job` ejemplo obtiene las propiedades de un trabajo de detección de entidades pii asíncronas.

```
aws comprehend describe-pii-entities-detection-job \  
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{  
  "PiiEntitiesDetectionJobProperties": {  
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:pii-entities-detection-  
job/123456abcdeb0e11022f22a11EXAMPLE",  
    "JobName": "example-pii-entities-job",  
    "JobStatus": "IN_PROGRESS",  
    "SubmitTime": "2023-06-08T21:30:15.323000+00:00",  
    "EndTime": "2023-06-08T21:40:23.509000+00:00",  
    "InputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",  
      "InputFormat": "ONE_DOC_PER_LINE"  
    },  
    "OutputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/thefolder/111122223333-  
NER-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"  
    },  
    "LanguageCode": "en",  
    "DataAccessRoleArn": "arn:aws:iam::12345678012:role/service-role/  
AmazonComprehendServiceRole-example-role"  
  }  
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para obtener API más información, consulte la Referencia de comandos [DescribePiiEntitiesDetectionJob](#).AWS CLI

## **describe-resource-policy**

En el siguiente ejemplo de código, se muestra cómo usar `describe-resource-policy`.

## AWS CLI

Para describir una política de recursos adjunta a un modelo

En el siguiente `describe-resource-policy` ejemplo, se obtienen las propiedades de una política basada en recursos adjunta a un modelo.

```
aws comprehend describe-resource-policy \  
  --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier/version/1
```

Salida:

```
{  
  "ResourcePolicy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":  
\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::444455556666:root\"},\"Action\":  
\"comprehend:ImportModel\",\"Resource\":\"*\"}]}\",  
  "CreationTime": "2023-06-19T18:44:26.028000+00:00",  
  "LastModifiedTime": "2023-06-19T18:53:02.002000+00:00",  
  "PolicyRevisionId": "baa675d069d07afaa2aa3106ae280f61"  
}
```

Para obtener más información, consulte [Copiar modelos personalizados entre AWS cuentas](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DescribeResourcePolicy](#) la Referencia de AWS CLI comandos.

## `describe-sentiment-detection-job`

En el siguiente ejemplo de código, se muestra cómo usar `describe-sentiment-detection-job`.

### AWS CLI

Para describir un trabajo de detección de opiniones

En el siguiente `describe-sentiment-detection-job` ejemplo, se obtienen las propiedades de un trabajo de detección de sentimientos asíncrono.

```
aws comprehend describe-sentiment-detection-job \  
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

**Salida:**

```
{
  "SentimentDetectionJobProperties": {
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:sentiment-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobName": "movie_review_analysis",
    "JobStatus": "IN_PROGRESS",
    "SubmitTime": "2023-06-09T23:16:15.956000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/MovieData",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/111122223333-
TS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-servicerole"
  }
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia de [DescribeSentimentDetectionJob](#) comandos AWS CLI .

**describe-targeted-sentiment-detection-job**

En el siguiente ejemplo de código, se muestra cómo usar `describe-targeted-sentiment-detection-job`.

**AWS CLI**

Para describir un trabajo de detección de sentimientos específico

En el siguiente `describe-targeted-sentiment-detection-job` ejemplo, se obtienen las propiedades de un trabajo de detección de opiniones segmentado y asíncrono.

```
aws comprehend describe-targeted-sentiment-detection-job \
```



```
--job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{
  "TargetedSentimentDetectionJobProperties": {
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:targeted-sentiment-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobName": "movie_review_analysis",
    "JobStatus": "IN_PROGRESS",
    "SubmitTime": "2023-06-09T23:16:15.956000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/MovieData",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/111122223333-
TS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-servicerole"
  }
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia de [DescribeTargetedSentimentDetectionJob](#) comandos AWS CLI .

## describe-topics-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-topics-detection-job`.

### AWS CLI

Para describir un trabajo de detección de temas

En el siguiente ejemplo de `describe-topics-detection-job`, se obtienen las propiedades de un trabajo de detección de temas asíncrono.

```
aws comprehend describe-topics-detection-job \
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{
  "TopicsDetectionJobProperties": {
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:topics-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobName": "example_topics_detection",
    "JobStatus": "IN_PROGRESS",
    "SubmitTime": "2023-06-09T18:44:43.414000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/111122223333-
TOPICS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
    },
    "NumberOfTopics": 10,
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-examplerole"
  }
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DescribeTopicsDetectionJob](#) la Referencia de AWS CLI comandos.

## detect-dominant-language

En el siguiente ejemplo de código, se muestra cómo usar detect-dominant-language.

### AWS CLI

Para detectar el idioma dominante del texto de entrada

El siguiente `detect-dominant-language` analiza el texto de entrada e identifica el idioma dominante. También se muestra la puntuación de confianza del modelo previamente entrenado.

```
aws comprehend detect-dominant-language \  
  --text "It is a beautiful day in Seattle."
```

Salida:

```
{  
  "Languages": [  
    {  
      "LanguageCode": "en",  
      "Score": 0.9877256155014038  
    }  
  ]  
}
```

Para obtener más información, consulte [idioma dominante](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DetectDominantLanguage](#) la Referencia de AWS CLI comandos.

## detect-entities

En el siguiente ejemplo de código, se muestra cómo usar `detect-entities`.

AWS CLI

Para detectar entidades con nombre en el texto de entrada

El siguiente ejemplo de `detect-entities` analiza el texto de entrada y devuelve las entidades nombradas. La puntuación de confianza del modelo previamente entrenado también se muestra para cada predicción.

```
aws comprehend detect-entities \  
  --language-code en \  
  --text "Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC  
  credit card \  
  account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July  
  31st. Based on your autopay settings, \  
  "
```

***we will withdraw your payment on the due date from your bank account number XXXXXX1111 with the routing number XXXXX0000. \***  
***Customer feedback for Sunshine Spa, 123 Main St, Anywhere. Send comments to Alice at AnySpa@example.com."***

Salida:

```
{
  "Entities": [
    {
      "Score": 0.9994556307792664,
      "Type": "PERSON",
      "Text": "Zhang Wei",
      "BeginOffset": 6,
      "EndOffset": 15
    },
    {
      "Score": 0.9981022477149963,
      "Type": "PERSON",
      "Text": "John",
      "BeginOffset": 22,
      "EndOffset": 26
    },
    {
      "Score": 0.9986887574195862,
      "Type": "ORGANIZATION",
      "Text": "AnyCompany Financial Services, LLC",
      "BeginOffset": 33,
      "EndOffset": 67
    },
    {
      "Score": 0.9959119558334351,
      "Type": "OTHER",
      "Text": "1111-XXXX-1111-XXXX",
      "BeginOffset": 88,
      "EndOffset": 107
    },
    {
      "Score": 0.9708039164543152,
      "Type": "QUANTITY",
      "Text": ".53",
      "BeginOffset": 133,
      "EndOffset": 136
    }
  ]
}
```

```
  },
  {
    "Score": 0.9987268447875977,
    "Type": "DATE",
    "Text": "July 31st",
    "BeginOffset": 152,
    "EndOffset": 161
  },
  {
    "Score": 0.9858865737915039,
    "Type": "OTHER",
    "Text": "XXXXXX1111",
    "BeginOffset": 271,
    "EndOffset": 281
  },
  {
    "Score": 0.9700471758842468,
    "Type": "OTHER",
    "Text": "XXXXX0000",
    "BeginOffset": 306,
    "EndOffset": 315
  },
  {
    "Score": 0.9591118693351746,
    "Type": "ORGANIZATION",
    "Text": "Sunshine Spa",
    "BeginOffset": 340,
    "EndOffset": 352
  },
  {
    "Score": 0.9797496795654297,
    "Type": "LOCATION",
    "Text": "123 Main St",
    "BeginOffset": 354,
    "EndOffset": 365
  },
  {
    "Score": 0.994929313659668,
    "Type": "PERSON",
    "Text": "Alice",
    "BeginOffset": 394,
    "EndOffset": 399
  },
  {
```

```

        "Score": 0.9949769377708435,
        "Type": "OTHER",
        "Text": "AnySpa@example.com",
        "BeginOffset": 403,
        "EndOffset": 418
    }
]
}

```

Para obtener más información, consulte [Entidades](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DetectEntities](#) la Referencia de AWS CLI comandos.

## detect-key-phrases

En el siguiente ejemplo de código, se muestra cómo usar `detect-key-phrases`.

### AWS CLI

Para detectar frases clave en el texto de entrada

El siguiente ejemplo de `detect-key-phrases` analiza el texto de entrada e identifica las frases nominales clave. La puntuación de confianza del modelo previamente entrenado también se muestra para cada predicción.

```

aws comprehend detect-key-phrases \
  --language-code en \
  --text "Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC credit card \
    account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July 31st. Based on your autopay settings, \
    we will withdraw your payment on the due date from your bank account number XXXXXX1111 with the routing number XXXXX0000. \
    Customer feedback for Sunshine Spa, 123 Main St, Anywhere. Send comments to Alice at AnySpa@example.com."

```

Salida:

```

{
  "KeyPhrases": [

```

```
{
  "Score": 0.8996376395225525,
  "Text": "Zhang Wei",
  "BeginOffset": 6,
  "EndOffset": 15
},
{
  "Score": 0.9992469549179077,
  "Text": "John",
  "BeginOffset": 22,
  "EndOffset": 26
},
{
  "Score": 0.988385021686554,
  "Text": "Your AnyCompany Financial Services",
  "BeginOffset": 28,
  "EndOffset": 62
},
{
  "Score": 0.8740853071212769,
  "Text": "LLC credit card account 1111-XXXX-1111-XXXX",
  "BeginOffset": 64,
  "EndOffset": 107
},
{
  "Score": 0.9999437928199768,
  "Text": "a minimum payment",
  "BeginOffset": 112,
  "EndOffset": 129
},
{
  "Score": 0.9998900890350342,
  "Text": ".53",
  "BeginOffset": 133,
  "EndOffset": 136
},
{
  "Score": 0.9979453086853027,
  "Text": "July 31st",
  "BeginOffset": 152,
  "EndOffset": 161
},
{
  "Score": 0.9983011484146118,
```

```
    "Text": "your autopay settings",
    "BeginOffset": 172,
    "EndOffset": 193
  },
  {
    "Score": 0.9996572136878967,
    "Text": "your payment",
    "BeginOffset": 211,
    "EndOffset": 223
  },
  {
    "Score": 0.9995037317276001,
    "Text": "the due date",
    "BeginOffset": 227,
    "EndOffset": 239
  },
  {
    "Score": 0.9702621698379517,
    "Text": "your bank account number XXXXXX1111",
    "BeginOffset": 245,
    "EndOffset": 280
  },
  {
    "Score": 0.9179925918579102,
    "Text": "the routing number XXXXX0000.Customer feedback",
    "BeginOffset": 286,
    "EndOffset": 332
  },
  {
    "Score": 0.9978160858154297,
    "Text": "Sunshine Spa",
    "BeginOffset": 337,
    "EndOffset": 349
  },
  {
    "Score": 0.9706913232803345,
    "Text": "123 Main St",
    "BeginOffset": 351,
    "EndOffset": 362
  },
  {
    "Score": 0.9941995143890381,
    "Text": "comments",
    "BeginOffset": 379,
```



```
    "EndOffset": 387
  },
  {
    "Score": 0.9759287238121033,
    "Text": "Alice",
    "BeginOffset": 391,
    "EndOffset": 396
  },
  {
    "Score": 0.8376792669296265,
    "Text": "AnySpa@example.com",
    "BeginOffset": 400,
    "EndOffset": 415
  }
]
}
```

Para obtener más información, consulte [Frases clave](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DetectKeyPhrases](#) la Referencia de AWS CLI comandos.

## detect-pii-entities

En el siguiente ejemplo de código, se muestra cómo usar `detect-pii-entities`.

### AWS CLI

Para detectar entidades pii con nombre en el texto de entrada

El siguiente `detect-pii-entities` ejemplo analiza el texto introducido e identifica las entidades que contienen información de identificación personal (PII). La puntuación de confianza del modelo previamente entrenado también se muestra para cada predicción.

```
aws comprehend detect-pii-entities \  
  --language-code en \  
  --text "Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC  
credit card \  
  account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by  
July 31st. Based on your autopay settings, \  
  we will withdraw your payment on the due date from your bank account number  
XXXXXX1111 with the routing number XXXXX0000. \  
"
```

***Customer feedback for Sunshine Spa, 123 Main St, Anywhere. Send comments to Alice at AnySpa@example.com."***

Salida:

```
{
  "Entities": [
    {
      "Score": 0.9998322129249573,
      "Type": "NAME",
      "BeginOffset": 6,
      "EndOffset": 15
    },
    {
      "Score": 0.9998878240585327,
      "Type": "NAME",
      "BeginOffset": 22,
      "EndOffset": 26
    },
    {
      "Score": 0.9994089603424072,
      "Type": "CREDIT_DEBIT_NUMBER",
      "BeginOffset": 88,
      "EndOffset": 107
    },
    {
      "Score": 0.9999760985374451,
      "Type": "DATE_TIME",
      "BeginOffset": 152,
      "EndOffset": 161
    },
    {
      "Score": 0.9999449253082275,
      "Type": "BANK_ACCOUNT_NUMBER",
      "BeginOffset": 271,
      "EndOffset": 281
    },
    {
      "Score": 0.9999847412109375,
      "Type": "BANK_ROUTING",
      "BeginOffset": 306,
      "EndOffset": 315
    }
  ],
}
```

```
{
  "Score": 0.999925434589386,
  "Type": "ADDRESS",
  "BeginOffset": 354,
  "EndOffset": 365
},
{
  "Score": 0.9989161491394043,
  "Type": "NAME",
  "BeginOffset": 394,
  "EndOffset": 399
},
{
  "Score": 0.9994171857833862,
  "Type": "EMAIL",
  "BeginOffset": 403,
  "EndOffset": 418
}
]
```

Para obtener más información, consulte [Información de identificación personal \(PII\)](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DetectPiiEntities](#) la Referencia de AWS CLI comandos.

## detect-sentiment

En el siguiente ejemplo de código, se muestra cómo usar `detect-sentiment`.

### AWS CLI

Para detectar la opinión de un texto de entrada

El siguiente ejemplo de `detect-sentiment` analiza el texto de entrada y devuelve una inferencia de la opinión predominante (POSITIVE, NEUTRAL, MIXED o NEGATIVE).

```
aws comprehend detect-sentiment \
  --language-code en \
  --text "It is a beautiful day in Seattle"
```

Salida:

```
{
  "Sentiment": "POSITIVE",
  "SentimentScore": {
    "Positive": 0.9976957440376282,
    "Negative": 9.653854067437351e-05,
    "Neutral": 0.002169104292988777,
    "Mixed": 3.857641786453314e-05
  }
}
```

Para obtener más información, consulte [Opinión](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DetectSentiment](#) la Referencia de AWS CLI comandos.

## detect-syntax

En el siguiente ejemplo de código, se muestra cómo usar `detect-syntax`.

### AWS CLI

Para detectar las partes del discurso en un texto de entrada

El siguiente ejemplo de `detect-syntax` analiza la sintaxis del texto de entrada y devuelve las distintas partes de la oración. La puntuación de confianza del modelo previamente entrenado también se muestra para cada predicción.

```
aws comprehend detect-syntax \
  --language-code en \
  --text "It is a beautiful day in Seattle."
```

Salida:

```
{
  "SyntaxTokens": [
    {
      "TokenId": 1,
      "Text": "It",
      "BeginOffset": 0,
      "EndOffset": 2,
      "PartOfSpeech": {
```

```
        "Tag": "PRON",
        "Score": 0.9999740719795227
    }
},
{
    "TokenId": 2,
    "Text": "is",
    "BeginOffset": 3,
    "EndOffset": 5,
    "PartOfSpeech": {
        "Tag": "VERB",
        "Score": 0.999901294708252
    }
},
{
    "TokenId": 3,
    "Text": "a",
    "BeginOffset": 6,
    "EndOffset": 7,
    "PartOfSpeech": {
        "Tag": "DET",
        "Score": 0.9999938607215881
    }
},
{
    "TokenId": 4,
    "Text": "beautiful",
    "BeginOffset": 8,
    "EndOffset": 17,
    "PartOfSpeech": {
        "Tag": "ADJ",
        "Score": 0.9987351894378662
    }
},
{
    "TokenId": 5,
    "Text": "day",
    "BeginOffset": 18,
    "EndOffset": 21,
    "PartOfSpeech": {
        "Tag": "NOUN",
        "Score": 0.9999796748161316
    }
},
},
```

```
    {
      "TokenId": 6,
      "Text": "in",
      "BeginOffset": 22,
      "EndOffset": 24,
      "PartOfSpeech": {
        "Tag": "ADP",
        "Score": 0.9998047947883606
      }
    },
    {
      "TokenId": 7,
      "Text": "Seattle",
      "BeginOffset": 25,
      "EndOffset": 32,
      "PartOfSpeech": {
        "Tag": "PROPN",
        "Score": 0.9940530061721802
      }
    }
  ]
}
```

Para obtener más información, consulte los [Análisis de sintaxis](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DetectSyntax](#) la Referencia de AWS CLI comandos.

## **detect-targeted-sentiment**

En el siguiente ejemplo de código, se muestra cómo usar `detect-targeted-sentiment`.

### AWS CLI

Para detectar la opinión específica de las entidades nombradas en un texto introducido

El siguiente `detect-targeted-sentiment` ejemplo analiza el texto introducido y devuelve las entidades nombradas, además de la opinión objetivo asociada a cada entidad. También se muestra la puntuación de confianza de los modelos previamente entrenados para cada predicción.

```
aws comprehend detect-targeted-sentiment \
```

```
--language-code en \  
--text "I do not enjoy January because it is too cold but August is the perfect  
temperature"
```

Salida:

```
{  
  "Entities": [  
    {  
      "DescriptiveMentionIndex": [  
        0  
      ],  
      "Mentions": [  
        {  
          "Score": 0.9999979734420776,  
          "GroupScore": 1.0,  
          "Text": "I",  
          "Type": "PERSON",  
          "MentionSentiment": {  
            "Sentiment": "NEUTRAL",  
            "SentimentScore": {  
              "Positive": 0.0,  
              "Negative": 0.0,  
              "Neutral": 1.0,  
              "Mixed": 0.0  
            }  
          },  
          "BeginOffset": 0,  
          "EndOffset": 1  
        }  
      ]  
    },  
    {  
      "DescriptiveMentionIndex": [  
        0  
      ],  
      "Mentions": [  
        {  
          "Score": 0.9638869762420654,  
          "GroupScore": 1.0,  
          "Text": "January",  
          "Type": "DATE",  
          "MentionSentiment": {
```

```
        "Sentiment": "NEGATIVE",
        "SentimentScore": {
            "Positive": 0.0031610000878572464,
            "Negative": 0.9967250227928162,
            "Neutral": 0.00011100000119768083,
            "Mixed": 1.9999999949504854e-06
        }
    },
    "BeginOffset": 15,
    "EndOffset": 22
}
]
},
{
    "DescriptiveMentionIndex": [
        0
    ],
    "Mentions": [
        {
            "Score": 0.9664419889450073,
            "GroupScore": 1.0,
            "Text": "August",
            "Type": "DATE",
            "MentionSentiment": {
                "Sentiment": "POSITIVE",
                "SentimentScore": {
                    "Positive": 0.9999549984931946,
                    "Negative": 3.999999989900971e-06,
                    "Neutral": 4.099999932805076e-05,
                    "Mixed": 0.0
                }
            }
        },
        {
            "BeginOffset": 50,
            "EndOffset": 56
        }
    ]
}
},
{
    "DescriptiveMentionIndex": [
        0
    ],
    "Mentions": [
        {
```



```

        "Score": 0.9803199768066406,
        "GroupScore": 1.0,
        "Text": "temperature",
        "Type": "ATTRIBUTE",
        "MentionSentiment": {
            "Sentiment": "POSITIVE",
            "SentimentScore": {
                "Positive": 1.0,
                "Negative": 0.0,
                "Neutral": 0.0,
                "Mixed": 0.0
            }
        },
        "BeginOffset": 77,
        "EndOffset": 88
    }
]
}

```

Para obtener más información, consulte [Targeted Sentiment](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [DetectTargetedSentiment](#) la Referencia de AWS CLI comandos.

## import-model

En el siguiente ejemplo de código, se muestra cómo usar `import-model`.

### AWS CLI

Para importar un modelo

En el siguiente `import-model` ejemplo, se importa un modelo de una AWS cuenta diferente. El modelo de clasificador de documentos de la cuenta 444455556666 tiene una política basada en los recursos que permite 111122223333 a la cuenta importar el modelo.

```

aws comprehend import-model \
  --source-model-arn arn:aws:comprehend:us-west-2:444455556666:document-
  classifier/example-classifier

```

Salida:

```
{
  "ModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier"
}
```

Para obtener más información, consulte [Copiar modelos personalizados entre AWS cuentas](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [ImportModel](#) la Referencia de AWS CLI comandos.

## list-datasets

En el siguiente ejemplo de código, se muestra cómo usar `list-datasets`.

AWS CLI

Para enumerar todos los conjuntos de datos del volante

En el siguiente `list-datasets` ejemplo, se enumeran todos los conjuntos de datos asociados a un volante.

```
aws comprehend list-datasets \
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-
entity
```

Salida:

```
{
  "DatasetPropertiesList": [
    {
      "DatasetArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
flywheel-entity/dataset/example-dataset-1",
      "DatasetName": "example-dataset-1",
      "DatasetType": "TRAIN",
      "DatasetS3Uri": "s3://DOC-EXAMPLE-BUCKET/flywheel-entity/
schemaVersion=1/20230616T200543Z/datasets/example-dataset-1/20230616T203710Z/",
      "Status": "CREATING",
      "CreationTime": "2023-06-16T20:37:10.400000+00:00"
    },
    {
```

```

        "DatasetArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
flywheel-entity/dataset/example-dataset-2",
        "DatasetName": "example-dataset-2",
        "DatasetType": "TRAIN",
        "DatasetS3Uri": "s3://DOC-EXAMPLE-BUCKET/flywheel-entity/
schemaVersion=1/20230616T200543Z/datasets/example-dataset-2/20230616T200607Z/",
        "Description": "TRAIN Dataset created by Flywheel creation.",
        "Status": "COMPLETED",
        "NumberOfDocuments": 5572,
        "CreationTime": "2023-06-16T20:06:07.722000+00:00"
    }
]
}

```

Para obtener más información, consulte [Descripción general de Flywheel](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [ListDatasets](#) de AWS CLI comandos.

## list-document-classification-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-document-classification-jobs`.

### AWS CLI

Para enumerar todos los trabajos de clasificación de documentos

El siguiente ejemplo de `list-document-classification-jobs` enumera todos los trabajos de clasificación de documentos.

```
aws comprehend list-document-classification-jobs
```

Salida:

```

{
  "DocumentClassificationJobPropertiesList": [
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:1234567890101:document-
classification-job/123456abcdeb0e11022f22a11EXAMPLE",
      "JobName": "exampleclassificationjob",
      "JobStatus": "COMPLETED",

```

```

    "SubmitTime": "2023-06-14T17:09:51.788000+00:00",
    "EndTime": "2023-06-14T17:15:58.582000+00:00",
    "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:1234567890101:document-classifier/mymodel/version/12",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/jobdata/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/1234567890101-CLN-e758dd56b824aa717ceab551f11749fb/output/output.tar.gz"
    },
    "DataAccessRoleArn": "arn:aws:iam::1234567890101:role/service-role/
AmazonComprehendServiceRole-example-role"
  },
  {
    "JobId": "123456abcdeb0e11022f22a1EXAMPLE2",
    "JobArn": "arn:aws:comprehend:us-west-2:1234567890101:document-
classification-job/123456abcdeb0e11022f22a1EXAMPLE2",
    "JobName": "exampleclassificationjob2",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2023-06-14T17:22:39.829000+00:00",
    "EndTime": "2023-06-14T17:28:46.107000+00:00",
    "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:1234567890101:document-classifier/mymodel/version/12",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/jobdata/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/1234567890101-CLN-123456abcdeb0e11022f22a1EXAMPLE2/output/output.tar.gz"
    },
    "DataAccessRoleArn": "arn:aws:iam::1234567890101:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
]
}

```

Para obtener más información, consulte [Clasificación personalizada](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [ListDocumentClassificationJobs](#) la Referencia de AWS CLI comandos.

## list-document-classifier-summaries

En el siguiente ejemplo de código, se muestra cómo usar `list-document-classifier-summaries`.

### AWS CLI

Para enumerar los resúmenes de todos los clasificadores de documentos creados

En el siguiente `list-document-classifier-summaries` ejemplo se enumeran todos los resúmenes de los clasificadores de documentos creados.

```
aws comprehend list-document-classifier-summaries
```

Salida:

```
{
  "DocumentClassifierSummariesList": [
    {
      "DocumentClassifierName": "example-classifier-1",
      "NumberOfVersions": 1,
      "LatestVersionCreatedAt": "2023-06-13T22:07:59.825000+00:00",
      "LatestVersionName": "1",
      "LatestVersionStatus": "TRAINED"
    },
    {
      "DocumentClassifierName": "example-classifier-2",
      "NumberOfVersions": 2,
      "LatestVersionCreatedAt": "2023-06-13T21:54:59.589000+00:00",
      "LatestVersionName": "2",
      "LatestVersionStatus": "TRAINED"
    }
  ]
}
```

Para obtener más información, consulte [Creación y administración de modelos personalizados](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [ListDocumentClassifierSummaries](#) de AWS CLI comandos.

## list-document-classifiers

En el siguiente ejemplo de código, se muestra cómo usar `list-document-classifiers`.

### AWS CLI

Para enumerar todos los clasificadores de documentos

El siguiente ejemplo de `list-document-classifiers` enumera todos los modelos de clasificadores de documentos entrenados y en proceso de entrenamiento.

```
aws comprehend list-document-classifiers
```

Salida:

```
{
  "DocumentClassifierPropertiesList": [
    {
      "DocumentClassifierArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/exampleclassifier1",
      "LanguageCode": "en",
      "Status": "TRAINED",
      "SubmitTime": "2023-06-13T19:04:15.735000+00:00",
      "EndTime": "2023-06-13T19:42:31.752000+00:00",
      "TrainingStartTime": "2023-06-13T19:08:20.114000+00:00",
      "TrainingEndTime": "2023-06-13T19:41:35.080000+00:00",
      "InputDataConfig": {
        "DataFormat": "COMPREHEND_CSV",
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata"
      },
      "OutputDataConfig": {},
      "ClassifierMetadata": {
        "NumberOfLabels": 3,
        "NumberOfTrainedDocuments": 5016,
        "NumberOfTestDocuments": 557,
        "EvaluationMetrics": {
          "Accuracy": 0.9856,
          "Precision": 0.9919,
          "Recall": 0.9459,
          "F1Score": 0.9673,
          "MicroPrecision": 0.9856,
          "MicroRecall": 0.9856,
          "MicroF1Score": 0.9856,
        }
      }
    }
  ]
}
```

```

        "HammingLoss": 0.0144
      }
    },
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-testorle",
    "Mode": "MULTI_CLASS"
  },
  {
    "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/exampleclassifier2",
    "LanguageCode": "en",
    "Status": "TRAINING",
    "SubmitTime": "2023-06-13T21:20:28.690000+00:00",
    "InputDataConfig": {
      "DataFormat": "COMPREHEND_CSV",
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata"
    },
    "OutputDataConfig": {},
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-testorle",
    "Mode": "MULTI_CLASS"
  }
]
}

```

Para obtener más información, consulte [Creación y administración de modelos personalizados](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [ListDocumentClassifiers](#) la Referencia de AWS CLI comandos.

## list-dominant-language-detection-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-dominant-language-detection-jobs`.

### AWS CLI

Para enumerar todos los trabajos de detección de idiomas dominantes

En el siguiente `list-dominant-language-detection-jobs` ejemplo, se enumeran todos los trabajos de detección de lenguaje dominante asíncronos en curso y finalizados.

**aws comprehend list-dominant-language-detection-jobs**

Salida:

```
{
  "DominantLanguageDetectionJobPropertiesList": [
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:dominant-language-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
      "JobName": "languageanalysis1",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-09T18:10:38.037000+00:00",
      "EndTime": "2023-06-09T18:18:45.498000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-LANGUAGE-123456abcdeb0e11022f22a11EXAMPLE/output/
output.tar.gz"
      },
      "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
    },
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:dominant-language-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
      "JobName": "languageanalysis2",
      "JobStatus": "STOPPED",
      "SubmitTime": "2023-06-09T18:16:33.690000+00:00",
      "EndTime": "2023-06-09T18:24:40.608000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-LANGUAGE-123456abcdeb0e11022f22a11EXAMPLE/output/
output.tar.gz"
      },
    }
  ]
}
```



```
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
    }
]
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para obtener API más información, consulte la Referencia de comandos [ListDominantLanguageDetectionJobs](#).AWS CLI

## list-endpoints

En el siguiente ejemplo de código, se muestra cómo usar `list-endpoints`.

### AWS CLI

Para ver una lista de todos los puntos finales

En el siguiente `list-endpoints` ejemplo, se enumeran todos los puntos finales activos específicos del modelo.

```
aws comprehend list-endpoints
```

Salida:

```
{
  "EndpointPropertiesList": [
    {
      "EndpointArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier-endpoint/ExampleClassifierEndpoint",
      "Status": "IN_SERVICE",
      "ModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier1",
      "DesiredModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier1",
      "DesiredInferenceUnits": 1,
      "CurrentInferenceUnits": 1,
      "CreationTime": "2023-06-13T20:32:54.526000+00:00",
      "LastModifiedTime": "2023-06-13T20:32:54.526000+00:00"
    },
    {
```

```

        "EndpointArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier-endpoint/ExampleClassifierEndpoint2",
        "Status": "IN_SERVICE",
        "ModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier2",
        "DesiredModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier2",
        "DesiredInferenceUnits": 1,
        "CurrentInferenceUnits": 1,
        "CreationTime": "2023-06-13T20:32:54.526000+00:00",
        "LastModifiedTime": "2023-06-13T20:32:54.526000+00:00"
    }
]
}

```

Para obtener más información, consulte [Administración de puntos de conexión de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia de [ListEndpoints](#) comandos AWS CLI .

## list-entities-detection-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-entities-detection-jobs`.

### AWS CLI

Para enumerar todos los trabajos de detección de entidades

En el siguiente `list-entities-detection-jobs` ejemplo, se enumeran todos los trabajos de detección de entidades asíncronas.

```
aws comprehend list-entities-detection-jobs
```

Salida:

```

{
  "EntitiesDetectionJobPropertiesList": [
    {
      "JobId": "468af39c28ab45b83eb0c4ab9EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-
job/468af39c28ab45b83eb0c4ab9EXAMPLE",
      "JobName": "example-entities-detection",

```

```
"JobStatus": "COMPLETED",
"SubmitTime": "2023-06-08T20:57:46.476000+00:00",
"EndTime": "2023-06-08T21:05:53.718000+00:00",
"InputDataConfig": {
  "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
  "InputFormat": "ONE_DOC_PER_LINE"
},
"OutputDataConfig": {
  "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-NER-468af39c28ab45b83eb0c4ab9EXAMPLE/output/output.tar.gz"
},
"LanguageCode": "en",
"DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
},
{
  "JobId": "809691caeaab0e71406f80a28EXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-
job/809691caeaab0e71406f80a28EXAMPLE",
  "JobName": "example-entities-detection-2",
  "JobStatus": "COMPLETED",
  "SubmitTime": "2023-06-08T21:30:15.323000+00:00",
  "EndTime": "2023-06-08T21:40:23.509000+00:00",
  "InputDataConfig": {
    "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
    "InputFormat": "ONE_DOC_PER_LINE"
  },
  "OutputDataConfig": {
    "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-NER-809691caeaab0e71406f80a28EXAMPLE/output/output.tar.gz"
  },
  "LanguageCode": "en",
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
},
{
  "JobId": "e00597c36b448b91d70dea165EXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-
job/e00597c36b448b91d70dea165EXAMPLE",
  "JobName": "example-entities-detection-3",
  "JobStatus": "STOPPED",
  "SubmitTime": "2023-06-08T22:19:28.528000+00:00",
  "EndTime": "2023-06-08T22:27:33.991000+00:00",
  "InputDataConfig": {
```

```

        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
        "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-NER-e00597c36b448b91d70dea165EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
    }
]
}

```

Para obtener más información, consulte [Entidades](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia de [ListEntitiesDetectionJobs](#) comandos AWS CLI .

## list-entity-recognizer-summaries

En el siguiente ejemplo de código, se muestra cómo usar `list-entity-recognizer-summaries`.

### AWS CLI

Para ver una lista de resúmenes de todos los reconocedores de entidades creados

En el siguiente `list-entity-recognizer-summaries` ejemplo, se enumeran todos los resúmenes de los reconocedores de entidades.

```
aws comprehend list-entity-recognizer-summaries
```

Salida:

```

{
  "EntityRecognizerSummariesList": [
    {
      "RecognizerName": "entity-recognizer-3",
      "NumberOfVersions": 2,
      "LatestVersionCreatedAt": "2023-06-15T23:15:07.621000+00:00",

```

```

    "LatestVersionName": "2",
    "LatestVersionStatus": "STOP_REQUESTED"
  },
  {
    "RecognizerName": "entity-recognizer-2",
    "NumberOfVersions": 1,
    "LatestVersionCreatedAt": "2023-06-14T22:55:27.805000+00:00",
    "LatestVersionName": "2"
    "LatestVersionStatus": "TRAINED"
  },
  {
    "RecognizerName": "entity-recognizer-1",
    "NumberOfVersions": 1,
    "LatestVersionCreatedAt": "2023-06-14T20:44:59.631000+00:00",
    "LatestVersionName": "1",
    "LatestVersionStatus": "TRAINED"
  }
]
}

```

Para obtener más información, consulte [Reconocimiento de entidades personalizado](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [ListEntityRecognizerSummaries](#) la Referencia de AWS CLI comandos.

## list-entity-recognizers

En el siguiente ejemplo de código, se muestra cómo usar `list-entity-recognizers`.

### AWS CLI

Para ver una lista de todos los reconocedores de entidades personalizados

En el siguiente `list-entity-recognizers` ejemplo se enumeran todos los reconocedores de entidades personalizados creados.

```
aws comprehend list-entity-recognizers
```

Salida:

```
{
  "EntityRecognizerPropertiesList": [
```

```
{
  "EntityRecognizerArn": "arn:aws:comprehend:us-
west-2:111122223333:entity-recognizer/EntityRecognizer/version/1",
  "LanguageCode": "en",
  "Status": "TRAINED",
  "SubmitTime": "2023-06-14T20:44:59.631000+00:00",
  "EndTime": "2023-06-14T20:59:19.532000+00:00",
  "TrainingStartTime": "2023-06-14T20:48:52.811000+00:00",
  "TrainingEndTime": "2023-06-14T20:58:11.473000+00:00",
  "InputDataConfig": {
    "DataFormat": "COMPREHEND_CSV",
    "EntityTypes": [
      {
        "Type": "BUSINESS"
      }
    ],
    "Documents": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata/dataset/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "EntityList": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata/entity.csv"
    }
  },
  "RecognizerMetadata": {
    "NumberOfTrainedDocuments": 1814,
    "NumberOfTestDocuments": 486,
    "EvaluationMetrics": {
      "Precision": 100.0,
      "Recall": 100.0,
      "F1Score": 100.0
    },
    "EntityTypes": [
      {
        "Type": "BUSINESS",
        "EvaluationMetrics": {
          "Precision": 100.0,
          "Recall": 100.0,
          "F1Score": 100.0
        },
        "NumberOfTrainMentions": 1520
      }
    ]
  },
}
```

```
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-servicerole",
    "VersionName": "1"
  },
  {
    "EntityRecognizerArn": "arn:aws:comprehend:us-
west-2:111122223333:entity-recognizer/entityrecognizer3",
    "LanguageCode": "en",
    "Status": "TRAINED",
    "SubmitTime": "2023-06-14T22:57:51.056000+00:00",
    "EndTime": "2023-06-14T23:14:13.894000+00:00",
    "TrainingStartTime": "2023-06-14T23:01:33.984000+00:00",
    "TrainingEndTime": "2023-06-14T23:13:02.984000+00:00",
    "InputDataConfig": {
      "DataFormat": "COMPREHEND_CSV",
      "EntityTypes": [
        {
          "Type": "DEVICE"
        }
      ],
      "Documents": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata/raw_txt.csv",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "EntityList": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/trainingdata/entity_list.csv"
      }
    },
    "RecognizerMetadata": {
      "NumberOfTrainedDocuments": 4616,
      "NumberOfTestDocuments": 3489,
      "EvaluationMetrics": {
        "Precision": 98.54227405247813,
        "Recall": 100.0,
        "F1Score": 99.26578560939794
      }
    },
    "EntityTypes": [
      {
        "Type": "DEVICE",
        "EvaluationMetrics": {
          "Precision": 98.54227405247813,
          "Recall": 100.0,
          "F1Score": 99.26578560939794
        }
      }
    ],
  },
}
```

```

        "NumberOfTrainMentions": 2764
      }
    ]
  },
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-servicerole"
}
]
}

```

Para obtener más información, consulte [Reconocimiento de entidades personalizado](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [ListEntityRecognizers](#) la Referencia de AWS CLI comandos.

## list-events-detection-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-events-detection-jobs`.

### AWS CLI

Para ver una lista de todos los trabajos de detección de eventos

En el siguiente `list-events-detection-jobs` ejemplo, se enumeran todos los trabajos de detección de eventos asíncronos.

```
aws comprehend list-events-detection-jobs
```

Salida:

```

{
  "EventsDetectionJobPropertiesList": [
    {
      "JobId": "aa9593f9203e84f3ef032ce18EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:events-detection-
job/aa9593f9203e84f3ef032ce18EXAMPLE",
      "JobName": "events_job_1",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-12T19:14:57.751000+00:00",
      "EndTime": "2023-06-12T19:21:04.962000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-SOURCE-BUCKET/EventsData/",

```



```
        "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/1111222233333-EVENTS-aa9593f9203e84f3ef032ce18EXAMPLE/output/"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::1111222233333:role/service-role/
AmazonComprehendServiceRole-example-role",
    "TargetEventTypes": [
        "BANKRUPTCY",
        "EMPLOYMENT",
        "CORPORATE_ACQUISITION",
        "CORPORATE_MERGER",
        "INVESTMENT_GENERAL"
    ]
},
{
    "JobId": "4a990a2f7e82adfca6e171135EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:1111222233333:events-detection-
job/4a990a2f7e82adfca6e171135EXAMPLE",
    "JobName": "events_job_2",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2023-06-12T19:55:43.702000+00:00",
    "EndTime": "2023-06-12T20:03:49.893000+00:00",
    "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-SOURCE-BUCKET/EventsData/",
        "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/1111222233333-EVENTS-4a990a2f7e82adfca6e171135EXAMPLE/output/"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::1111222233333:role/service-role/
AmazonComprehendServiceRole-example-role",
    "TargetEventTypes": [
        "BANKRUPTCY",
        "EMPLOYMENT",
        "CORPORATE_ACQUISITION",
        "CORPORATE_MERGER",
        "INVESTMENT_GENERAL"
    ]
}
}
```

```
]
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia de [ListEventsDetectionJobs](#) comandos AWS CLI .

## list-flywheel-iteration-history

En el siguiente ejemplo de código, se muestra cómo usar `list-flywheel-iteration-history`.

### AWS CLI

Para ver todo el historial de iteraciones del volante

En el siguiente `list-flywheel-iteration-history` ejemplo, se enumeran todas las iteraciones de un volante.

```
aws comprehend list-flywheel-iteration-history
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel
```

Salida:

```
{
  "FlywheelIterationPropertiesList": [
    {
      "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel",
      "FlywheelIterationId": "20230619EXAMPLE",
      "CreationTime": "2023-06-19T04:00:32.594000+00:00",
      "EndTime": "2023-06-19T04:00:49.248000+00:00",
      "Status": "COMPLETED",
      "Message": "FULL_ITERATION: Flywheel iteration performed all functions successfully.",
      "EvaluatedModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/example-classifier/version/1",
      "EvaluatedModelMetrics": {
        "AverageF1Score": 0.7742663922375772,
        "AverageF1Score": 0.9876464664646313,
        "AveragePrecision": 0.9800000253081214,
```

```

        "AverageRecall": 0.9445600253081214,
        "AverageAccuracy": 0.9997281665190434
    },
    "EvaluationManifestS3Prefix": "s3://DOC-EXAMPLE-BUCKET/example-flywheel/
schemaVersion=1/20230619TEXAMPLE/evaluation/20230619TEXAMPLE/"
    },
    {
        "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
example-flywheel-2",
        "FlywheelIterationId": "20230616TEXAMPLE",
        "CreationTime": "2023-06-16T21:10:26.385000+00:00",
        "EndTime": "2023-06-16T23:33:16.827000+00:00",
        "Status": "COMPLETED",
        "Message": "FULL_ITERATION: Flywheel iteration performed all functions
successfully.",
        "EvaluatedModelArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/spamvshamclassify/version/1",
        "EvaluatedModelMetrics": {
            "AverageF1Score": 0.7742663922375772,
            "AverageF1Score": 0.9767700253081214,
            "AveragePrecision": 0.9767700253081214,
            "AverageRecall": 0.9767700253081214,
            "AverageAccuracy": 0.9858281665190434
        },
        "EvaluationManifestS3Prefix": "s3://DOC-EXAMPLE-BUCKET/example-
flywheel-2/schemaVersion=1/20230616TEXAMPLE/evaluation/20230616TEXAMPLE/"
    }
]
}

```

Para obtener más información, consulte la [descripción general de Flywheel](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [ListFlywheelIterationHistory](#) de AWS CLI comandos.

## list-flywheels

En el siguiente ejemplo de código, se muestra cómo usar `list-flywheels`.

### AWS CLI

Para ver una lista de todos los volantes

En el `list-flywheels` ejemplo siguiente se enumeran todos los volantes creados.

```
aws comprehend list-flywheels
```

Salida:

```
{
  "FlywheelSummaryList": [
    {
      "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel-1",
      "ActiveModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/exampleclassifier/version/1",
      "DataLakeS3Uri": "s3://DOC-EXAMPLE-BUCKET/example-flywheel-1/schemaVersion=1/20230616T200543Z/",
      "Status": "ACTIVE",
      "ModelType": "DOCUMENT_CLASSIFIER",
      "CreationTime": "2023-06-16T20:05:43.242000+00:00",
      "LastModifiedTime": "2023-06-19T04:00:43.027000+00:00",
      "LatestFlywheelIteration": "20230619T040032Z"
    },
    {
      "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel-2",
      "ActiveModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/exampleclassifier2/version/1",
      "DataLakeS3Uri": "s3://DOC-EXAMPLE-BUCKET/example-flywheel-2/schemaVersion=1/20220616T200543Z/",
      "Status": "ACTIVE",
      "ModelType": "DOCUMENT_CLASSIFIER",
      "CreationTime": "2022-06-16T20:05:43.242000+00:00",
      "LastModifiedTime": "2022-06-19T04:00:43.027000+00:00",
      "LatestFlywheelIteration": "20220619T040032Z"
    }
  ]
}
```

Para obtener más información, consulte la [descripción general de Flywheel](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [ListFlywheels](#) de AWS CLI comandos.

## list-key-phrases-detection-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-key-phrases-detection-jobs`.

### AWS CLI

Para enumerar todos los trabajos de detección de frases clave

En el siguiente `list-key-phrases-detection-jobs` ejemplo se enumeran todos los trabajos de detección de frases clave asíncronas en curso y finalizados.

```
aws comprehend list-key-phrases-detection-jobs
```

Salida:

```
{
  "KeyPhrasesDetectionJobPropertiesList": [
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
      "JobName": "keyphrasesanalysis1",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-08T22:31:43.767000+00:00",
      "EndTime": "2023-06-08T22:39:52.565000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-SOURCE-BUCKET/AsyncBatchJobs/",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-KP-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
      },
      "LanguageCode": "en",
      "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
    },
    {
      "JobId": "123456abcdeb0e11022f22a33EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-
detection-job/123456abcdeb0e11022f22a33EXAMPLE",
      "JobName": "keyphrasesanalysis2",
      "JobStatus": "STOPPED",
```

```

    "SubmitTime": "2023-06-08T22:57:52.154000+00:00",
    "EndTime": "2023-06-08T23:05:48.385000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-KP-123456abcdeb0e11022f22a33EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
  },
  {
    "JobId": "123456abcdeb0e11022f22a44EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-
detection-job/123456abcdeb0e11022f22a44EXAMPLE",
    "JobName": "keyphrasesanalysis3",
    "JobStatus": "FAILED",
    "Message": "NO_READ_ACCESS_TO_INPUT: The provided data access role does
not have proper access to the input data.",
    "SubmitTime": "2023-06-09T16:47:04.029000+00:00",
    "EndTime": "2023-06-09T16:47:18.413000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-KP-123456abcdeb0e11022f22a44EXAMPLE/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
]
}

```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para obtener API más información, consulte la Referencia de comandos [ListKeyPhrasesDetectionJobs.AWS CLI](#)

## list-pii-entities-detection-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-pii-entities-detection-jobs`.

### AWS CLI

Para enumerar todos los trabajos de detección de entidades pii

En el siguiente `list-pii-entities-detection-jobs` ejemplo, se enumeran todos los trabajos de detección asíncrona de pii en curso y finalizados.

```
aws comprehend list-pii-entities-detection-jobs
```

Salida:

```
{
  "PiiEntitiesDetectionJobPropertiesList": [
    {
      "JobId": "6f9db0c42d0c810e814670ee4EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:pii-entities-
detection-job/6f9db0c42d0c810e814670ee4EXAMPLE",
      "JobName": "example-pii-detection-job",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-09T21:02:46.241000+00:00",
      "EndTime": "2023-06-09T21:12:52.602000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-SOURCE-BUCKET/111122223333-
PII-6f9db0c42d0c810e814670ee4EXAMPLE/output/"
      },
      "LanguageCode": "en",
      "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
      "Mode": "ONLY_OFFSETS"
    },
    {
      "JobId": "d927562638cfa739331a99b3cEXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:pii-entities-
detection-job/d927562638cfa739331a99b3cEXAMPLE",
```

```

    "JobName": "example-pii-detection-job-2",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2023-06-09T21:20:58.211000+00:00",
    "EndTime": "2023-06-09T21:31:06.027000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/AsyncBatchJobs/",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-PII-d927562638cfa739331a99b3cEXAMPLE/output/"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
    "Mode": "ONLY_OFFSETS"
  }
]
}

```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para obtener API más información, consulte la Referencia de comandos. [ListPiiEntitiesDetectionJobs](#) AWS CLI

## list-sentiment-detection-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-sentiment-detection-jobs`.

AWS CLI

Para ver una lista de todos los trabajos de detección de opiniones

En el siguiente `list-sentiment-detection-jobs` ejemplo, se enumeran todos los trabajos de detección de sentimientos asíncronos en curso y finalizados.

```
aws comprehend list-sentiment-detection-jobs
```

Salida:

```
{
  "SentimentDetectionJobPropertiesList": [
```



```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:sentiment-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
  "JobName": "example-sentiment-detection-job",
  "JobStatus": "IN_PROGRESS",
  "SubmitTime": "2023-06-09T22:42:20.545000+00:00",
  "EndTime": "2023-06-09T22:52:27.416000+00:00",
  "InputDataConfig": {
    "S3Uri": "s3://DOC-EXAMPLE-BUCKET/MovieData",
    "InputFormat": "ONE_DOC_PER_LINE"
  },
  "OutputDataConfig": {
    "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-TS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
  },
  "LanguageCode": "en",
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
},
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE2",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:sentiment-
detection-job/123456abcdeb0e11022f22a11EXAMPLE2",
  "JobName": "example-sentiment-detection-job-2",
  "JobStatus": "COMPLETED",
  "SubmitTime": "2023-06-09T23:16:15.956000+00:00",
  "EndTime": "2023-06-09T23:26:00.168000+00:00",
  "InputDataConfig": {
    "S3Uri": "s3://DOC-EXAMPLE-BUCKET/MovieData2",
    "InputFormat": "ONE_DOC_PER_LINE"
  },
  "OutputDataConfig": {
    "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-TS-123456abcdeb0e11022f22a11EXAMPLE2/output/output.tar.gz"
  },
  "LanguageCode": "en",
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
}
]
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para obtener API más información, consulte la Referencia de comandos [ListSentimentDetectionJobs](#).AWS CLI

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas del recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas de un recurso de Amazon Comprehend.

```
aws comprehend list-tags-for-resource \  
  --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier/version/1
```

Salida:

```
{  
  "ResourceArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier/version/1",  
  "Tags": [  
    {  
      "Key": "Department",  
      "Value": "Finance"  
    },  
    {  
      "Key": "location",  
      "Value": "Seattle"  
    }  
  ]  
}
```

Para obtener más información, consulte [Etiquetar sus recursos](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [ListTagsForResource](#) de AWS CLI comandos.

## list-targeted-sentiment-detection-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-targeted-sentiment-detection-jobs`.

### AWS CLI

Para enumerar todos los trabajos de detección de sentimientos específicos

En el siguiente `list-targeted-sentiment-detection-jobs` ejemplo, se enumeran todos los trabajos de detección de sentimientos segmentados asíncronos en curso y finalizados.

```
aws comprehend list-targeted-sentiment-detection-jobs
```

Salida:

```
{
  "TargetedSentimentDetectionJobPropertiesList": [
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:targeted-sentiment-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
      "JobName": "example-targeted-sentiment-detection-job",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-09T22:42:20.545000+00:00",
      "EndTime": "2023-06-09T22:52:27.416000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/MovieData",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-TS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
      },
      "LanguageCode": "en",
      "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-I0role"
    },
    {
```

```

    "JobId": "123456abcdeb0e11022f22a1EXAMPLE2",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:targeted-sentiment-
detection-job/123456abcdeb0e11022f22a1EXAMPLE2",
    "JobName": "example-targeted-sentiment-detection-job-2",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2023-06-09T23:16:15.956000+00:00",
    "EndTime": "2023-06-09T23:26:00.168000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/MovieData2",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
testfolder/111122223333-TS-123456abcdeb0e11022f22a1EXAMPLE2/output/output.tar.gz"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
]
}

```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para obtener API más información, consulte la Referencia de comandos [ListTargetedSentimentDetectionJobs](#).AWS CLI

## list-topics-detection-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-topics-detection-jobs`.

### AWS CLI

Para enumerar todos los trabajos de detección de temas

El siguiente ejemplo de `list-topics-detection-jobs`, se enumeran todos los trabajos de detección de temas asíncronos en curso y finalizados.

```
aws comprehend list-topics-detection-jobs
```

Salida:

```
{
  "TopicsDetectionJobPropertiesList": [
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:topics-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
      "JobName": "topic-analysis-1"
      "JobStatus": "IN_PROGRESS",
      "SubmitTime": "2023-06-09T18:40:35.384000+00:00",
      "EndTime": "2023-06-09T18:46:41.936000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-TOPICS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
      },
      "NumberOfTopics": 10,
      "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
    },
    {
      "JobId": "123456abcdeb0e11022f22a11EXAMPLE2",
      "JobArn": "arn:aws:comprehend:us-west-2:111122223333:topics-detection-
job/123456abcdeb0e11022f22a11EXAMPLE2",
      "JobName": "topic-analysis-2",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2023-06-09T18:44:43.414000+00:00",
      "EndTime": "2023-06-09T18:50:50.872000+00:00",
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
        "InputFormat": "ONE_DOC_PER_LINE"
      },
      "OutputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-TOPICS-123456abcdeb0e11022f22a11EXAMPLE2/output/output.tar.gz"
      },
      "NumberOfTopics": 10,
      "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
    },
    {
```

```

    "JobId": "123456abcdeb0e11022f22a1EXAMPLE3",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:topics-detection-
job/123456abcdeb0e11022f22a1EXAMPLE3",
    "JobName": "topic-analysis-2",
    "JobStatus": "IN_PROGRESS",
    "SubmitTime": "2023-06-09T18:50:56.737000+00:00",
    "InputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET",
      "InputFormat": "ONE_DOC_PER_LINE"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/
thefolder/111122223333-TOPICS-123456abcdeb0e11022f22a1EXAMPLE3/output/output.tar.gz"
    },
    "NumberOfTopics": 10,
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
  }
]
}

```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [ListTopicsDetectionJobs](#) la Referencia de AWS CLI comandos.

## put-resource-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-resource-policy`.

### AWS CLI

Para adjuntar una política basada en recursos

En el siguiente `put-resource-policy` ejemplo, se adjunta una política basada en recursos a un modelo para que otra cuenta pueda importarla. AWS La política está asociada al modelo de la cuenta 111122223333 y permite a la cuenta 444455556666 importar el modelo.

```

aws comprehend put-resource-policy \
  --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1 \

```

```
--resource-policy '{"Version":"2012-10-17","Statement":
[{"Effect":"Allow","Action":"comprehend:ImportModel","Resource":"*","Principal":
{"AWS":["arn:aws:iam::444455556666:root"]}]}]}'
```

Salida:

```
{
  "PolicyRevisionId": "aaa111d069d07afaa2aa3106aEXAMPLE"
}
```

Para obtener más información, consulte [Copiar modelos personalizados entre AWS cuentas](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [PutResourcePolicy](#) la Referencia de AWS CLI comandos.

## start-document-classification-job

En el siguiente ejemplo de código, se muestra cómo usar `start-document-classification-job`.

### AWS CLI

Para empezar un trabajo de clasificación de documentos

El siguiente ejemplo de `start-document-classification-job` inicia un trabajo de clasificación de documentos con un modelo personalizado en todos los archivos de la dirección especificada en la etiqueta `--input-data-config`. En este ejemplo, el bucket de S3 de entrada contiene `SampleSMStext1.txt`, `SampleSMStext2.txt` y `SampleSMStext3.txt`. El modelo se entrenó previamente para clasificar los mensajes de spam y no spam o «ham». Cuando se completa el trabajo, `output.tar.gz` se coloca en la ubicación especificada por la etiqueta `--output-data-config`. `output.tar.gz` contiene `predictions.jsonl`, que enumera la clasificación de cada documento. La salida Json se imprime en una sola línea por archivo, pero se formatea aquí para que sea legible.

```
aws comprehend start-document-classification-job \
  --job-name exampleclassificationjob \
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET-INPUT/jobdata/" \
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
```

```
--document-classifier-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/mymodel/version/12
```

Contenidos de SampleSMStext1.txt:

```
"CONGRATULATIONS! TXT 2155550100 to win $5000"
```

Contenidos de SampleSMStext2.txt:

```
"Hi, when do you want me to pick you up from practice?"
```

Contenidos de SampleSMStext3.txt:

```
"Plz send bank account # to 2155550100 to claim prize!!"
```

Salida:

```
{
  "JobId": "e758dd56b824aa717ceab551fEXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:document-classification-job/e758dd56b824aa717ceab551fEXAMPLE",
  "JobStatus": "SUBMITTED"
}
```

Contenidos de predictions.jsonl:

```
{"File": "SampleSMStext1.txt", "Line": "0", "Classes": [{"Name": "spam", "Score": 0.9999}, {"Name": "ham", "Score": 0.0001}]}
{"File": "SampleSMStext2.txt", "Line": "0", "Classes": [{"Name": "ham", "Score": 0.9994}, {"Name": "spam", "Score": 0.0006}]}
{"File": "SampleSMStext3.txt", "Line": "0", "Classes": [{"Name": "spam", "Score": 0.9999}, {"Name": "ham", "Score": 0.0001}]}
```

Para obtener más información, consulte [Clasificación personalizada](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [StartDocumentClassificationJob](#) de AWS CLI comandos.



## start-dominant-language-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `start-dominant-language-detection-job`.

### AWS CLI

Para iniciar un trabajo de detección de idioma asíncrono

El siguiente `start-dominant-language-detection-job` ejemplo inicia un trabajo de detección de idioma asíncrono para todos los archivos ubicados en la dirección especificada por la etiqueta. `--input-data-config` El bucket S3 de este ejemplo contiene. `Sampletext1.txt` Cuando se completa el trabajo, la carpeta, `output`, se coloca en la ubicación especificada por la `--output-data-config` etiqueta. La carpeta `output.txt` contiene el idioma dominante de cada uno de los archivos de texto, así como la puntuación de confianza del modelo previamente entrenado para cada predicción.

```
aws comprehend start-dominant-language-detection-job \  
  --job-name example_language_analysis_job \  
  --language-code en \  
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/" \  
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \  
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role \  
  --language-code en
```

Contenido de `Sampletext1.txt`:

```
"Physics is the natural science that involves the study of matter and its motion and  
behavior through space and time, along with related concepts such as energy and  
force."
```

Salida:

```
{  
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:dominant-language-  
detection-job/123456abcdeb0e11022f22a11EXAMPLE",  
  "JobStatus": "SUBMITTED"  
}
```

Contenidos de `output.txt`:

```
{"File": "Sampletext1.txt", "Languages": [{"LanguageCode": "en", "Score": 0.9913753867149353}], "Line": 0}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [StartDominantLanguageDetectionJob](#) la Referencia de AWS CLI comandos.

## start-entities-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `start-entities-detection-job`.

### AWS CLI

Ejemplo 1: Para iniciar un trabajo de detección de entidades estándar utilizando el modelo previamente entrenado

El siguiente `start-entities-detection-job` ejemplo inicia un trabajo de detección de entidades asíncronas para todos los archivos ubicados en la dirección especificada por la etiqueta. `--input-data-config` El depósito S3 de este ejemplo contiene `Sampletext1.txt`, `Sampletext2.txt`, `Sampletext3.txt`. Cuando se completa el trabajo, la carpeta, `output`, se coloca en la ubicación especificada por la `--output-data-config` etiqueta. La carpeta `output.txt` contiene una lista de todas las entidades nombradas detectadas en cada archivo de texto, así como la puntuación de confianza del modelo previamente entrenado para cada predicción. La salida de Json se imprime en una línea por archivo de entrada, pero se formatea aquí para facilitar la lectura.

```
aws comprehend start-entities-detection-job \
  --job-name entitiestest \
  --language-code en \
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/" \
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-example-role \
  --language-code en
```

Contenidos de `Sampletext1.txt`:

```
"Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC credit card account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July 31st."
```

Contenidos de Sampletext2.txt:

```
"Dear Max, based on your autopay settings for your account example1.org account, we will withdraw your payment on the due date from your bank account number XXXXXX1111 with the routing number XXXXX0000. "
```

Contenidos de Sampletext3.txt:

```
"Jane, please submit any customer feedback from this weekend to AnySpa, 123 Main St, Anywhere and send comments to Alice at AnySpa@example.com."
```

Salida:

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-job/123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "SUBMITTED"
}
```

Contenido output.txt con sangrados de línea para facilitar la lectura:

```
{
  "Entities": [
    {
      "BeginOffset": 6,
      "EndOffset": 15,
      "Score": 0.9994006636420306,
      "Text": "Zhang Wei",
      "Type": "PERSON"
    },
    {
      "BeginOffset": 22,
      "EndOffset": 26,
      "Score": 0.9976647915128143,
      "Text": "John",
      "Type": "PERSON"
    }
  ]
}
```

```
  },
  {
    "BeginOffset": 33,
    "EndOffset": 67,
    "Score": 0.9984608700836206,
    "Text": "AnyCompany Financial Services, LLC",
    "Type": "ORGANIZATION"
  },
  {
    "BeginOffset": 88,
    "EndOffset": 107,
    "Score": 0.9868521019555556,
    "Text": "1111-XXXX-1111-XXXX",
    "Type": "OTHER"
  },
  {
    "BeginOffset": 133,
    "EndOffset": 139,
    "Score": 0.998242565709204,
    "Text": "$24.53",
    "Type": "QUANTITY"
  },
  {
    "BeginOffset": 155,
    "EndOffset": 164,
    "Score": 0.9993039263159287,
    "Text": "July 31st",
    "Type": "DATE"
  }
],
"File": "SampleText1.txt",
"Line": 0
}
{
  "Entities": [
    {
      "BeginOffset": 5,
      "EndOffset": 8,
      "Score": 0.9866232147545232,
      "Text": "Max",
      "Type": "PERSON"
    },
    {
      "BeginOffset": 156,
```

```
"EndOffset": 166,
"Score": 0.9797723450933329,
"Text": "XXXXXX1111",
"Type": "OTHER"
},
{
"BeginOffset": 191,
"EndOffset": 200,
"Score": 0.9247838572396843,
"Text": "XXXXXX0000",
"Type": "OTHER"
}
],
"File": "SampleText2.txt",
"Line": 0
}
{
"Entities": [
{
"Score": 0.9990532994270325,
"Type": "PERSON",
"Text": "Jane",
"BeginOffset": 0,
"EndOffset": 4
},
{
"Score": 0.9519651532173157,
"Type": "DATE",
"Text": "this weekend",
"BeginOffset": 47,
"EndOffset": 59
},
{
"Score": 0.5566426515579224,
"Type": "ORGANIZATION",
"Text": "AnySpa",
"BeginOffset": 63,
"EndOffset": 69
},
{
"Score": 0.8059805631637573,
"Type": "LOCATION",
"Text": "123 Main St, Anywhere",
"BeginOffset": 71,
```

```

    "EndOffset": 92
  },
  {
    "Score": 0.998830258846283,
    "Type": "PERSON",
    "Text": "Alice",
    "BeginOffset": 114,
    "EndOffset": 119
  },
  {
    "Score": 0.997818112373352,
    "Type": "OTHER",
    "Text": "AnySpa@example.com",
    "BeginOffset": 123,
    "EndOffset": 138
  }
],
"File": "SampleText3.txt",
"Line": 0
}

```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

Ejemplo 2: Para iniciar un trabajo de detección de entidades personalizado

El siguiente `start-entities-detection-job` ejemplo inicia un trabajo de detección de entidades personalizado asíncrono para todos los archivos ubicados en la dirección especificada por la etiqueta. `--input-data-config` En este ejemplo, el depósito S3 de este ejemplo contiene `SampleFeedback1.txt`, `SampleFeedback2.txt`, `SampleFeedback3.txt`. El modelo de reconocimiento de entidades se ha basado en los comentarios del servicio de atención al cliente para reconocer los nombres de los dispositivos. Cuando se completa el trabajo, la carpeta, `output`, se coloca en la ubicación especificada por la `--output-data-config` etiqueta. La carpeta contiene `output.txt` una lista de todas las entidades nombradas detectadas en cada archivo de texto, así como la puntuación de confianza del modelo previamente entrenado para cada predicción. La salida Json se imprime en una sola línea por archivo, pero se formatea aquí para que sea legible.

```

aws comprehend start-entities-detection-job \
  --job-name customentiestest \

```

```
--entity-recognizer-arn "arn:aws:comprehend:us-west-2:111122223333:entity-recognizer/entityrecognizer" \
--language-code en \
--input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/jobdata/" \
--output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \
--data-access-role-arn "arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-I0role"
```

Contenidos de SampleFeedback1.txt:

```
"I've been on the AnyPhone app have had issues for 24 hours when trying to pay bill. Cannot make payment. Sigh. | Oh man! Lets get that app up and running. DM me, and we can get to work!"
```

Contenidos de SampleFeedback2.txt:

```
"Hi, I have a discrepancy with my new bill. Could we get it sorted out? A rep added stuff I didnt sign up for when I did my AnyPhone 10 upgrade. | We can absolutely get this sorted!"
```

Contenidos de SampleFeedback3.txt:

```
"Is the by 1 get 1 free AnySmartPhone promo still going on? | Hi Christian! It ended yesterday, send us a DM if you have any questions and we can take a look at your options!"
```

Salida:

```
{
  "JobId": "019ea9edac758806850fa8a79ff83021",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-job/019ea9edac758806850fa8a79ff83021",
  "JobStatus": "SUBMITTED"
}
```

Contenido output.txt con sangrados de línea para facilitar la lectura:

```
{
  "Entities": [
    {
```

```
    "BeginOffset": 17,
    "EndOffset": 25,
    "Score": 0.9999728210205924,
    "Text": "AnyPhone",
    "Type": "DEVICE"
  }
],
"File": "SampleFeedback1.txt",
"Line": 0
}
{
"Entities": [
  {
    "BeginOffset": 123,
    "EndOffset": 133,
    "Score": 0.9999892116761524,
    "Text": "AnyPhone 10",
    "Type": "DEVICE"
  }
],
"File": "SampleFeedback2.txt",
"Line": 0
}
{
"Entities": [
  {
    "BeginOffset": 23,
    "EndOffset": 35,
    "Score": 0.9999971389852362,
    "Text": "AnySmartPhone",
    "Type": "DEVICE"
  }
],
"File": "SampleFeedback3.txt",
"Line": 0
}
```

Para obtener más información, consulte [Reconocimiento de entidades personalizado](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [StartEntitiesDetectionJob](#) la Referencia de AWS CLI comandos.



## start-events-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `start-events-detection-job`.

### AWS CLI

Para iniciar un trabajo de detección de eventos asíncronos

El siguiente `start-events-detection-job` ejemplo inicia un trabajo de detección de eventos asíncronos para todos los archivos ubicados en la dirección especificada por la etiqueta. `--input-data-config` Los posibles tipos de eventos objetivo incluyen `BANKRUPTCYEMPLOYMENT,CORPORATE_ACQUISITION,INVESTMENT_GENERAL,CORPORATE_MERGER,,RIGHTS_ISSUE SECONDARY_OFFERINGSHELF_OFFERING, TENDER_OFFERING` y `STOCK_SPLIT` El depósito S3 de este ejemplo contiene `SampleText1.txt,SampleText2.txt, ySampleText3.txt`. Cuando se completa el trabajo, la carpeta, `output`, se coloca en la ubicación especificada por la `--output-data-config` etiqueta. La carpeta contiene `SampleText1.txt.outSampleText2.txt.out, ySampleText3.txt.out`. El JSON resultado se imprime en una línea por archivo, pero se formatea aquí para facilitar la lectura.

```
aws comprehend start-events-detection-job \
  --job-name events-detection-1 \
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/EventsData" \
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-servicerole \
  --language-code en \
  --target-event-
types "BANKRUPTCY" "EMPLOYMENT" "CORPORATE_ACQUISITION" "CORPORATE_MERGER" "INVESTMENT_GENER
```

Contenidos de `SampleText1.txt`:

```
"Company AnyCompany grew by increasing sales and through acquisitions. After
purchasing competing firms in 2020, AnyBusiness, a part of the AnyBusinessGroup,
gave Jane Does firm a going rate of one cent a gallon or forty-two cents a barrel."
```

Contenidos de `SampleText2.txt`:

```
"In 2021, AnyCompany officially purchased AnyBusiness for 100 billion dollars,
surprising and exciting the shareholders."
```

Contenidos de `SampleText3.txt`:

```
"In 2022, AnyCompany stock crashed 50. Eventually later that year they filed for bankruptcy."
```

Salida:

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:events-detection-job/123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "SUBMITTED"
}
```

Contenido de líneas SampleText1.txt.out con sangrados para facilitar la lectura:

```
{
  "Entities": [
    {
      "Mentions": [
        {
          "BeginOffset": 8,
          "EndOffset": 18,
          "Score": 0.99977,
          "Text": "AnyCompany",
          "Type": "ORGANIZATION",
          "GroupScore": 1
        },
        {
          "BeginOffset": 112,
          "EndOffset": 123,
          "Score": 0.999747,
          "Text": "AnyBusiness",
          "Type": "ORGANIZATION",
          "GroupScore": 0.979826
        },
        {
          "BeginOffset": 171,
          "EndOffset": 175,
          "Score": 0.999615,
          "Text": "firm",
          "Type": "ORGANIZATION",
          "GroupScore": 0.871647
        }
      ]
    }
  ]
}
```

```
]
},
{
  "Mentions": [
    {
      "BeginOffset": 97,
      "EndOffset": 102,
      "Score": 0.987687,
      "Text": "firms",
      "Type": "ORGANIZATION",
      "GroupScore": 1
    }
  ]
},
{
  "Mentions": [
    {
      "BeginOffset": 103,
      "EndOffset": 110,
      "Score": 0.999458,
      "Text": "in 2020",
      "Type": "DATE",
      "GroupScore": 1
    }
  ]
},
{
  "Mentions": [
    {
      "BeginOffset": 160,
      "EndOffset": 168,
      "Score": 0.999649,
      "Text": "John Doe",
      "Type": "PERSON",
      "GroupScore": 1
    }
  ]
}
],
"Events": [
  {
    "Type": "CORPORATE_ACQUISITION",
    "Arguments": [
      {
```

```
    "EntityIndex": 0,  
    "Role": "INVESTOR",  
    "Score": 0.99977  
  }  
],  
"Triggers": [  
  {  
    "BeginOffset": 56,  
    "EndOffset": 68,  
    "Score": 0.999967,  
    "Text": "acquisitions",  
    "Type": "CORPORATE_ACQUISITION",  
    "GroupScore": 1  
  }  
]  
},  
{  
  "Type": "CORPORATE_ACQUISITION",  
  "Arguments": [  
    {  
      "EntityIndex": 1,  
      "Role": "INVESTEES",  
      "Score": 0.987687  
    },  
    {  
      "EntityIndex": 2,  
      "Role": "DATE",  
      "Score": 0.999458  
    },  
    {  
      "EntityIndex": 3,  
      "Role": "INVESTOR",  
      "Score": 0.999649  
    }  
  ],  
  "Triggers": [  
    {  
      "BeginOffset": 76,  
      "EndOffset": 86,  
      "Score": 0.999973,  
      "Text": "purchasing",  
      "Type": "CORPORATE_ACQUISITION",  
      "GroupScore": 1  
    }  
  ]  
}
```

```
    ]
  }
],
"File": "SampleText1.txt",
"Line": 0
}
```

### Contenidos de SampleText2.txt.out:

```
{
  "Entities": [
    {
      "Mentions": [
        {
          "BeginOffset": 0,
          "EndOffset": 7,
          "Score": 0.999473,
          "Text": "In 2021",
          "Type": "DATE",
          "GroupScore": 1
        }
      ]
    },
    {
      "Mentions": [
        {
          "BeginOffset": 9,
          "EndOffset": 19,
          "Score": 0.999636,
          "Text": "AnyCompany",
          "Type": "ORGANIZATION",
          "GroupScore": 1
        }
      ]
    },
    {
      "Mentions": [
        {
          "BeginOffset": 45,
          "EndOffset": 56,
          "Score": 0.999712,
          "Text": "AnyBusiness",
          "Type": "ORGANIZATION",

```

```
        "GroupScore": 1
      }
    ]
  },
  {
    "Mentions": [
      {
        "BeginOffset": 61,
        "EndOffset": 80,
        "Score": 0.998886,
        "Text": "100 billion dollars",
        "Type": "MONETARY_VALUE",
        "GroupScore": 1
      }
    ]
  }
],
"Events": [
  {
    "Type": "CORPORATE_ACQUISITION",
    "Arguments": [
      {
        "EntityIndex": 3,
        "Role": "AMOUNT",
        "Score": 0.998886
      },
      {
        "EntityIndex": 2,
        "Role": "INVESTEES",
        "Score": 0.999712
      },
      {
        "EntityIndex": 0,
        "Role": "DATE",
        "Score": 0.999473
      },
      {
        "EntityIndex": 1,
        "Role": "INVESTOR",
        "Score": 0.999636
      }
    ]
  },
  {
    "Triggers": [
      {
```

```
        "BeginOffset": 31,
        "EndOffset": 40,
        "Score": 0.99995,
        "Text": "purchased",
        "Type": "CORPORATE_ACQUISITION",
        "GroupScore": 1
      }
    ]
  }
],
"File": "SampleText2.txt",
"Line": 0
}
```

Contenidos de SampleText3.txt.out:

```
{
  "Entities": [
    {
      "Mentions": [
        {
          "BeginOffset": 9,
          "EndOffset": 19,
          "Score": 0.999774,
          "Text": "AnyCompany",
          "Type": "ORGANIZATION",
          "GroupScore": 1
        },
        {
          "BeginOffset": 66,
          "EndOffset": 70,
          "Score": 0.995717,
          "Text": "they",
          "Type": "ORGANIZATION",
          "GroupScore": 0.997626
        }
      ]
    },
    {
      "Mentions": [
        {
          "BeginOffset": 50,
          "EndOffset": 65,
```

```
        "Score": 0.999656,
        "Text": "later that year",
        "Type": "DATE",
        "GroupScore": 1
      }
    ]
  }
],
"Events": [
  {
    "Type": "BANKRUPTCY",
    "Arguments": [
      {
        "EntityIndex": 1,
        "Role": "DATE",
        "Score": 0.999656
      },
      {
        "EntityIndex": 0,
        "Role": "FILER",
        "Score": 0.995717
      }
    ],
    "Triggers": [
      {
        "BeginOffset": 81,
        "EndOffset": 91,
        "Score": 0.999936,
        "Text": "bankruptcy",
        "Type": "BANKRUPTCY",
        "GroupScore": 1
      }
    ]
  }
],
"File": "SampleText3.txt",
"Line": 0
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [StartEventsDetectionJob](#) de AWS CLI comandos.



## start-flywheel-iteration

En el siguiente ejemplo de código, se muestra cómo usar `start-flywheel-iteration`.

### AWS CLI

Para iniciar una iteración del volante

En el siguiente `start-flywheel-iteration` ejemplo se inicia una iteración del volante. Esta operación utiliza todos los conjuntos de datos nuevos del volante para entrenar una nueva versión del modelo.

```
aws comprehend start-flywheel-iteration \  
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-  
flywheel
```

Salida:

```
{  
  "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-  
flywheel",  
  "FlywheelIterationId": "12345123EXAMPLE"  
}
```

Para obtener más información, consulte la [descripción general de Flywheel](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [StartFlywheelIteration](#) de AWS CLI comandos.

## start-key-phrases-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `start-key-phrases-detection-job`.

### AWS CLI

Para iniciar un trabajo de detección de frases clave

El siguiente `start-key-phrases-detection-job` ejemplo inicia un trabajo de detección asíncrona de frases clave para todos los archivos ubicados en la dirección especificada por la etiqueta. `--input-data-config` El bucket S3 de este ejemplo contiene `Sampletext1.txt`,

ysampletext2.txt. Sampletext3.txt Cuando se completa el trabajo, la carpeta, output, se coloca en la ubicación especificada por la `--output-data-config` etiqueta. La carpeta contiene el archivo `output.txt` que contiene todas las frases clave detectadas en cada archivo de texto y la puntuación de confianza del modelo previamente entrenado para cada predicción. La salida Json se imprime en una sola línea por archivo, pero se formatea aquí para que sea legible.

```
aws comprehend start-key-phrases-detection-job \  
  --job-name keyphrasesanalysisitest1 \  
  --language-code en \  
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/" \  
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \  
  --data-access-role-arn "arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role" \  
  --language-code en
```

Contenidos de `Sampletext1.txt`:

```
"Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC credit card  
account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July  
31st."
```

Contenidos de `Sampletext2.txt`:

```
"Dear Max, based on your autopay settings for your account Internet.org account, we  
will withdraw your payment on the due date from your bank account number XXXXXX1111  
with the routing number XXXXX0000. "
```

Contenidos de `Sampletext3.txt`:

```
"Jane, please submit any customer feedback from this weekend to Sunshine Spa, 123  
Main St, Anywhere and send comments to Alice at AnySpa@example.com."
```

Salida:

```
{  
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-detection-  
job/123456abcdeb0e11022f22a11EXAMPLE",  
  "JobStatus": "SUBMITTED"  
}
```

## Contenido de output .txt con sangrados de línea para mayor legibilidad:

```
{
  "File": "SampleText1.txt",
  "KeyPhrases": [
    {
      "BeginOffset": 6,
      "EndOffset": 15,
      "Score": 0.9748965572679326,
      "Text": "Zhang Wei"
    },
    {
      "BeginOffset": 22,
      "EndOffset": 26,
      "Score": 0.9997344722354619,
      "Text": "John"
    },
    {
      "BeginOffset": 28,
      "EndOffset": 62,
      "Score": 0.9843791074032948,
      "Text": "Your AnyCompany Financial Services"
    },
    {
      "BeginOffset": 64,
      "EndOffset": 107,
      "Score": 0.8976122401721824,
      "Text": "LLC credit card account 1111-XXXX-1111-XXXX"
    },
    {
      "BeginOffset": 112,
      "EndOffset": 129,
      "Score": 0.9999612982629748,
      "Text": "a minimum payment"
    },
    {
      "BeginOffset": 133,
      "EndOffset": 139,
      "Score": 0.99975728947036,
      "Text": "$24.53"
    },
    {
      "BeginOffset": 155,
      "EndOffset": 164,
```

```
    "Score": 0.9940866241449973,
    "Text": "July 31st"
  }
],
"Line": 0
}
{
"File": "SampleText2.txt",
"KeyPhrases": [
  {
    "BeginOffset": 0,
    "EndOffset": 8,
    "Score": 0.9974021100118472,
    "Text": "Dear Max"
  },
  {
    "BeginOffset": 19,
    "EndOffset": 40,
    "Score": 0.9961120519515884,
    "Text": "your autopay settings"
  },
  {
    "BeginOffset": 45,
    "EndOffset": 78,
    "Score": 0.9980620070116009,
    "Text": "your account Internet.org account"
  },
  {
    "BeginOffset": 97,
    "EndOffset": 109,
    "Score": 0.999919660140754,
    "Text": "your payment"
  },
  {
    "BeginOffset": 113,
    "EndOffset": 125,
    "Score": 0.9998370719754205,
    "Text": "the due date"
  },
  {
    "BeginOffset": 131,
    "EndOffset": 166,
    "Score": 0.9955068678502509,
    "Text": "your bank account number XXXXXX1111"
```

```
    },
    {
      "BeginOffset": 172,
      "EndOffset": 200,
      "Score": 0.8653433315829526,
      "Text": "the routing number XXXXX0000"
    }
  ],
  "Line": 0
}
{
  "File": "SampleText3.txt",
  "KeyPhrases": [
    {
      "BeginOffset": 0,
      "EndOffset": 4,
      "Score": 0.9142947833681668,
      "Text": "Jane"
    },
    {
      "BeginOffset": 20,
      "EndOffset": 41,
      "Score": 0.9984325676596763,
      "Text": "any customer feedback"
    },
    {
      "BeginOffset": 47,
      "EndOffset": 59,
      "Score": 0.9998782448150636,
      "Text": "this weekend"
    },
    {
      "BeginOffset": 63,
      "EndOffset": 75,
      "Score": 0.99866741830757,
      "Text": "Sunshine Spa"
    },
    {
      "BeginOffset": 77,
      "EndOffset": 88,
      "Score": 0.9695803485466054,
      "Text": "123 Main St"
    }
  ]
}
```

```
    "BeginOffset": 108,  
    "EndOffset": 116,  
    "Score": 0.9997065928550928,  
    "Text": "comments"  
  },  
  {  
    "BeginOffset": 120,  
    "EndOffset": 125,  
    "Score": 0.9993466833825161,  
    "Text": "Alice"  
  },  
  {  
    "BeginOffset": 129,  
    "EndOffset": 144,  
    "Score": 0.9654563612885667,  
    "Text": "AnySpa@example.com"  
  }  
],  
"Line": 0  
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia de [StartKeyPhrasesDetectionJob](#) comandos AWS CLI .

## start-pii-entities-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `start-pii-entities-detection-job`.

### AWS CLI

Para iniciar un trabajo de detección asíncrona PII

El siguiente `start-pii-entities-detection-job` ejemplo inicia un trabajo asincrónico de detección de entidades con información de identificación personal (PII) para todos los archivos ubicados en la dirección especificada por la etiqueta. `--input-data-config` El depósito S3 de este ejemplo contiene `Sampletext1.txt`, `Sampletext2.txt`, `Sampletext3.txt`. Cuando se completa el trabajo, la carpeta, `output`, se coloca en la ubicación especificada por la `--output-data-config` etiqueta. La carpeta contiene `SampleText1.txt.out`, `SampleText2.txt.out`,

y `SampleText3.txt.out` contiene una lista de las entidades nombradas dentro de cada archivo de texto. La salida Json se imprime en una sola línea por archivo, pero se formatea aquí para que sea legible.

```
aws comprehend start-pii-entities-detection-job \
  --job-name entities_test \
  --language-code en \
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/" \
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role \
  --language-code en \
  --mode ONLY_OFFSETS
```

Contenidos de `Sampletext1.txt`:

```
"Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC credit card
account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July
31st."
```

Contenidos de `Sampletext2.txt`:

```
"Dear Max, based on your autopay settings for your account Internet.org account, we
will withdraw your payment on the due date from your bank account number XXXXXX1111
with the routing number XXXXX0000. "
```

Contenidos de `Sampletext3.txt`:

```
"Jane, please submit any customer feedback from this weekend to Sunshine Spa, 123
Main St, Anywhere and send comments to Alice at AnySpa@example.com."
```

Salida:

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:pii-entities-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "SUBMITTED"
}
```

Contenido `SampleText1.txt.out` con sangrados de línea para facilitar la lectura:

```
{
  "Entities": [
    {
      "BeginOffset": 6,
      "EndOffset": 15,
      "Type": "NAME",
      "Score": 0.9998490510222595
    },
    {
      "BeginOffset": 22,
      "EndOffset": 26,
      "Type": "NAME",
      "Score": 0.9998937958019426
    },
    {
      "BeginOffset": 88,
      "EndOffset": 107,
      "Type": "CREDIT_DEBIT_NUMBER",
      "Score": 0.9554297245278491
    },
    {
      "BeginOffset": 155,
      "EndOffset": 164,
      "Type": "DATE_TIME",
      "Score": 0.9999720462925257
    }
  ],
  "File": "SampleText1.txt",
  "Line": 0
}
```

Contenido de sangrados SampleText2.txt.out con líneas para facilitar la lectura:

```
{
  "Entities": [
    {
      "BeginOffset": 5,
      "EndOffset": 8,
      "Type": "NAME",
      "Score": 0.9994390774924007
    },
    {
      "BeginOffset": 58,
```



```
    "EndOffset": 70,
    "Type": "URL",
    "Score": 0.9999958276922101
  },
  {
    "BeginOffset": 156,
    "EndOffset": 166,
    "Type": "BANK_ACCOUNT_NUMBER",
    "Score": 0.9999721058045592
  },
  {
    "BeginOffset": 191,
    "EndOffset": 200,
    "Type": "BANK_ROUTING",
    "Score": 0.9998968945989909
  }
],
"File": "SampleText2.txt",
"Line": 0
}
```

Contenido de sangrados SampleText3.txt.out con líneas para facilitar la lectura:

```
{
  "Entities": [
    {
      "BeginOffset": 0,
      "EndOffset": 4,
      "Type": "NAME",
      "Score": 0.999949934606805
    },
    {
      "BeginOffset": 77,
      "EndOffset": 88,
      "Type": "ADDRESS",
      "Score": 0.9999035300466904
    },
    {
      "BeginOffset": 120,
      "EndOffset": 125,
      "Type": "NAME",
      "Score": 0.9998203838716296
    }
  ],
}
```

```

    {
      "BeginOffset": 129,
      "EndOffset": 144,
      "Type": "EMAIL",
      "Score": 0.9998313473105228
    }
  ],
  "File": "SampleText3.txt",
  "Line": 0
}

```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [StartPiiEntitiesDetectionJob](#) de AWS CLI comandos.

## start-sentiment-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `start-sentiment-detection-job`.

### AWS CLI

Para iniciar un trabajo de análisis de sentimientos asíncrono

El siguiente `start-sentiment-detection-job` ejemplo inicia un trabajo de detección asíncrono de análisis de sentimientos para todos los archivos ubicados en la dirección especificada por la etiqueta. `--input-data-config` La carpeta de bucket S3 de este ejemplo contiene `SampleMovieReview1.txt`, `SampleMovieReview2.txt`, `SampleMovieReview3.txt`. Cuando se completa el trabajo, la carpeta, `output`, se coloca en la ubicación especificada por la `--output-data-config` etiqueta. La carpeta contiene el archivo `output.txt`, que contiene las opiniones predominantes para cada archivo de texto y la puntuación de confianza del modelo previamente entrenado para cada predicción. La salida Json se imprime en una sola línea por archivo, pero se formatea aquí para que sea legible.

```

aws comprehend start-sentiment-detection-job \
  --job-name example-sentiment-detection-job \
  --language-code en \
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/MovieData" \
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \

```

```
--data-access-role-arn arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-example-role
```

Contenidos de SampleMovieReview1.txt:

```
"The film, AnyMovie2, is fairly predictable and just okay."
```

Contenidos de SampleMovieReview2.txt:

```
"AnyMovie2 is the essential sci-fi film that I grew up watching when I was a kid. I highly recommend this movie."
```

Contenidos de SampleMovieReview3.txt:

```
"Don't get fooled by the 'awards' for AnyMovie2. All parts of the film were poorly stolen from other modern directors."
```

Salida:

```
{
  "JobId": "0b5001e25f62ebb40631a9a1a7fde7b3",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:sentiment-detection-job/0b5001e25f62ebb40631a9a1a7fde7b3",
  "JobStatus": "SUBMITTED"
}
```

Contenido output.txt con una línea de guiones para facilitar la lectura:

```
{
  "File": "SampleMovieReview1.txt",
  "Line": 0,
  "Sentiment": "MIXED",
  "SentimentScore": {
    "Mixed": 0.6591159105300903,
    "Negative": 0.26492202281951904,
    "Neutral": 0.035430654883384705,
    "Positive": 0.04053137078881264
  }
}
{
  "File": "SampleMovieReview2.txt",
```

```
    "Line": 0,
    "Sentiment": "POSITIVE",
    "SentimentScore": {
      "Mixed": 0.000008718466233403888,
      "Negative": 0.00006134175055194646,
      "Neutral": 0.0002941041602753103,
      "Positive": 0.9996358156204224
    }
  }
{
  "File": "SampleMovieReview3.txt",
  "Line": 0,
  "Sentiment": "NEGATIVE",
  "SentimentScore": {
    "Mixed": 0.004146667663007975,
    "Negative": 0.9645107984542847,
    "Neutral": 0.016559595242142677,
    "Positive": 0.014782938174903393
  }
}
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [StartSentimentDetectionJob](#) de AWS CLI comandos.

## start-targeted-sentiment-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `start-targeted-sentiment-detection-job`.

### AWS CLI

Para iniciar un trabajo de análisis de sentimientos segmentado y asíncrono

En el siguiente `start-targeted-sentiment-detection-job` ejemplo, se inicia un trabajo de detección asíncrono de análisis de opiniones segmentado para todos los archivos ubicados en la dirección especificada por la etiqueta. `--input-data-config` La carpeta de bucket S3 de este ejemplo contiene `SampleMovieReview1.txt`, `ySampleMovieReview2.txt`. `SampleMovieReview3.txt` Cuando se completa el

trabajo, `output.tar.gz` se coloca en la ubicación especificada por la `--output-data-config` etiqueta. `output.tar.gz` contiene los archivos y `SampleMovieReview1.txt.out`, `SampleMovieReview2.txt.out`, `SampleMovieReview3.txt.out`, cada uno de los cuales contiene todas las entidades nombradas y las opiniones asociadas para un único archivo de texto de entrada.

```
aws comprehend start-targeted-sentiment-detection-job \  
  --job-name targeted_movie_review_analysis1 \  
  --language-code en \  
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/MovieData" \  
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \  
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/  
AmazonComprehendServiceRole-example-role
```

Contenidos de `SampleMovieReview1.txt`:

```
"The film, AnyMovie, is fairly predictable and just okay."
```

Contenidos de `SampleMovieReview2.txt`:

```
"AnyMovie is the essential sci-fi film that I grew up watching when I was a kid. I  
highly recommend this movie."
```

Contenidos de `SampleMovieReview3.txt`:

```
"Don't get fooled by the 'awards' for AnyMovie. All parts of the film were poorly  
stolen from other modern directors."
```

Salida:

```
{  
  "JobId": "0b5001e25f62ebb40631a9a1a7fde7b3",  
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:targeted-sentiment-  
detection-job/0b5001e25f62ebb40631a9a1a7fde7b3",  
  "JobStatus": "SUBMITTED"  
}
```

Contenido `SampleMovieReview1.txt.out` con sangrados de línea para facilitar la lectura:

```
{
```

```
"Entities": [
  {
    "DescriptiveMentionIndex": [
      0
    ],
    "Mentions": [
      {
        "BeginOffset": 4,
        "EndOffset": 8,
        "Score": 0.994972,
        "GroupScore": 1,
        "Text": "film",
        "Type": "MOVIE",
        "MentionSentiment": {
          "Sentiment": "NEUTRAL",
          "SentimentScore": {
            "Mixed": 0,
            "Negative": 0,
            "Neutral": 1,
            "Positive": 0
          }
        }
      }
    ]
  },
  {
    "DescriptiveMentionIndex": [
      0
    ],
    "Mentions": [
      {
        "BeginOffset": 10,
        "EndOffset": 18,
        "Score": 0.631368,
        "GroupScore": 1,
        "Text": "AnyMovie",
        "Type": "ORGANIZATION",
        "MentionSentiment": {
          "Sentiment": "POSITIVE",
          "SentimentScore": {
            "Mixed": 0.001729,
            "Negative": 0.000001,
            "Neutral": 0.000318,
            "Positive": 0.997952
          }
        }
      }
    ]
  }
]
```

```
        }
      }
    }
  ],
  "File": "SampleMovieReview1.txt",
  "Line": 0
}
```

Contenido de los guiones de las SampleMovieReview2.txt.out líneas para facilitar la lectura:

```
{
  "Entities": [
    {
      "DescriptiveMentionIndex": [
        0
      ],
      "Mentions": [
        {
          "BeginOffset": 0,
          "EndOffset": 8,
          "Score": 0.854024,
          "GroupScore": 1,
          "Text": "AnyMovie",
          "Type": "MOVIE",
          "MentionSentiment": {
            "Sentiment": "POSITIVE",
            "SentimentScore": {
              "Mixed": 0,
              "Negative": 0,
              "Neutral": 0.000007,
              "Positive": 0.999993
            }
          }
        }
      ],
      {
        "BeginOffset": 104,
        "EndOffset": 109,
        "Score": 0.999129,
        "GroupScore": 0.502937,
        "Text": "movie",
        "Type": "MOVIE",

```

```
    "MentionSentiment": {
      "Sentiment": "POSITIVE",
      "SentimentScore": {
        "Mixed": 0,
        "Negative": 0,
        "Neutral": 0,
        "Positive": 1
      }
    }
  },
  {
    "BeginOffset": 33,
    "EndOffset": 37,
    "Score": 0.999823,
    "GroupScore": 0.999252,
    "Text": "film",
    "Type": "MOVIE",
    "MentionSentiment": {
      "Sentiment": "POSITIVE",
      "SentimentScore": {
        "Mixed": 0,
        "Negative": 0,
        "Neutral": 0.000001,
        "Positive": 0.999999
      }
    }
  }
],
{
  "DescriptiveMentionIndex": [
    0,
    1,
    2
  ],
  "Mentions": [
    {
      "BeginOffset": 43,
      "EndOffset": 44,
      "Score": 0.999997,
      "GroupScore": 1,
      "Text": "I",
      "Type": "PERSON",
      "MentionSentiment": {
```



```
        "Sentiment": "NEUTRAL",
        "SentimentScore": {
            "Mixed": 0,
            "Negative": 0,
            "Neutral": 1,
            "Positive": 0
        }
    },
    {
        "BeginOffset": 80,
        "EndOffset": 81,
        "Score": 0.999996,
        "GroupScore": 0.52523,
        "Text": "I",
        "Type": "PERSON",
        "MentionSentiment": {
            "Sentiment": "NEUTRAL",
            "SentimentScore": {
                "Mixed": 0,
                "Negative": 0,
                "Neutral": 1,
                "Positive": 0
            }
        }
    },
    {
        "BeginOffset": 67,
        "EndOffset": 68,
        "Score": 0.999994,
        "GroupScore": 0.999499,
        "Text": "I",
        "Type": "PERSON",
        "MentionSentiment": {
            "Sentiment": "NEUTRAL",
            "SentimentScore": {
                "Mixed": 0,
                "Negative": 0,
                "Neutral": 1,
                "Positive": 0
            }
        }
    }
]
```

```

    },
    {
      "DescriptiveMentionIndex": [
        0
      ],
      "Mentions": [
        {
          "BeginOffset": 75,
          "EndOffset": 78,
          "Score": 0.999978,
          "GroupScore": 1,
          "Text": "kid",
          "Type": "PERSON",
          "MentionSentiment": {
            "Sentiment": "NEUTRAL",
            "SentimentScore": {
              "Mixed": 0,
              "Negative": 0,
              "Neutral": 1,
              "Positive": 0
            }
          }
        }
      ]
    }
  ],
  "File": "SampleMovieReview2.txt",
  "Line": 0
}

```

Contenido de los guiones `SampleMovieReview3.txt.out` con líneas para facilitar la legibilidad:

```

{
  "Entities": [
    {
      "DescriptiveMentionIndex": [
        1
      ],
      "Mentions": [
        {
          "BeginOffset": 64,
          "EndOffset": 68,

```

```
    "Score": 0.992953,
    "GroupScore": 0.999814,
    "Text": "film",
    "Type": "MOVIE",
    "MentionSentiment": {
      "Sentiment": "NEUTRAL",
      "SentimentScore": {
        "Mixed": 0.000004,
        "Negative": 0.010425,
        "Neutral": 0.989543,
        "Positive": 0.000027
      }
    }
  },
  {
    "BeginOffset": 37,
    "EndOffset": 45,
    "Score": 0.999782,
    "GroupScore": 1,
    "Text": "AnyMovie",
    "Type": "ORGANIZATION",
    "MentionSentiment": {
      "Sentiment": "POSITIVE",
      "SentimentScore": {
        "Mixed": 0.000095,
        "Negative": 0.039847,
        "Neutral": 0.000673,
        "Positive": 0.959384
      }
    }
  }
],
{
  "DescriptiveMentionIndex": [
    0
  ],
  "Mentions": [
    {
      "BeginOffset": 47,
      "EndOffset": 50,
      "Score": 0.999991,
      "GroupScore": 1,
      "Text": "All",
```

```
    "Type": "QUANTITY",
    "MentionSentiment": {
      "Sentiment": "NEUTRAL",
      "SentimentScore": {
        "Mixed": 0.000001,
        "Negative": 0.000001,
        "Neutral": 0.999998,
        "Positive": 0
      }
    }
  }
],
},
{
  "DescriptiveMentionIndex": [
    0
  ],
  "Mentions": [
    {
      "BeginOffset": 106,
      "EndOffset": 115,
      "Score": 0.542083,
      "GroupScore": 1,
      "Text": "directors",
      "Type": "PERSON",
      "MentionSentiment": {
        "Sentiment": "NEUTRAL",
        "SentimentScore": {
          "Mixed": 0,
          "Negative": 0,
          "Neutral": 1,
          "Positive": 0
        }
      }
    }
  ]
}
],
"File": "SampleMovieReview3.txt",
"Line": 0
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia de [StartTargetedSentimentDetectionJob](#) comandos AWS CLI .

## start-topics-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `start-topics-detection-job`.

### AWS CLI

Para iniciar un trabajo de análisis de detección de temas

El siguiente ejemplo de `start-topics-detection-job` inicia un trabajo de detección de temas asíncrono para todos los archivos ubicados en la dirección especificada por la etiqueta `--input-data-config`. Una vez finalizado el trabajo, la carpeta, `output`, se coloca en la ubicación especificada por la etiqueta `--output-data-config`. `output` contiene `topic-terms.csv` y `doc-topics.csv`. El primer archivo de salida, `topic-terms.csv`, es una lista de temas del conjunto. Para cada tema, la lista incluye, de forma predeterminada, los principales términos por tema según su importancia. El segundo archivo, `doc-topics.csv`, enumera los documentos relacionados con un tema y la proporción del documento que trata sobre el tema.

```
aws comprehend start-topics-detection-job \
  --job-name example_topics_detection_job \
  --language-code en \
  --input-data-config "S3Uri=s3://DOC-EXAMPLE-BUCKET/" \
  --output-data-config "S3Uri=s3://DOC-EXAMPLE-DESTINATION-BUCKET/testfolder/" \
  --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
  --language-code en
```

Salida:

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "SUBMITTED"
}
```

Para obtener más información, consulte [Modelado de temas](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [StartTopicsDetectionJob](#) la Referencia de AWS CLI comandos.

## stop-dominant-language-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `stop-dominant-language-detection-job`.

### AWS CLI

Para detener un trabajo de detección asíncrona de lenguaje dominante

El siguiente `stop-dominant-language-detection-job` ejemplo detiene un trabajo de detección de idioma dominante asíncrono en curso. Si el estado actual del trabajo es, `IN_PROGRESS` el trabajo se marca para su finalización y se coloca en ese estado. `STOP_REQUESTED` Si el trabajo se completa antes de poder detenerlo, se coloca en ese `COMPLETED` estado.

```
aws comprehend stop-dominant-language-detection-job \
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "STOP_REQUESTED"
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [StopDominantLanguageDetectionJob](#) la Referencia de AWS CLI comandos.

## stop-entities-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `stop-entities-detection-job`.

## AWS CLI

Para detener un trabajo de detección de entidades asíncronas

El siguiente `stop-entities-detection-job` ejemplo detiene un trabajo de detección de entidades asincrónico en curso. Si el estado actual del trabajo es, `IN_PROGRESS` el trabajo se marca para su finalización y se coloca en ese estado. `STOP_REQUESTED` Si el trabajo se completa antes de poder detenerlo, se coloca en ese `COMPLETED` estado.

```
aws comprehend stop-entities-detection-job \  
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{  
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
  "JobStatus": "STOP_REQUESTED"  
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [StopEntitiesDetectionJob](#) la Referencia de AWS CLI comandos.

## stop-events-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `stop-events-detection-job`.

### AWS CLI

Para detener un trabajo de detección de eventos asíncronos

El siguiente `stop-events-detection-job` ejemplo detiene un trabajo de detección de eventos asincrónico en curso. Si el estado actual del trabajo es, `IN_PROGRESS` el trabajo se marca para su finalización y se coloca en ese estado. `STOP_REQUESTED` Si el trabajo se completa antes de poder detenerlo, se coloca en ese `COMPLETED` estado.

```
aws comprehend stop-events-detection-job \  
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "STOP_REQUESTED"
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [StopEventsDetectionJob](#) la Referencia de AWS CLI comandos.

### stop-key-phrases-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `stop-key-phrases-detection-job`.

AWS CLI

Para detener un trabajo de detección asíncrona de frases clave

El siguiente `stop-key-phrases-detection-job` ejemplo detiene un trabajo de detección de frases clave asíncrono en curso. Si el estado actual del trabajo es, `IN_PROGRESS` el trabajo se marca para su finalización y se coloca en ese estado. `STOP_REQUESTED` Si el trabajo se completa antes de poder detenerlo, se coloca en ese `COMPLETED` estado.

```
aws comprehend stop-key-phrases-detection-job \
  --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
  "JobStatus": "STOP_REQUESTED"
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [StopKeyPhrasesDetectionJob](#) la Referencia de AWS CLI comandos.



## stop-pii-entities-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `stop-pii-entities-detection-job`.

### AWS CLI

Para detener un trabajo asíncrono de detección de entidades pii

El siguiente `stop-pii-entities-detection-job` ejemplo detiene un trabajo de detección de entidades pii asincrónico en curso. Si el estado actual del trabajo es, `IN_PROGRESS` el trabajo se marca para su finalización y se coloca en ese estado. `STOP_REQUESTED` Si el trabajo se completa antes de poder detenerlo, se coloca en ese `COMPLETED` estado.

```
aws comprehend stop-pii-entities-detection-job \  
--job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{  
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
  "JobStatus": "STOP_REQUESTED"  
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [StopPiiEntitiesDetectionJob](#) la Referencia de AWS CLI comandos.

## stop-sentiment-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `stop-sentiment-detection-job`.

### AWS CLI

Para detener un trabajo asíncrono de detección de sentimientos

El siguiente `stop-sentiment-detection-job` ejemplo detiene un trabajo de detección de sentimientos asincrónico en curso. Si el estado actual del trabajo es, `IN_PROGRESS` el trabajo se marca para su finalización y se coloca en ese estado. `STOP_REQUESTED` Si el trabajo se completa antes de poder detenerlo, se coloca en ese `COMPLETED` estado.

```
aws comprehend stop-sentiment-detection-job \  
--job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{  
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
  "JobStatus": "STOP_REQUESTED"  
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [StopSentimentDetectionJob](#) la Referencia de AWS CLI comandos.

## stop-targeted-sentiment-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `stop-targeted-sentiment-detection-job`.

### AWS CLI

Para detener un trabajo asíncrono de detección de opiniones segmentadas

El siguiente `stop-targeted-sentiment-detection-job` ejemplo detiene un trabajo de detección de opiniones segmentado asíncrono en curso. Si el estado actual del trabajo es, `IN_PROGRESS` el trabajo se marca para su finalización y se coloca en ese estado. `STOP_REQUESTED` Si el trabajo se completa antes de poder detenerlo, se coloca en ese `COMPLETED` estado.

```
aws comprehend stop-targeted-sentiment-detection-job \  
--job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Salida:

```
{  
  "JobId": "123456abcdeb0e11022f22a11EXAMPLE",  
  "JobStatus": "STOP_REQUESTED"  
}
```

Para obtener más información, consulte [Análisis asincrónico para información de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [StopTargetedSentimentDetectionJob](#) la Referencia de AWS CLI comandos.

## stop-training-document-classifier

En el siguiente ejemplo de código, se muestra cómo usar `stop-training-document-classifier`.

### AWS CLI

Para detener el entrenamiento de un modelo de clasificador de documentos

El siguiente `stop-training-document-classifier` ejemplo detiene el entrenamiento de un modelo de clasificador de documentos mientras está en curso.

```
aws comprehend stop-training-document-classifier
  --document-classifier-arn arn:aws:comprehend:us-west-2:111122223333:document-
  classifier/example-classifier
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación y administración de modelos personalizados](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [StopTrainingDocumentClassifier](#) de AWS CLI comandos.

## stop-training-entity-recognizer

En el siguiente ejemplo de código, se muestra cómo usar `stop-training-entity-recognizer`.

### AWS CLI

Para detener el entrenamiento de un modelo de reconocimiento de entidades

El siguiente `stop-training-entity-recognizer` ejemplo detiene el entrenamiento de un modelo de reconocimiento de entidades mientras está en curso.

```
aws comprehend stop-training-entity-recognizer
```

```
--entity-recognizer-arn "arn:aws:comprehend:us-west-2:111122223333:entity-recognizer/examplerecognizer1"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación y administración de modelos personalizados](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [StopTrainingEntityRecognizer](#) de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

### AWS CLI

Ejemplo 1: etiquetar un recurso

En el siguiente tag-resource ejemplo, se añade una sola etiqueta a un recurso de Amazon Comprehend.

```
aws comprehend tag-resource \  
  --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier/version/1 \  
  --tags Key=Location,Value=Seattle
```

Este comando no tiene salida.

Para obtener más información, consulte [Etiquetar sus recursos](#) en la Guía para desarrolladores de Amazon Comprehend.

Ejemplo 2: Para añadir varias etiquetas a un recurso

En el siguiente tag-resource ejemplo, se añaden varias etiquetas a un recurso de Amazon Comprehend.

```
aws comprehend tag-resource \  
  --resource-arn "arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier/version/1" \  
  --tags Key=Location,Value=Seattle Key=Department,Value=Finance
```

Este comando no tiene salida.

Para obtener más información, consulte [Etiquetar sus recursos](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [TagResource](#) de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Ejemplo 1: Para eliminar una sola etiqueta de un recurso

El siguiente `untag-resource` ejemplo elimina una sola etiqueta de un recurso de Amazon Comprehend.

```
aws comprehend untag-resource \  
  --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier/version/1  
  --tag-keys Location
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar sus recursos](#) en la Guía para desarrolladores de Amazon Comprehend.

Ejemplo 2: Para eliminar varias etiquetas de un recurso

El siguiente `untag-resource` ejemplo elimina varias etiquetas de un recurso de Amazon Comprehend.

```
aws comprehend untag-resource \  
  --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/  
example-classifier/version/1  
  --tag-keys Location Department
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar sus recursos](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [UntagResource](#) de AWS CLI comandos.

## update-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `update-endpoint`.

### AWS CLI

Ejemplo 1: Para actualizar las unidades de inferencia de un punto final

El siguiente `update-endpoint` ejemplo actualiza la información sobre un punto final. En este ejemplo, se aumenta el número de unidades de inferencia.

```
aws comprehend update-endpoint \  
  --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-  
endpoint/example-classifier-endpoint \  
  --desired-inference-units 2
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de puntos de conexión de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

Ejemplo 2: Para actualizar el modelo activo de un punto final

El siguiente `update-endpoint` ejemplo actualiza la información sobre un punto final. En este ejemplo, se cambia el modelo activo.

```
aws comprehend update-endpoint \  
  --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-  
endpoint/example-classifier-endpoint \  
  --active-model-arn arn:aws:comprehend:us-west-2:111122223333:document-  
classifier/example-classifier-new
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de puntos de conexión de Amazon Comprehend](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte [UpdateEndpoint](#) la Referencia de AWS CLI comandos.

## update-flywheel

En el siguiente ejemplo de código, se muestra cómo usar `update-flywheel`.

### AWS CLI

Para actualizar la configuración de un volante

En el siguiente `update-flywheel` ejemplo se actualiza la configuración de un volante. En este ejemplo, se actualiza el modelo activo del volante.

```
aws comprehend update-flywheel \  
  --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel-1 \  
  --active-model-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/example-classifier/version/new-example-classifier-model
```

Salida:

```
{  
  "FlywheelProperties": {  
    "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-entity",  
    "ActiveModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/example-classifier/version/new-example-classifier-model",  
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-example-role",  
    "TaskConfig": {  
      "LanguageCode": "en",  
      "DocumentClassificationConfig": {  
        "Mode": "MULTI_CLASS"  
      }  
    },  
    "DataLakeS3Uri": "s3://DOC-EXAMPLE-BUCKET/flywheel-entity/schemaVersion=1/20230616T200543Z/",  
    "DataSecurityConfig": {},  
    "Status": "ACTIVE",  
    "ModelType": "DOCUMENT_CLASSIFIER",  
    "CreationTime": "2023-06-16T20:05:43.242000+00:00",  
    "LastModifiedTime": "2023-06-19T04:00:43.027000+00:00",  
    "LatestFlywheelIteration": "20230619T040032Z"  
  }  
}
```

Para obtener más información, consulte la [descripción general de Flywheel](#) en la Guía para desarrolladores de Amazon Comprehend.

- Para API obtener más información, consulte la Referencia [UpdateFlywheel](#) de AWS CLI comandos.

## Ejemplos de Amazon Comprehend Medical utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con Amazon Comprehend Medical.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **describe-entities-detection-v2-job**

En el siguiente ejemplo de código, se muestra cómo usar `describe-entities-detection-v2-job`.

AWS CLI

Para describir un trabajo de detección de entidades

El siguiente `describe-entities-detection-v2-job` ejemplo muestra las propiedades asociadas a un trabajo de detección de entidades asincrónico.

```
aws comprehendmedical describe-entities-detection-v2-job \  
  --job-id "ab9887877365fe70299089371c043b96"
```

Salida:

```
{
```



```

"ComprehendMedicalAsyncJobProperties": {
  "JobId": "ab9887877365fe70299089371c043b96",
  "JobStatus": "COMPLETED",
  "SubmitTime": "2020-03-18T21:20:15.614000+00:00",
  "EndTime": "2020-03-18T21:27:07.350000+00:00",
  "ExpirationTime": "2020-07-16T21:20:15+00:00",
  "InputDataConfig": {
    "S3Bucket": "comp-med-input",
    "S3Key": ""
  },
  "OutputDataConfig": {
    "S3Bucket": "comp-med-output",
    "S3Key": "867139942017-EntitiesDetection-
ab9887877365fe70299089371c043b96/"
  },
  "LanguageCode": "en",
  "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
  "ModelVersion": "DetectEntitiesModelV20190930"
}
}

```

Para obtener más información, consulte [Batch APIs](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte [DescribeEntitiesDetectionV2Job](#) en AWS CLI Command Reference.

## describe-icd10-cm-inference-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-icd10-cm-inference-job`.

### AWS CLI

Para describir un trabajo de inferencia de ICD -10-CM

El siguiente `describe-icd10-cm-inference-job` ejemplo describe las propiedades del trabajo de inferencia solicitado con el identificador de trabajo especificado.

```

aws comprehendmedical describe-icd10-cm-inference-job \
  --job-id "5780034166536cdb52ffa3295a1b00a7"

```

Salida:

```
{
  "ComprehendMedicalAsyncJobProperties": {
    "JobId": "5780034166536cdb52ffa3295a1b00a7",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2020-05-18T21:20:15.614000+00:00",
    "EndTime": "2020-05-18T21:27:07.350000+00:00",
    "ExpirationTime": "2020-09-16T21:20:15+00:00",
    "InputDataConfig": {
      "S3Bucket": "comp-med-input",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "OutputDataConfig": {
      "S3Bucket": "comp-med-output",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "0.1.0"
  }
}
```

Para obtener más información, consulte [Ontology que vincula el análisis por lotes](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la sección [DescribeCdmInferenceJob](#) en AWS CLI Command Reference.

## describe-phi-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-phi-detection-job`.

### AWS CLI

Para describir un trabajo PHI de detección

En el siguiente `describe-phi-detection-job` ejemplo, se muestran las propiedades asociadas a un trabajo de detección asíncrono de información de salud protegida (PHI).

```
aws comprehendmedical describe-phi-detection-job \
  --job-id "4750034166536cdb52ffa3295a1b00a3"
```

**Salida:**

```
{
  "ComprehendMedicalAsyncJobProperties": {
    "JobId": "4750034166536cdb52ffa3295a1b00a3",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2020-03-19T20:38:37.594000+00:00",
    "EndTime": "2020-03-19T20:45:07.894000+00:00",
    "ExpirationTime": "2020-07-17T20:38:37+00:00",
    "InputDataConfig": {
      "S3Bucket": "comp-med-input",
      "S3Key": ""
    },
    "OutputDataConfig": {
      "S3Bucket": "comp-med-output",
      "S3Key": "867139942017-PHIDetection-4750034166536cdb52ffa3295a1b00a3/"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "PHIModelV20190903"
  }
}
```

Para obtener más información, consulte [Batch APIs](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte [DescribePhiDetectionJob](#) la Referencia de AWS CLI comandos.

**describe-rx-norm-inference-job**

En el siguiente ejemplo de código, se muestra cómo usar `describe-rx-norm-inference-job`.

**AWS CLI**

Para describir un trabajo de RxNorm inferencia

El siguiente `describe-rx-norm-inference-job` ejemplo describe las propiedades del trabajo de inferencia solicitado con el identificador de trabajo especificado.

```
aws comprehendmedical describe-rx-norm-inference-job \
```

```
--job-id "eg8199877365fc70299089371c043b96"
```

Salida:

```
{
  "ComprehendMedicalAsyncJobProperties": {
    "JobId": "g8199877365fc70299089371c043b96",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2020-05-18T21:20:15.614000+00:00",
    "EndTime": "2020-05-18T21:27:07.350000+00:00",
    "ExpirationTime": "2020-09-16T21:20:15+00:00",
    "InputDataConfig": {
      "S3Bucket": "comp-med-input",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "OutputDataConfig": {
      "S3Bucket": "comp-med-output",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "0.0.0"
  }
}
```

Para obtener más información, consulte [Ontology que vincula el análisis por lotes](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la Referencia [DescribeRxNormInferenceJob](#) de AWS CLI comandos.

## describe-snomedct-inference-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-snomedct-inference-job`.

### AWS CLI

Para describir un trabajo de inferencia de SNOMED tomografía computarizada

El siguiente `describe-snomedct-inference-job` ejemplo describe las propiedades del trabajo de inferencia solicitado con el identificador de trabajo especificado.

```
aws comprehendmedical describe-snomedct-inference-job \  
--job-id "2630034166536cdb52ffa3295a1b00a7"
```

Salida:

```
{  
  "ComprehendMedicalAsyncJobProperties": {  
    "JobId": "2630034166536cdb52ffa3295a1b00a7",  
    "JobStatus": "COMPLETED",  
    "SubmitTime": "2021-12-18T21:20:15.614000+00:00",  
    "EndTime": "2021-12-18T21:27:07.350000+00:00",  
    "ExpirationTime": "2022-05-16T21:20:15+00:00",  
    "InputDataConfig": {  
      "S3Bucket": "comp-med-input",  
      "S3Key": "AKIAIOSFODNN7EXAMPLE"  
    },  
    "OutputDataConfig": {  
      "S3Bucket": "comp-med-output",  
      "S3Key": "AKIAIOSFODNN7EXAMPLE"  
    },  
    "LanguageCode": "en",  
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/  
ComprehendMedicalBatchProcessingRole",  
    "ModelVersion": "0.1.0"  
  }  
}
```

Para obtener más información, consulte [Ontology que vincula el análisis por lotes](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la Referencia [DescribeSnomedctInferenceJob](#) de AWS CLI comandos.

## detect-entities-v2

En el siguiente ejemplo de código, se muestra cómo usar `detect-entities-v2`.

### AWS CLI

Ejemplo 1: Para detectar entidades directamente desde el texto

El siguiente `detect-entities-v2` ejemplo muestra las entidades detectadas y las etiqueta según su tipo, directamente a partir del texto introducido.

```
aws comprehendmedical detect-entities-v2 \  
  --text "Sleeping trouble on present dosage of Clonidine. Severe rash on face and  
  leg, slightly itchy."
```

Salida:

```
{  
  "Id": 0,  
  "BeginOffset": 38,  
  "EndOffset": 47,  
  "Score": 0.9942955374717712,  
  "Text": "Clonidine",  
  "Category": "MEDICATION",  
  "Type": "GENERIC_NAME",  
  "Traits": []  
}
```

Para obtener más información, consulte [Detect Entities versión 2](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

Ejemplo 2: Para detectar entidades desde la ruta de un archivo

El siguiente `detect-entities-v2` ejemplo muestra las entidades detectadas y las etiqueta según el tipo de ruta de un archivo.

```
aws comprehendmedical detect-entities-v2 \  
  --text file://medical_entities.txt
```

Contenidos de `medical_entities.txt`:

```
{  
  "Sleeping trouble on present dosage of Clonidine. Severe rash on face and leg,  
  slightly itchy."  
}
```

Salida:

```
{
```

```
"Id": 0,  
"BeginOffset": 38,  
"EndOffset": 47,  
"Score": 0.9942955374717712,  
"Text": "Clonidine",  
"Category": "MEDICATION",  
"Type": "GENERIC_NAME",  
"Traits": []  
}
```

Para obtener más información, consulte [Detect Entities versión 2](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la [DetectEntitiesversión 2](#) en la Referencia de AWS CLI comandos.

## detect-phi

En el siguiente ejemplo de código, se muestra cómo usar `detect-phi`.

### AWS CLI

Ejemplo 1: Para detectar información de salud protegida (PHI) directamente a partir del texto

En el siguiente `detect-phi` ejemplo, se muestran las entidades de información de salud protegida detectadas (PHI) directamente a partir del texto introducido.

```
aws comprehendmedical detect-phi \  
  --text "Patient Carlos Salazar presented with rash on his upper extremities and  
  dry cough. He lives at 100 Main Street, Anytown, USA where he works from his home  
  as a carpenter."
```

Salida:

```
{  
  "Entities": [  
    {  
      "Id": 0,  
      "BeginOffset": 8,  
      "EndOffset": 21,  
      "Score": 0.9914507269859314,
```

```

        "Text": "Carlos Salazar",
        "Category": "PROTECTED_HEALTH_INFORMATION",
        "Type": "NAME",
        "Traits": []
    },
    {
        "Id": 1,
        "BeginOffset": 94,
        "EndOffset": 109,
        "Score": 0.871849775314331,
        "Text": "100 Main Street, Anytown, USA",
        "Category": "PROTECTED_HEALTH_INFORMATION",
        "Type": "ADDRESS",
        "Traits": []
    },
    {
        "Id": 2,
        "BeginOffset": 145,
        "EndOffset": 154,
        "Score": 0.8302185535430908,
        "Text": "carpenter",
        "Category": "PROTECTED_HEALTH_INFORMATION",
        "Type": "PROFESSION",
        "Traits": []
    }
],
"ModelVersion": "0.0.0"
}

```

Para obtener más información, consulte [Detect PHI](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

Ejemplo 2: Para detectar, proteja la información de salud (PHI) directamente desde la ruta de un archivo

El siguiente detect-phi ejemplo muestra las entidades de información de salud protegida (PHI) detectadas a partir de una ruta de archivo.

```

aws comprehendmedical detect-phi \
  --text file://phi.txt

```

Contenidos de phi.txt:



"Patient Carlos Salazar presented with a rash on his upper extremities and a dry cough. He lives at 100 Main Street, Anytown, USA, where he works from his home as a carpenter."

Salida:

```
{
  "Entities": [
    {
      "Id": 0,
      "BeginOffset": 8,
      "EndOffset": 21,
      "Score": 0.9914507269859314,
      "Text": "Carlos Salazar",
      "Category": "PROTECTED_HEALTH_INFORMATION",
      "Type": "NAME",
      "Traits": []
    },
    {
      "Id": 1,
      "BeginOffset": 94,
      "EndOffset": 109,
      "Score": 0.871849775314331,
      "Text": "100 Main Street, Anytown, USA",
      "Category": "PROTECTED_HEALTH_INFORMATION",
      "Type": "ADDRESS",
      "Traits": []
    },
    {
      "Id": 2,
      "BeginOffset": 145,
      "EndOffset": 154,
      "Score": 0.8302185535430908,
      "Text": "carpenter",
      "Category": "PROTECTED_HEALTH_INFORMATION",
      "Type": "PROFESSION",
      "Traits": []
    }
  ],
  "ModelVersion": "0.0.0"
}
```

Para obtener más información, consulte [Detect PHI](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte [DetectPhi](#) la Referencia de AWS CLI comandos.

## **infer-icd10-cm**

En el siguiente ejemplo de código, se muestra cómo usar `infer-icd10-cm`.

### AWS CLI

Ejemplo 1: Para detectar entidades relacionadas con afecciones médicas y establecer un enlace a la ontología ICD -10-CM directamente desde el texto

El siguiente `infer-icd10-cm` ejemplo etiqueta las entidades de afecciones médicas detectadas y las vincula con los códigos de la edición de 2019 de la Modificación Clínica de la Clasificación Internacional de Enfermedades (ICD-10-CM).

```
aws comprehendmedical infer-icd10-cm \  
  --text "The patient complains of abdominal pain, has a long-standing history of  
  diabetes treated with Micronase daily."
```

Salida:

```
{  
  "Entities": [  
    {  
      "Id": 0,  
      "Text": "abdominal pain",  
      "Category": "MEDICAL_CONDITION",  
      "Type": "DX_NAME",  
      "Score": 0.9475538730621338,  
      "BeginOffset": 28,  
      "EndOffset": 42,  
      "Attributes": [],  
      "Traits": [  
        {  
          "Name": "SYMPTOM",  
          "Score": 0.6724207401275635  
        }  
      ],  
      "ICD10CMConcepts": [  
        {
```

```

        "Description": "Unspecified abdominal pain",
        "Code": "R10.9",
        "Score": 0.6904221177101135
    },
    {
        "Description": "Epigastric pain",
        "Code": "R10.13",
        "Score": 0.1364113688468933
    },
    {
        "Description": "Generalized abdominal pain",
        "Code": "R10.84",
        "Score": 0.12508003413677216
    },
    {
        "Description": "Left lower quadrant pain",
        "Code": "R10.32",
        "Score": 0.10063883662223816
    },
    {
        "Description": "Lower abdominal pain, unspecified",
        "Code": "R10.30",
        "Score": 0.09933677315711975
    }
]
},
{
    "Id": 1,
    "Text": "diabetes",
    "Category": "MEDICAL_CONDITION",
    "Type": "DX_NAME",
    "Score": 0.9899052977561951,
    "BeginOffset": 75,
    "EndOffset": 83,
    "Attributes": [],
    "Traits": [
        {
            "Name": "DIAGNOSIS",
            "Score": 0.9258432388305664
        }
    ],
    "ICD10CMConcepts": [
        {
            "Description": "Type 2 diabetes mellitus without complications",

```

```

        "Code": "E11.9",
        "Score": 0.7158446311950684
    },
    {
        "Description": "Family history of diabetes mellitus",
        "Code": "Z83.3",
        "Score": 0.5704703330993652
    },
    {
        "Description": "Family history of other endocrine, nutritional
and metabolic diseases",
        "Code": "Z83.49",
        "Score": 0.19856023788452148
    },
    {
        "Description": "Type 1 diabetes mellitus with ketoacidosis
without coma",
        "Code": "E10.10",
        "Score": 0.13285516202449799
    },
    {
        "Description": "Type 2 diabetes mellitus with hyperglycemia",
        "Code": "E11.65",
        "Score": 0.0993388369679451
    }
    ]
}
],
"ModelVersion": "0.1.0"
}

```

Para obtener más información, consulte [Inferir ICD1 0-CM](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

Ejemplo 2: Para detectar entidades relacionadas con afecciones médicas y establecer un enlace a la ontología ICD -10-CM desde una ruta de archivos

El siguiente `infer-icd-10-cm` ejemplo etiqueta las entidades de afecciones médicas detectadas y las vincula con los códigos de la edición de 2019 de la Modificación Clínica de la Clasificación Internacional de Enfermedades (ICD-10-CM).

```
aws comprehendmedical infer-icd10-cm \
```

```
--text file://icd10cm.txt
```

Contenidos de `icd10cm.txt`:

```
{  
  "The patient complains of abdominal pain, has a long-standing history of  
  diabetes treated with Micronase daily."  
}
```

Salida:

```
{  
  "Entities": [  
    {  
      "Id": 0,  
      "Text": "abdominal pain",  
      "Category": "MEDICAL_CONDITION",  
      "Type": "DX_NAME",  
      "Score": 0.9475538730621338,  
      "BeginOffset": 28,  
      "EndOffset": 42,  
      "Attributes": [],  
      "Traits": [  
        {  
          "Name": "SYMPTOM",  
          "Score": 0.6724207401275635  
        }  
      ],  
      "ICD10CMConcepts": [  
        {  
          "Description": "Unspecified abdominal pain",  
          "Code": "R10.9",  
          "Score": 0.6904221177101135  
        },  
        {  
          "Description": "Epigastric pain",  
          "Code": "R10.13",  
          "Score": 0.1364113688468933  
        },  
        {  
          "Description": "Generalized abdominal pain",  
          "Code": "R10.84",  
          "Score": 0.12508003413677216  
        }  
      ]  
    }  
  ]  
}
```

```
    },
    {
      "Description": "Left lower quadrant pain",
      "Code": "R10.32",
      "Score": 0.10063883662223816
    },
    {
      "Description": "Lower abdominal pain, unspecified",
      "Code": "R10.30",
      "Score": 0.09933677315711975
    }
  ]
},
{
  "Id": 1,
  "Text": "diabetes",
  "Category": "MEDICAL_CONDITION",
  "Type": "DX_NAME",
  "Score": 0.9899052977561951,
  "BeginOffset": 75,
  "EndOffset": 83,
  "Attributes": [],
  "Traits": [
    {
      "Name": "DIAGNOSIS",
      "Score": 0.9258432388305664
    }
  ],
  "ICD10CMConcepts": [
    {
      "Description": "Type 2 diabetes mellitus without complications",
      "Code": "E11.9",
      "Score": 0.7158446311950684
    },
    {
      "Description": "Family history of diabetes mellitus",
      "Code": "Z83.3",
      "Score": 0.5704703330993652
    },
    {
      "Description": "Family history of other endocrine, nutritional
and metabolic diseases",
      "Code": "Z83.49",
      "Score": 0.19856023788452148
    }
  ]
}
```

```

        },
        {
            "Description": "Type 1 diabetes mellitus with ketoacidosis
without coma",
            "Code": "E10.10",
            "Score": 0.13285516202449799
        },
        {
            "Description": "Type 2 diabetes mellitus with hyperglycemia",
            "Code": "E11.65",
            "Score": 0.0993388369679451
        }
    ]
}
],
"ModelVersion": "0.1.0"
}

```

Para obtener más información, consulte [Infer- ICD1 0-CM](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte [InferIcd10 cm](#) en la referencia de comandos.AWS CLI

## **infer-rx-norm**

En el siguiente ejemplo de código, se muestra cómo usar `infer-rx-norm`.

### AWS CLI

Ejemplo 1: Para detectar entidades medicamentosas y vincularlas RxNorm directamente desde el texto

El siguiente `infer-rx-norm` ejemplo muestra y etiqueta las entidades medicamentosas detectadas y las vincula a los identificadores conceptuales (RxCUI) de la RxNorm base de datos de la Biblioteca Nacional de Medicina.

```

aws comprehendmedical infer-rx-norm \
  --text "Patient reports taking Levothyroxine 125 micrograms p.o. once daily, but
denies taking Synthroid."

```

Salida:

```
{
  "Entities": [
    {
      "Id": 0,
      "Text": "Levothyroxine",
      "Category": "MEDICATION",
      "Type": "GENERIC_NAME",
      "Score": 0.9996285438537598,
      "BeginOffset": 23,
      "EndOffset": 36,
      "Attributes": [
        {
          "Type": "DOSAGE",
          "Score": 0.9892290830612183,
          "RelationshipScore": 0.9997978806495667,
          "Id": 1,
          "BeginOffset": 37,
          "EndOffset": 51,
          "Text": "125 micrograms",
          "Traits": []
        },
        {
          "Type": "ROUTE_OR_MODE",
          "Score": 0.9988924860954285,
          "RelationshipScore": 0.998291552066803,
          "Id": 2,
          "BeginOffset": 52,
          "EndOffset": 56,
          "Text": "p.o.",
          "Traits": []
        },
        {
          "Type": "FREQUENCY",
          "Score": 0.9953463673591614,
          "RelationshipScore": 0.9999889135360718,
          "Id": 3,
          "BeginOffset": 57,
          "EndOffset": 67,
          "Text": "once daily",
          "Traits": []
        }
      ]
    },
    {
      "Traits": [],
    }
  ]
}
```



```

    "RxNormConcepts": [
      {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet",
        "Code": "966224",
        "Score": 0.9912070631980896
      },
      {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Capsule",
        "Code": "966405",
        "Score": 0.8698278665542603
      },
      {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Synthroid]",
        "Code": "966191",
        "Score": 0.7448257803916931
      },
      {
        "Description": "levothyroxine",
        "Code": "10582",
        "Score": 0.7050482630729675
      },
      {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Levoxy1]",
        "Code": "966190",
        "Score": 0.6921631693840027
      }
    ]
  },
  {
    "Id": 4,
    "Text": "Synthroid",
    "Category": "MEDICATION",
    "Type": "BRAND_NAME",
    "Score": 0.9946461319923401,
    "BeginOffset": 86,
    "EndOffset": 95,
    "Attributes": [],
    "Traits": [
      {
        "Name": "NEGATION",
        "Score": 0.5167351961135864
      }
    ]
  }

```

```

    ],
    "RxNormConcepts": [
      {
        "Description": "Synthroid",
        "Code": "224920",
        "Score": 0.9462039470672607
      },
      {
        "Description": "Levothyroxine Sodium 0.088 MG Oral Tablet
[Synthroid]",
        "Code": "966282",
        "Score": 0.8309829235076904
      },
      {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Synthroid]",
        "Code": "966191",
        "Score": 0.4945160448551178
      },
      {
        "Description": "Levothyroxine Sodium 0.05 MG Oral Tablet
[Synthroid]",
        "Code": "966247",
        "Score": 0.3674522042274475
      },
      {
        "Description": "Levothyroxine Sodium 0.025 MG Oral Tablet
[Synthroid]",
        "Code": "966158",
        "Score": 0.2588822841644287
      }
    ]
  }
],
  "ModelVersion": "0.0.0"
}

```

Para obtener más información, consulte [Infer RxNorm](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

Ejemplo 2: Para detectar entidades medicamentosas y vincularlas RxNorm desde una ruta de archivo.

El siguiente `infer-rx-norm` ejemplo muestra y etiqueta las entidades farmacológicas detectadas y las vincula a los identificadores conceptuales (RxCUI) de la RxNorm base de datos de la Biblioteca Nacional de Medicina.

```
aws comprehendmedical infer-rx-norm \  
  --text file://rxnorm.txt
```

Contenidos de `rxnorm.txt`:

```
{  
  "Patient reports taking Levothyroxine 125 micrograms p.o. once daily, but denies  
  taking Synthroid."  
}
```

Salida:

```
{  
  "Entities": [  
    {  
      "Id": 0,  
      "Text": "Levothyroxine",  
      "Category": "MEDICATION",  
      "Type": "GENERIC_NAME",  
      "Score": 0.9996285438537598,  
      "BeginOffset": 23,  
      "EndOffset": 36,  
      "Attributes": [  
        {  
          "Type": "DOSAGE",  
          "Score": 0.9892290830612183,  
          "RelationshipScore": 0.9997978806495667,  
          "Id": 1,  
          "BeginOffset": 37,  
          "EndOffset": 51,  
          "Text": "125 micrograms",  
          "Traits": []  
        },  
        {  
          "Type": "ROUTE_OR_MODE",  
          "Score": 0.9988924860954285,  
          "RelationshipScore": 0.998291552066803,  
          "Id": 2,  
          "BeginOffset": 52,  
          "EndOffset": 66,  
          "Text": "p.o.",  
          "Traits": []  
        }  
      ]  
    }  
  ]  
}
```

```

        "BeginOffset": 52,
        "EndOffset": 56,
        "Text": "p.o.",
        "Traits": []
    },
    {
        "Type": "FREQUENCY",
        "Score": 0.9953463673591614,
        "RelationshipScore": 0.9999889135360718,
        "Id": 3,
        "BeginOffset": 57,
        "EndOffset": 67,
        "Text": "once daily",
        "Traits": []
    }
],
"Traits": [],
"RxNormConcepts": [
    {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet",
        "Code": "966224",
        "Score": 0.9912070631980896
    },
    {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Capsule",
        "Code": "966405",
        "Score": 0.8698278665542603
    },
    {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Synthroid]",
        "Code": "966191",
        "Score": 0.7448257803916931
    },
    {
        "Description": "levothyroxine",
        "Code": "10582",
        "Score": 0.7050482630729675
    },
    {
        "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Levoxy]",
        "Code": "966190",
        "Score": 0.6921631693840027
    }
]

```

```

    }
  ]
},
{
  "Id": 4,
  "Text": "Synthroid",
  "Category": "MEDICATION",
  "Type": "BRAND_NAME",
  "Score": 0.9946461319923401,
  "BeginOffset": 86,
  "EndOffset": 95,
  "Attributes": [],
  "Traits": [
    {
      "Name": "NEGATION",
      "Score": 0.5167351961135864
    }
  ],
  "RxNormConcepts": [
    {
      "Description": "Synthroid",
      "Code": "224920",
      "Score": 0.9462039470672607
    },
    {
      "Description": "Levothyroxine Sodium 0.088 MG Oral Tablet
[Synthroid]",
      "Code": "966282",
      "Score": 0.8309829235076904
    },
    {
      "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Synthroid]",
      "Code": "966191",
      "Score": 0.4945160448551178
    },
    {
      "Description": "Levothyroxine Sodium 0.05 MG Oral Tablet
[Synthroid]",
      "Code": "966247",
      "Score": 0.3674522042274475
    },
    {

```

```

        "Description": "Levothyroxine Sodium 0.025 MG Oral Tablet
[Synthroid]",
        "Code": "966158",
        "Score": 0.2588822841644287
    }
]
}
],
"ModelVersion": "0.0.0"
}

```

Para obtener más información, consulte [Infer RxNorm](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la Referencia [InferRxNorm](#) de AWS CLI comandos.

## infer-snomedct

En el siguiente ejemplo de código, se muestra cómo usar `infer-snomedct`.

### AWS CLI

Ejemplo: detectar entidades y vincularlas a la ontología SNOMED CT directamente desde el texto

El siguiente `infer-snomedct` ejemplo muestra cómo detectar entidades médicas y vincularlas con conceptos de la versión 2021-03 de la Nomenclatura Sistematizada de la Medicina, Términos Clínicos (CT). SNOMED

```

aws comprehendmedical infer-snomedct \
  --text "The patient complains of abdominal pain, has a long-standing history of
diabetes treated with Micronase daily."

```

Salida:

```

{
  "Entities": [
    {
      "Id": 3,
      "BeginOffset": 26,
      "EndOffset": 40,
      "Score": 0.9598260521888733,
      "Text": "abdominal pain",
      "Category": "MEDICAL_CONDITION",

```

```
    "Type": "DX_NAME",
    "Traits": [
      {
        "Name": "SYMPTOM",
        "Score": 0.6819021701812744
      }
    ]
  },
  {
    "Id": 4,
    "BeginOffset": 73,
    "EndOffset": 81,
    "Score": 0.9905840158462524,
    "Text": "diabetes",
    "Category": "MEDICAL_CONDITION",
    "Type": "DX_NAME",
    "Traits": [
      {
        "Name": "DIAGNOSIS",
        "Score": 0.9255214333534241
      }
    ]
  },
  {
    "Id": 1,
    "BeginOffset": 95,
    "EndOffset": 104,
    "Score": 0.6371926665306091,
    "Text": "Micronase",
    "Category": "MEDICATION",
    "Type": "BRAND_NAME",
    "Traits": [],
    "Attributes": [
      {
        "Type": "FREQUENCY",
        "Score": 0.9761165380477905,
        "RelationshipScore": 0.9984188079833984,
        "RelationshipType": "FREQUENCY",
        "Id": 2,
        "BeginOffset": 105,
        "EndOffset": 110,
        "Text": "daily",
        "Category": "MEDICATION",
        "Traits": []
      }
    ]
  }
}
```

```

    }
  ]
}
],
"UnmappedAttributes": [],
"ModelVersion": "1.0.0"
}

```

Para obtener más información, consulte [Infer SNOMEDCT](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la Referencia [InferSnomedct](#) de AWS CLI comandos.

## list-entities-detection-v2-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-entities-detection-v2-jobs`.

### AWS CLI

Para enumerar los trabajos de detección de entidades

En el siguiente `list-entities-detection-v2-jobs` ejemplo, se enumeran los trabajos de detección asíncrona actuales.

```
aws comprehendmedical list-entities-detection-v2-jobs
```

Salida:

```

{
  "ComprehendMedicalAsyncJobPropertiesList": [
    {
      "JobId": "ab9887877365fe70299089371c043b96",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2020-03-19T20:38:37.594000+00:00",
      "EndTime": "2020-03-19T20:45:07.894000+00:00",
      "ExpirationTime": "2020-07-17T20:38:37+00:00",
      "InputDataConfig": {
        "S3Bucket": "comp-med-input",
        "S3Key": ""
      },
      "OutputDataConfig": {

```



```

        "S3Bucket": "comp-med-output",
        "S3Key": "867139942017-EntitiesDetection-
ab9887877365fe70299089371c043b96/"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "DetectEntitiesModelV20190930"
}
]
}

```

Para obtener más información, consulte [Batch APIs](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte [ListEntitiesDetectionV2Jobs](#) en AWS CLI Command Reference.

## list-icd10-cm-inference-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-icd10-cm-inference-jobs`.

### AWS CLI

Para ver una lista de todos los trabajos de inferencia actuales de ICD -10-CM

El siguiente ejemplo muestra cómo la `list-icd10-cm-inference-jobs` operación devuelve una lista de los trabajos de inferencia por lotes asíncronos ICD actuales de 10 CM.

```
aws comprehendmedical list-icd10-cm-inference-jobs
```

Salida:

```

{
  "ComprehendMedicalAsyncJobPropertiesList": [
    {
      "JobId": "5780034166536cdb52ffa3295a1b00a7",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2020-05-19T20:38:37.594000+00:00",
      "EndTime": "2020-05-19T20:45:07.894000+00:00",
      "ExpirationTime": "2020-09-17T20:38:37+00:00",
      "InputDataConfig": {
        "S3Bucket": "comp-med-input",

```

```

        "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "OutputDataConfig": {
        "S3Bucket": "comp-med-output",
        "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "0.1.0"
}
]
}

```

Para obtener más información, consulte [Ontology que vincula el análisis por lotes](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la sección [ListId10 CmlInferenceJobs](#) en AWS CLI Command Reference.

## list-phi-detection-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-phi-detection-jobs`.

### AWS CLI

Para enumerar los trabajos de detección de información de salud protegida (PHI)

En el siguiente `list-phi-detection-jobs` ejemplo se enumeran los trabajos de detección de información de salud protegida (PHI) actuales

```
aws comprehendmedical list-phi-detection-jobs
```

Salida:

```

{
  "ComprehendMedicalAsyncJobPropertiesList": [
    {
      "JobId": "4750034166536cdb52ffa3295a1b00a3",
      "JobStatus": "COMPLETED",
      "SubmitTime": "2020-03-19T20:38:37.594000+00:00",
      "EndTime": "2020-03-19T20:45:07.894000+00:00",
      "ExpirationTime": "2020-07-17T20:38:37+00:00",
    }
  ]
}

```

```

        "InputDataConfig": {
            "S3Bucket": "comp-med-input",
            "S3Key": ""
        },
        "OutputDataConfig": {
            "S3Bucket": "comp-med-output",
            "S3Key": "867139942017-
PHIDetection-4750034166536cdb52ffa3295a1b00a3/"
        },
        "LanguageCode": "en",
        "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
        "ModelVersion": "PHIModelV20190903"
    }
]
}

```

Para obtener más información, consulte [Batch APIs](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte [ListPhiDetectionJobs](#) la Referencia de AWS CLI comandos.

## list-rx-norm-inference-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-rx-norm-inference-jobs`.

### AWS CLI

Para enumerar todos los trabajos de inferencia de Rx-Norm actuales

El siguiente ejemplo muestra cómo se `list-rx-norm-inference-jobs` devuelve una lista de los trabajos de inferencia por lotes de Rx-Norm asíncronos actuales.

```
aws comprehendmedical list-rx-norm-inference-jobs
```

Salida:

```

{
  "ComprehendMedicalAsyncJobPropertiesList": [
    {
      "JobId": "4980034166536cfb52gga3295a1b00a3",
      "JobStatus": "COMPLETED",

```

```

    "SubmitTime": "2020-05-19T20:38:37.594000+00:00",
    "EndTime": "2020-05-19T20:45:07.894000+00:00",
    "ExpirationTime": "2020-09-17T20:38:37+00:00",
    "InputDataConfig": {
      "S3Bucket": "comp-med-input",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "OutputDataConfig": {
      "S3Bucket": "comp-med-output",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "0.0.0"
  }
]
}

```

Para obtener más información, consulte [Ontology que vincula el análisis por lotes](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la Referencia [ListRxNormInferenceJobs](#) de AWS CLI comandos.

## list-snomedct-inference-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-snomedct-inference-jobs`.

### AWS CLI

Para enumerar todos los trabajos de inferencia de SNOMED tomografía computarizada

El siguiente ejemplo muestra cómo la `list-snomedct-inference-jobs` operación devuelve una lista de los trabajos de inferencia por lotes SNOMED CT asíncronos actuales.

```
aws comprehendmedical list-snomedct-inference-jobs
```

Salida:

```

{
  "ComprehendMedicalAsyncJobPropertiesList": [
    {

```

```

    "JobId": "5780034166536cdb52ffa3295a1b00a7",
    "JobStatus": "COMPLETED",
    "SubmitTime": "2020-05-19T20:38:37.594000+00:00",
    "EndTime": "2020-05-19T20:45:07.894000+00:00",
    "ExpirationTime": "2020-09-17T20:38:37+00:00",
    "InputDataConfig": {
      "S3Bucket": "comp-med-input",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "OutputDataConfig": {
      "S3Bucket": "comp-med-output",
      "S3Key": "AKIAIOSFODNN7EXAMPLE"
    },
    "LanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
    "ModelVersion": "0.1.0"
  }
]
}

```

Para obtener más información, consulte [Ontology que vincula el análisis por lotes](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la Referencia [ListSnomedctInferenceJobs](#) de AWS CLI comandos.

## start-entities-detection-v2-job

En el siguiente ejemplo de código, se muestra cómo usar `start-entities-detection-v2-job`.

### AWS CLI

Para iniciar un trabajo de detección de entidades

El siguiente `start-entities-detection-v2-job` ejemplo inicia un trabajo de detección de entidades asincrónico.

```

aws comprehendmedical start-entities-detection-v2-job \
  --input-data-config "S3Bucket=comp-med-input" \
  --output-data-config "S3Bucket=comp-med-output" \
  --data-access-role-arn arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole \

```

```
--language-code en
```

Salida:

```
{  
  "JobId": "ab9887877365fe70299089371c043b96"  
}
```

Para obtener más información, consulte [Batch APIs](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte [StartEntitiesDetectionV2Job](#) en AWS CLI Command Reference.

### **start-icd10-cm-inference-job**

En el siguiente ejemplo de código, se muestra cómo usar `start-icd10-cm-inference-job`.

AWS CLI

Para iniciar un trabajo de inferencia de ICD -10-CM

El siguiente `start-icd10-cm-inference-job` ejemplo inicia un trabajo de análisis por lotes de ICD inferencias de 10 CM.

```
aws comprehendmedical start-icd10-cm-inference-job \  
  --input-data-config "S3Bucket=comp-med-input" \  
  --output-data-config "S3Bucket=comp-med-output" \  
  --data-access-role-arn arn:aws:iam::867139942017:role/  
ComprehendMedicalBatchProcessingRole \  
  --language-code en
```

Salida:

```
{  
  "JobId": "ef7289877365fc70299089371c043b96"  
}
```

Para obtener más información, consulte [Ontology que vincula el análisis por lotes](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la sección [StartIcd10 CmInferenceJob](#) en AWS CLI Command Reference.

## start-phi-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `start-phi-detection-job`.

### AWS CLI

Para iniciar un trabajo PHI de detección

El siguiente `start-phi-detection-job` ejemplo inicia un trabajo de detección de PHI entidades asincrónico.

```
aws comprehendmedical start-phi-detection-job \  
  --input-data-config "S3Bucket=comp-med-input" \  
  --output-data-config "S3Bucket=comp-med-output" \  
  --data-access-role-arn arn:aws:iam::867139942017:role/  
  ComprehendMedicalBatchProcessingRole \  
  --language-code en
```

Salida:

```
{  
  "JobId": "ab9887877365fe70299089371c043b96"  
}
```

Para obtener más información, consulte [Batch APIs](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte [StartPhiDetectionJob](#) la Referencia de AWS CLI comandos.

## start-rx-norm-inference-job

En el siguiente ejemplo de código, se muestra cómo usar `start-rx-norm-inference-job`.

### AWS CLI

Para iniciar un trabajo de RxNorm inferencia

El siguiente `start-rx-norm-inference-job` ejemplo inicia un trabajo de análisis por lotes RxNorm de inferencias.

```
aws comprehendmedical start-rx-norm-inference-job \  
  --input-data-config "S3Bucket=comp-med-input" \  
  --output-data-config "S3Bucket=comp-med-output" \  
  --data-access-role-arn arn:aws:iam::867139942017:role/  
  ComprehendMedicalBatchProcessingRole \  
  --language-code en
```

Salida:

```
{  
  "JobId": "eg8199877365fc70299089371c043b96"  
}
```

Para obtener más información, consulte [Ontology que vincula el análisis por lotes](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la Referencia [StartRxNormInferenceJob](#) de AWS CLI comandos.

## **start-snomedct-inference-job**

En el siguiente ejemplo de código, se muestra cómo usar `start-snomedct-inference-job`.

AWS CLI

Para iniciar un trabajo de inferencia de SNOMED tomografía computarizada

El siguiente `start-snomedct-inference-job` ejemplo inicia un trabajo de análisis por lotes de inferencia por SNOMED tomografía computarizada.

```
aws comprehendmedical start-snomedct-inference-job \  
  --input-data-config "S3Bucket=comp-med-input" \  
  --output-data-config "S3Bucket=comp-med-output" \  
  --data-access-role-arn arn:aws:iam::867139942017:role/  
  ComprehendMedicalBatchProcessingRole \  
  --language-code en
```

Salida:



```
{
  "JobId": "dg7289877365fc70299089371c043b96"
}
```

Para obtener más información, consulte [Ontology que vincula el análisis por lotes](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la Referencia [StartSnomedctInferenceJob](#) de AWS CLI comandos.

## stop-entities-detection-v2-job

En el siguiente ejemplo de código, se muestra cómo usar `stop-entities-detection-v2-job`.

### AWS CLI

Para detener un trabajo de detección de entidades

El siguiente `stop-entities-detection-v2-job` ejemplo detiene un trabajo de detección de entidades asincrónico.

```
aws comprehendmedical stop-entities-detection-v2-job \
  --job-id "ab9887877365fe70299089371c043b96"
```

Salida:

```
{
  "JobId": "ab9887877365fe70299089371c043b96"
}
```

Para obtener más información, consulte [Batch APIs](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte [StopEntitiesDetectionV2Job](#) en AWS CLI Command Reference.

## stop-icd10-cm-inference-job

En el siguiente ejemplo de código, se muestra cómo usar `stop-icd10-cm-inference-job`.

## AWS CLI

Para detener un trabajo de inferencia de ICD -10-CM

El siguiente `stop-icd10-cm-inference-job` ejemplo detiene un trabajo de análisis por lotes de ICD inferencias de 10 CM.

```
aws comprehendmedical stop-icd10-cm-inference-job \  
  --job-id "4750034166536cdb52ffa3295a1b00a3"
```

Salida:

```
{  
  "JobId": "ef7289877365fc70299089371c043b96",  
}
```

Para obtener más información, consulte [Ontology que vincula el análisis por lotes](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la sección [StopIcd10 CmInferenceJob](#) en AWS CLI Command Reference.

## stop-phi-detection-job

En el siguiente ejemplo de código, se muestra cómo usar `stop-phi-detection-job`.

## AWS CLI

Para detener un trabajo de detección de información de salud protegida (PHI)

El siguiente `stop-phi-detection-job` ejemplo detiene un trabajo de detección asíncrono de información de salud protegida (PHI).

```
aws comprehendmedical stop-phi-detection-job \  
  --job-id "4750034166536cdb52ffa3295a1b00a3"
```

Salida:

```
{  
  "JobId": "ab9887877365fe70299089371c043b96"
```

```
}
```

Para obtener más información, consulte [Batch APIs](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte [StopPhiDetectionJob](#) la Referencia de AWS CLI comandos.

## **stop-rx-norm-inference-job**

En el siguiente ejemplo de código, se muestra cómo usar `stop-rx-norm-inference-job`.

### AWS CLI

Para detener un trabajo de RxNorm inferencia

El siguiente `stop-rx-norm-inference-job` ejemplo detiene un trabajo de análisis por ICD lotes de inferencias de 10 CM.

```
aws comprehendmedical stop-rx-norm-inference-job \  
  --job-id "eg8199877365fc70299089371c043b96"
```

Salida:

```
{  
  "JobId": "eg8199877365fc70299089371c043b96",  
}
```

Para obtener más información, consulte [Ontology que vincula el análisis por lotes](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la Referencia [StopRxNormInferenceJob](#) de AWS CLI comandos.

## **stop-snomedct-inference-job**

En el siguiente ejemplo de código, se muestra cómo usar `stop-snomedct-inference-job`.

### AWS CLI

Para detener un trabajo de inferencia de SNOMED tomografía computarizada

El siguiente `stop-snomedct-inference-job` ejemplo detiene un trabajo de análisis por lotes de inferencias por SNOMED tomografía computarizada.

```
aws comprehendmedical stop-snomedct-inference-job \  
  --job-id "8750034166436cdb52ffa3295a1b00a1"
```

Salida:

```
{  
  "JobId": "8750034166436cdb52ffa3295a1b00a1",  
}
```

Para obtener más información, consulte [Ontology que vincula el análisis por lotes](#) en la Guía para desarrolladores de Amazon Comprehend Medical.

- Para API obtener más información, consulte la Referencia [StopSnomedctInferenceJob](#) de AWS CLI comandos.

## AWS Config ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Config.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

Acciones

### **delete-config-rule**

En el siguiente ejemplo de código, se muestra cómo usar `delete-config-rule`.

## AWS CLI

Para eliminar una regla de AWS Config

El siguiente comando elimina una regla de AWS Config denominada `MyConfigRule`:

```
aws configservice delete-config-rule --config-rule-name MyConfigRule
```

- Para API obtener más información, consulte [DeleteConfigRule](#) la Referencia de AWS CLI comandos.

## `delete-delivery-channel`

En el siguiente ejemplo de código, se muestra cómo usar `delete-delivery-channel`.

### AWS CLI

Para eliminar un canal de entrega

El siguiente comando elimina el canal de entrega predeterminado:

```
aws configservice delete-delivery-channel --delivery-channel-name default
```

- Para API obtener más información, consulte [DeleteDeliveryChannel](#) la Referencia de AWS CLI comandos.

## `delete-evaluation-results`

En el siguiente ejemplo de código, se muestra cómo usar `delete-evaluation-results`.

### AWS CLI

Para eliminar manualmente los resultados de la evaluación

El siguiente comando elimina los resultados de la evaluación actual de la regla AWS gestionada `s3-bucket-versioning-enabled`

```
aws configservice delete-evaluation-results --config-rule-name s3-bucket-versioning-enabled
```

- Para API obtener más información, consulte [DeleteEvaluationResults](#) la Referencia de AWS CLI comandos.

## **deliver-config-snapshot**

En el siguiente ejemplo de código, se muestra cómo usar `deliver-config-snapshot`.

### AWS CLI

Para entregar una instantánea de la configuración

El siguiente comando entrega una instantánea de la configuración al bucket de Amazon S3 que pertenece al canal de entrega predeterminado:

```
aws configservice deliver-config-snapshot --delivery-channel-name default
```

Salida:

```
{
  "configSnapshotId": "d0333b00-a683-44af-921e-examplefb794"
}
```

- Para API obtener más información, consulte [DeliverConfigSnapshot](#) la Referencia de AWS CLI comandos.

## **describe-compliance-by-config-rule**

En el siguiente ejemplo de código, se muestra cómo usar `describe-compliance-by-config-rule`.

### AWS CLI

Para obtener información sobre el cumplimiento de tus reglas de AWS Config

El siguiente comando devuelve la información de conformidad de cada regla de AWS Config que uno o más AWS recursos infrinjan:

```
aws configservice describe-compliance-by-config-rule --compliance-  
types NON_COMPLIANT
```

En el resultado, el valor de cada CappedCount atributo indica cuántos recursos no cumplen con la regla relacionada. Por ejemplo, el siguiente resultado indica que tres recursos no cumplen con la regla indicada InstanceTypesAreT2micro.

Salida:

```
{
  "ComplianceByConfigRules": [
    {
      "Compliance": {
        "ComplianceContributorCount": {
          "CappedCount": 3,
          "CapExceeded": false
        },
        "ComplianceType": "NON_COMPLIANT"
      },
      "ConfigRuleName": "InstanceTypesAreT2micro"
    },
    {
      "Compliance": {
        "ComplianceContributorCount": {
          "CappedCount": 10,
          "CapExceeded": false
        },
        "ComplianceType": "NON_COMPLIANT"
      },
      "ConfigRuleName": "RequiredTagsForVolumes"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeComplianceByConfigRule](#) la Referencia de AWS CLI comandos.

## describe-compliance-by-resource

En el siguiente ejemplo de código, se muestra cómo usar describe-compliance-by-resource.

AWS CLI

Para obtener información sobre el cumplimiento de sus AWS recursos

El siguiente comando devuelve la información de conformidad de cada EC2 instancia registrada por AWS Config y que infringe una o más reglas:

```
aws configservice describe-compliance-by-resource --resource-type AWS::EC2::Instance
--compliance-types NON_COMPLIANT
```

En el resultado, el valor de cada CappedCount atributo indica cuántas reglas infringe el recurso. Por ejemplo, el siguiente resultado indica que la instancia i-1a2b3c4d infringe dos reglas.

Salida:

```
{
  "ComplianceByResources": [
    {
      "ResourceType": "AWS::EC2::Instance",
      "ResourceId": "i-1a2b3c4d",
      "Compliance": {
        "ComplianceContributorCount": {
          "CappedCount": 2,
          "CapExceeded": false
        },
        "ComplianceType": "NON_COMPLIANT"
      }
    },
    {
      "ResourceType": "AWS::EC2::Instance",
      "ResourceId": "i-2a2b3c4d ",
      "Compliance": {
        "ComplianceContributorCount": {
          "CappedCount": 3,
          "CapExceeded": false
        },
        "ComplianceType": "NON_COMPLIANT"
      }
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeComplianceByResource](#) la Referencia de AWS CLI comandos.



## describe-config-rule-evaluation-status

En el siguiente ejemplo de código, se muestra cómo usar `describe-config-rule-evaluation-status`.

### AWS CLI

Para obtener información de estado de una regla de AWS Config

El siguiente comando devuelve la información de estado de una regla de AWS Config denominada `MyConfigRule`:

```
aws configservice describe-config-rule-evaluation-status --config-rule-names MyConfigRule
```

Salida:

```
{
  "ConfigRulesEvaluationStatus": [
    {
      "ConfigRuleArn": "arn:aws:config:us-east-1:123456789012:config-rule/config-rule-abcdef",
      "FirstActivatedTime": 1450311703.844,
      "ConfigRuleId": "config-rule-abcdef",
      "LastSuccessfulInvocationTime": 1450314643.156,
      "ConfigRuleName": "MyConfigRule"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeConfigRuleEvaluationStatus](#) la Referencia de AWS CLI comandos.

## describe-config-rules

En el siguiente ejemplo de código, se muestra cómo usar `describe-config-rules`.

### AWS CLI

Para obtener detalles de una regla de AWS Config

El siguiente comando devuelve los detalles de una regla de AWS Config denominada `InstanceTypesAreT2micro`:

```
aws configservice describe-config-rules --config-rule-names InstanceTypesAreT2micro
```

Salida:

```
{
  "ConfigRules": [
    {
      "ConfigRuleState": "ACTIVE",
      "Description": "Evaluates whether EC2 instances are the t2.micro type.",
      "ConfigRuleName": "InstanceTypesAreT2micro",
      "ConfigRuleArn": "arn:aws:config:us-east-1:123456789012:config-rule/
config-rule-abcdef",
      "Source": {
        "Owner": "CUSTOM_LAMBDA",
        "SourceIdentifier": "arn:aws:lambda:us-
east-1:123456789012:function:InstanceTypeCheck",
        "SourceDetails": [
          {
            "EventSource": "aws.config",
            "MessageType": "ConfigurationItemChangeNotification"
          }
        ]
      },
      "InputParameters": "{\"desiredInstanceType\":\"t2.micro\"}",
      "Scope": {
        "ComplianceResourceTypes": [
          "AWS::EC2::Instance"
        ]
      },
      "ConfigRuleId": "config-rule-abcdef"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeConfigRules](#) la Referencia de AWS CLI comandos.

## describe-configuration-recorder-status

En el siguiente ejemplo de código, se muestra cómo usar `describe-configuration-recorder-status`.

### AWS CLI

Para obtener información sobre el estado de la grabadora de configuración

El siguiente comando devuelve el estado de la grabadora de configuración predeterminada:

```
aws configservice describe-configuration-recorder-status
```

Salida:

```
{
  "ConfigurationRecordersStatus": [
    {
      "name": "default",
      "lastStatus": "SUCCESS",
      "recording": true,
      "lastStatusChangeTime": 1452193834.344,
      "lastStartTime": 1441039997.819,
      "lastStopTime": 1441039992.835
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeConfigurationRecorderStatus](#) la Referencia de AWS CLI comandos.

## describe-configuration-recorders

En el siguiente ejemplo de código, se muestra cómo usar `describe-configuration-recorders`.

### AWS CLI

Para obtener detalles sobre la grabadora de configuración

El siguiente comando devuelve detalles sobre la grabadora de configuración predeterminada:

```
aws configservice describe-configuration-records
```

Salida:

```
{
  "ConfigurationRecorders": [
    {
      "recordingGroup": {
        "allSupported": true,
        "resourceTypes": [],
        "includeGlobalResourceTypes": true
      },
      "roleARN": "arn:aws:iam::123456789012:role/config-ConfigRole-
A1B2C3D4E5F6",
      "name": "default"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeConfigurationRecorders](#) la Referencia de AWS CLI comandos.

## describe-delivery-channel-status

En el siguiente ejemplo de código, se muestra cómo usar `describe-delivery-channel-status`.

AWS CLI

Para obtener información sobre el estado del canal de entrega

El siguiente comando devuelve el estado del canal de entrega:

```
aws configservice describe-delivery-channel-status
```

Salida:

```
{
  "DeliveryChannelsStatus": [
    {
      "configStreamDeliveryInfo": {
```

```
        "lastStatusChangeTime": 1452193834.381,  
        "lastStatus": "SUCCESS"  
    },  
    "configHistoryDeliveryInfo": {  
        "lastSuccessfulTime": 1450317838.412,  
        "lastStatus": "SUCCESS",  
        "lastAttemptTime": 1450317838.412  
    },  
    "configSnapshotDeliveryInfo": {  
        "lastSuccessfulTime": 1452185597.094,  
        "lastStatus": "SUCCESS",  
        "lastAttemptTime": 1452185597.094  
    },  
    "name": "default"  
  }  
]  
}
```

- Para API obtener más información, consulte [DescribeDeliveryChannelStatus](#) la Referencia de AWS CLI comandos.

## describe-delivery-channels

En el siguiente ejemplo de código, se muestra cómo usar `describe-delivery-channels`.

### AWS CLI

Para obtener detalles sobre el canal de entrega

El siguiente comando devuelve detalles sobre el canal de entrega:

```
aws configservice describe-delivery-channels
```

Salida:

```
{  
  "DeliveryChannels": [  
    {  
      "snsTopicARN": "arn:aws:sns:us-east-1:123456789012:config-topic",  
      "name": "default",  
      "s3BucketName": "config-bucket-123456789012"  
    }  
  ]  
}
```

```
]
}
```

- Para API obtener más información, consulte [DescribeDeliveryChannels](#) la Referencia de AWS CLI comandos.

## get-compliance-details-by-config-rule

En el siguiente ejemplo de código, se muestra cómo usar `get-compliance-details-by-config-rule`.

### AWS CLI

Para obtener los resultados de la evaluación de una regla de AWS Config

El siguiente comando devuelve los resultados de la evaluación de todos los recursos que no cumplen con una regla de AWS Config denominada `InstanceTypesAreT2micro`:

```
aws configservice get-compliance-details-by-config-rule --config-rule-
name InstanceTypesAreT2micro --compliance-types NON_COMPLIANT
```

Salida:

```
{
  "EvaluationResults": [
    {
      "EvaluationResultIdentifier": {
        "OrderingTimestamp": 1450314635.065,
        "EvaluationResultQualifier": {
          "ResourceType": "AWS::EC2::Instance",
          "ResourceId": "i-1a2b3c4d",
          "ConfigRuleName": "InstanceTypesAreT2micro"
        }
      },
      "ResultRecordedTime": 1450314645.261,
      "ConfigRuleInvokedTime": 1450314642.948,
      "ComplianceType": "NON_COMPLIANT"
    },
    {
      "EvaluationResultIdentifier": {
        "OrderingTimestamp": 1450314635.065,
        "EvaluationResultQualifier": {
```

```

        "ResourceType": "AWS::EC2::Instance",
        "ResourceId": "i-2a2b3c4d",
        "ConfigRuleName": "InstanceTypesAreT2micro"
    },
    "ResultRecordedTime": 1450314645.18,
    "ConfigRuleInvokedTime": 1450314642.902,
    "ComplianceType": "NON_COMPLIANT"
},
{
    "EvaluationResultIdentifier": {
        "OrderingTimestamp": 1450314635.065,
        "EvaluationResultQualifier": {
            "ResourceType": "AWS::EC2::Instance",
            "ResourceId": "i-3a2b3c4d",
            "ConfigRuleName": "InstanceTypesAreT2micro"
        }
    },
    "ResultRecordedTime": 1450314643.346,
    "ConfigRuleInvokedTime": 1450314643.124,
    "ComplianceType": "NON_COMPLIANT"
}
]
}

```

- Para API obtener más información, consulte [GetComplianceDetailsByConfigRule](#) la Referencia de AWS CLI comandos.

## get-compliance-details-by-resource

En el siguiente ejemplo de código, se muestra cómo usar `get-compliance-details-by-resource`.

### AWS CLI

Para obtener los resultados de la evaluación de un AWS recurso

El siguiente comando devuelve los resultados de la evaluación de cada regla que la EC2 instancia `i-1a2b3c4d` no cumpla:

```
aws configservice get-compliance-details-by-resource --resource-type AWS::EC2::Instance --resource-id i-1a2b3c4d --compliance-types NON_COMPLIANT
```

Salida:

```
{
  "EvaluationResults": [
    {
      "EvaluationResultIdentifier": {
        "OrderingTimestamp": 1450314635.065,
        "EvaluationResultQualifier": {
          "ResourceType": "AWS::EC2::Instance",
          "ResourceId": "i-1a2b3c4d",
          "ConfigRuleName": "InstanceTypesAreT2micro"
        }
      },
      "ResultRecordedTime": 1450314643.288,
      "ConfigRuleInvokedTime": 1450314643.034,
      "ComplianceType": "NON_COMPLIANT"
    },
    {
      "EvaluationResultIdentifier": {
        "OrderingTimestamp": 1450314635.065,
        "EvaluationResultQualifier": {
          "ResourceType": "AWS::EC2::Instance",
          "ResourceId": "i-1a2b3c4d",
          "ConfigRuleName": "RequiredTagForEC2Instances"
        }
      },
      "ResultRecordedTime": 1450314645.261,
      "ConfigRuleInvokedTime": 1450314642.948,
      "ComplianceType": "NON_COMPLIANT"
    }
  ]
}
```

- Para API obtener más información, consulte [GetComplianceDetailsByResource](#) la Referencia de AWS CLI comandos.

## get-compliance-summary-by-config-rule

En el siguiente ejemplo de código, se muestra cómo usar `get-compliance-summary-by-config-rule`.



## AWS CLI

Para obtener el resumen de conformidad de tus reglas de AWS Config

El siguiente comando devuelve el número de reglas que cumplen y el número que no lo son:

```
aws configservice get-compliance-summary-by-config-rule
```

En el resultado, el valor de cada CappedCount atributo indica cuántas reglas son conformes o no.

Salida:

```
{
  "ComplianceSummary": {
    "NonCompliantResourceCount": {
      "CappedCount": 3,
      "CapExceeded": false
    },
    "ComplianceSummaryTimestamp": 1452204131.493,
    "CompliantResourceCount": {
      "CappedCount": 2,
      "CapExceeded": false
    }
  }
}
```

- Para API obtener más información, consulte [GetComplianceSummaryByConfigRule](#) la Referencia de AWS CLI comandos.

### get-compliance-summary-by-resource-type

En el siguiente ejemplo de código, se muestra cómo usar `get-compliance-summary-by-resource-type`.

## AWS CLI

Para obtener el resumen de conformidad de todos los tipos de recursos

El siguiente comando devuelve el número de AWS recursos que no son conformes y el número que sí lo son:

```
aws configservice get-compliance-summary-by-resource-type
```

En el resultado, el valor de cada CappedCount atributo indica cuántos recursos son compatibles o no.

Salida:

```
{
  "ComplianceSummariesByResourceType": [
    {
      "ComplianceSummary": {
        "NonCompliantResourceCount": {
          "CappedCount": 16,
          "CapExceeded": false
        },
        "ComplianceSummaryTimestamp": 1453237464.543,
        "CompliantResourceCount": {
          "CappedCount": 10,
          "CapExceeded": false
        }
      }
    }
  ]
}
```

Para obtener el resumen de conformidad de un tipo de recurso específico

El siguiente comando devuelve el número de EC2 instancias que no son compatibles y el número que sí lo son:

```
aws configservice get-compliance-summary-by-resource-type --resource-
types AWS::EC2::Instance
```

En el resultado, el valor de cada CappedCount atributo indica cuántos recursos son compatibles o no.

Salida:

```
{
  "ComplianceSummariesByResourceType": [
    {
      "ResourceType": "AWS::EC2::Instance",
```

```

    "ComplianceSummary": {
      "NonCompliantResourceCount": {
        "CappedCount": 3,
        "CapExceeded": false
      },
      "ComplianceSummaryTimestamp": 1452204923.518,
      "CompliantResourceCount": {
        "CappedCount": 7,
        "CapExceeded": false
      }
    }
  ]
}

```

- Para API obtener más información, consulte [GetComplianceSummaryByResourceType](#) la Referencia de AWS CLI comandos.

## get-resource-config-history

En el siguiente ejemplo de código, se muestra cómo usar `get-resource-config-history`.

### AWS CLI

Para obtener el historial de configuración de un AWS recurso

El siguiente comando devuelve una lista de los elementos de configuración de una EC2 instancia con un ID de `i-1a2b3c4d`:

```
aws configservice get-resource-config-history --resource-type AWS::EC2::Instance --resource-id i-1a2b3c4d
```

- Para API obtener más información, consulta [GetResourceConfigHistory](#) la Referencia de AWS CLI comandos.

## get-status

En el siguiente ejemplo de código, se muestra cómo usar `get-status`.

### AWS CLI

Para obtener el estado de AWS Config

El siguiente comando devuelve el estado del canal de entrega y del grabador de configuración:

```
aws configservice get-status
```

Salida:

```
Configuration Recorders:

name: default
recorder: ON
last status: SUCCESS

Delivery Channels:

name: default
last stream delivery status: SUCCESS
last history delivery status: SUCCESS
last snapshot delivery status: SUCCESS
```

- Para API obtener más información, consulte [GetStatus](#) la Referencia de AWS CLI comandos.

## list-discovered-resources

En el siguiente ejemplo de código, se muestra cómo usar `list-discovered-resources`.

AWS CLI

Para enumerar los recursos que AWS Config ha descubierto

El siguiente comando muestra las EC2 instancias que AWS Config ha descubierto:

```
aws configservice list-discovered-resources --resource-type AWS::EC2::Instance
```

Salida:

```
{
  "resourceIdentifiers": [
    {
      "resourceType": "AWS::EC2::Instance",
      "resourceId": "i-1a2b3c4d"
    },
    {
```

```

        "resourceType": "AWS::EC2::Instance",
        "resourceId": "i-2a2b3c4d"
    },
    {
        "resourceType": "AWS::EC2::Instance",
        "resourceId": "i-3a2b3c4d"
    }
]
}

```

- Para API obtener más información, consulte [ListDiscoveredResources](#) la Referencia de AWS CLI comandos.

## put-config-rule

En el siguiente ejemplo de código, se muestra cómo usar `put-config-rule`.

### AWS CLI

Para añadir una regla de Config AWS gestionada

El siguiente comando proporciona JSON código para añadir una regla de Config AWS gestionada:

```
aws configservice put-config-rule --config-rule file://
RequiredTagsForEC2Instances.json
```

`RequiredTagsForEC2Instances.json` es un JSON archivo que contiene la configuración de la regla:

```

{
  "ConfigRuleName": "RequiredTagsForEC2Instances",
  "Description": "Checks whether the CostCenter and Owner tags are applied to EC2 instances.",
  "Scope": {
    "ComplianceResourceTypes": [
      "AWS::EC2::Instance"
    ]
  },
  "Source": {
    "Owner": "AWS",
    "SourceIdentifier": "REQUIRED_TAGS"
  }
}

```

```
},  
  "InputParameters": "{\"tag1Key\": \"CostCenter\", \"tag2Key\": \"Owner\"}"  
}
```

Para el `ComplianceResourceTypes` atributo, este JSON código limita el alcance a los recursos de este `AWS::EC2::Instance` tipo, por lo que AWS Config evaluará solo EC2 las instancias según la regla. Puesto que la regla es una regla administrada, el atributo `Owner` se establece en AWS y el atributo `SourceIdentifier` en el identificador de la regla, `REQUIRED_TAGS`. Para el atributo `InputParameters`, se especifican las claves de etiqueta que requiere la regla, `CostCenter` y `Owner`.

Si el comando se ejecuta correctamente, AWS Config no devuelve ningún resultado. Para verificar la configuración de la regla, ejecute el `describe-config-rules` comando y especifique el nombre de la regla.

### Adición de una regla de configuración administrada por el cliente

El siguiente comando proporciona JSON código para añadir una regla de Config gestionada por el cliente:

```
aws configservice put-config-rule --config-rule file://InstanceTypesAreT2micro.json
```

`InstanceTypesAreT2micro.json` es un JSON archivo que contiene la configuración de la regla:

```
{  
  "ConfigRuleName": "InstanceTypesAreT2micro",  
  "Description": "Evaluates whether EC2 instances are the t2.micro type.",  
  "Scope": {  
    "ComplianceResourceTypes": [  
      "AWS::EC2::Instance"  
    ]  
  },  
  "Source": {  
    "Owner": "CUSTOM_LAMBDA",  
    "SourceIdentifier": "arn:aws:lambda:us-east-1:123456789012:function:InstanceTypeCheck",  
    "SourceDetails": [  
      {  
        "EventSource": "aws.config",  
        "MessageType": "ConfigurationItemChangeNotification"  
      }  
    ]  
  }  
}
```

```
    ]  
  },  
  "InputParameters": "{\\"desiredInstanceType\\":\\"t2.micro\\"}"  
}
```

Para el `ComplianceResourceTypes` atributo, este JSON código limita el alcance a los recursos de este `AWS::EC2::Instance` tipo, por lo que AWS Config evaluará solo EC2 las instancias según la regla. Como esta regla es una regla administrada por el cliente, el `Owner` atributo se establece en la función Lambda y el `SourceIdentifier` atributo se establece en ARN la función AWS Lambda. `CUSTOM_LAMBDA` El objeto `SourceDetails` es obligatorio. Los parámetros que se especifican para el `InputParameters` atributo se pasan a la función AWS Lambda cuando AWS Config la invoca para evaluar los recursos en función de la regla.

Si el comando se ejecuta correctamente, AWS Config no devuelve ningún resultado. Para verificar la configuración de la regla, ejecute el `describe-config-rules` comando y especifique el nombre de la regla.

- Para API obtener más información, consulte [PutConfigRule](#) la Referencia de AWS CLI comandos.

## put-configuration-recorder

En el siguiente ejemplo de código, se muestra cómo usar `put-configuration-recorder`.

### AWS CLI

Ejemplo 1: Para registrar todos los recursos compatibles

El siguiente comando crea una grabadora de configuración que rastrea los cambios en todos los tipos de recursos compatibles, incluidos los tipos de recursos globales:

```
aws configservice put-configuration-recorder \  
  --configuration-recorder name=default,roleARN=arn:aws:iam::123456789012:role/  
config-role \  
  --recording-group allSupported=true,includeGlobalResourceTypes=true
```

Si el comando se ejecuta correctamente, AWS Config no devuelve ningún resultado. Para verificar los ajustes de la grabadora de configuración, ejecute el `describe-configuration-recorders` comando.

Ejemplo 2: Para registrar tipos específicos de recursos

El siguiente comando crea una grabadora de configuración que rastrea los cambios realizados únicamente en los tipos de recursos que se especifican en el JSON archivo de la opción `--recording-group`:

```
aws configservice put-configuration-recorder \  
  --configuration-recorder name=default,roleARN=arn:aws:iam::123456789012:role/  
config-role \  
  --recording-group file://recordingGroup.json
```

`recordingGroup.json` es un JSON archivo que especifica los tipos de recursos que AWS Config registrará:

```
{  
  "allSupported": false,  
  "includeGlobalResourceTypes": false,  
  "resourceTypes": [  
    "AWS::EC2::EIP",  
    "AWS::EC2::Instance",  
    "AWS::EC2::NetworkAcl",  
    "AWS::EC2::SecurityGroup",  
    "AWS::CloudTrail::Trail",  
    "AWS::EC2::Volume",  
    "AWS::EC2::VPC",  
    "AWS::IAM::User",  
    "AWS::IAM::Policy"  
  ]  
}
```

Antes de poder especificar los tipos de recursos para la `resourceTypes` clave, debe establecer las opciones `allSupported` y `includeGlobalResource` Tipos en `false` u omitirlas.

Si el comando se ejecuta correctamente, AWS Config no devuelve ningún resultado. Para verificar los ajustes de la grabadora de configuración, ejecute el `describe-configuration-recorders` comando.

Ejemplo 3: Para seleccionar todos los recursos compatibles, excepto tipos específicos de recursos

El siguiente comando crea una grabadora de configuración que rastrea los cambios en todos los tipos de recursos compatibles actuales y futuros, excepto los tipos de recursos que se especifican en el JSON archivo para la opción `--recording-group`:



```
aws configservice put-configuration-recorder \  
  --configuration-recorder name=default,roleARN=arn:aws:iam::123456789012:role/  
config-role \  
  --recording-group file://recordingGroup.json
```

recordingGroup.json es un JSON archivo que especifica los tipos de recursos que AWS Config registrará:

```
{  
  "allSupported": false,  
  "exclusionByResourceTypes": {  
    "resourceTypes": [  
      "AWS::Redshift::ClusterSnapshot",  
      "AWS::RDS::DBClusterSnapshot",  
      "AWS::CloudFront::StreamingDistribution"  
    ]  
  },  
  "includeGlobalResourceTypes": false,  
  "recordingStrategy": {  
    "useOnly": "EXCLUSION_BY_RESOURCE_TYPES"  
  },  
}
```

Antes de poder especificar los tipos de recursos para excluirlos del registro: 1) Debe establecer las opciones allSupported y includeGlobalResource Tipos en false u omitirlas, y 2) Debe establecer el useOnly campo de RecordingStrategy en EXCLUSION\_RESOURCE\_BY\_.. TYPES

Si el comando se ejecuta correctamente, AWS Config no devuelve ningún resultado. Para verificar los ajustes de la grabadora de configuración, ejecute el describe-configuration-recorders comando.

- Para API obtener más información, consulte [PutConfigurationRecorder](#) la Referencia de AWS CLI comandos.

## put-delivery-channel

En el siguiente ejemplo de código, se muestra cómo usar put-delivery-channel.

### AWS CLI

Para crear un canal de entrega

El siguiente comando proporciona la configuración del canal de entrega en forma de JSON código:

```
aws configservice put-delivery-channel --delivery-channel file://  
deliveryChannel.json
```

El `deliveryChannel.json` archivo especifica los atributos del canal de entrega:

```
{  
  "name": "default",  
  "s3BucketName": "config-bucket-123456789012",  
  "snsTopicARN": "arn:aws:sns:us-east-1:123456789012:config-topic",  
  "configSnapshotDeliveryProperties": {  
    "deliveryFrequency": "Twelve_Hours"  
  }  
}
```

Este ejemplo establece los siguientes atributos:

**name**- El nombre del canal de entrega. De forma predeterminada, AWS Config asigna el nombre `default` a un nuevo canal de entrega. No puede actualizar el nombre del canal de entrega con el comando `put-delivery-channel`. Para conocer los pasos para cambiar el nombre, consulte [Cambiar el nombre del canal de entrega](#). **s3BucketName** - El nombre del bucket de Amazon S3 al que AWS Config entrega las instantáneas de configuración y los archivos del historial de configuración. Si especifica un bucket que pertenece a otra AWS cuenta, ese bucket debe tener políticas que concedan permisos de acceso a Config. AWS Para obtener más información, consulte [Permisos para el bucket de Amazon S3](#).

**snsTopicARN**- El nombre del recurso de Amazon (ARN) del SNS tema de Amazon al que AWS Config envía notificaciones sobre los cambios de configuración. Si eliges un tema de otra cuenta, el tema debe tener políticas que concedan permisos de acceso a Config AWS . Para obtener más información, consulta el [SNS tema Permisos para Amazon](#).

**configSnapshotDeliveryProperties**- Contiene el `deliveryFrequency` atributo, que establece la frecuencia con la que AWS Config entrega instantáneas de la configuración y la frecuencia con la que invoca las evaluaciones de las reglas de Config periódicas.

Si el comando se ejecuta correctamente, AWS Config no devuelve ningún resultado. Para verificar la configuración de su canal de entrega, ejecute el `describe-delivery-channels` comando.

- Para API obtener más información, consulte [PutDeliveryChannel](#) la Referencia de AWS CLI comandos.

## start-config-rules-evaluation

En el siguiente ejemplo de código, se muestra cómo usar `start-config-rules-evaluation`.

AWS CLI

Para ejecutar una evaluación bajo demanda de las reglas de AWS Config

El siguiente comando inicia una evaluación de dos reglas AWS administradas:

```
aws configservice start-config-rules-evaluation --config-rule-names s3-bucket-  
versioning-enabled cloudtrail-enabled
```

- Para API obtener más información, consulte [StartConfigRulesEvaluation](#) la Referencia de AWS CLI comandos.

## start-configuration-recorder

En el siguiente ejemplo de código, se muestra cómo usar `start-configuration-recorder`.

AWS CLI

Para iniciar la grabadora de configuración

El siguiente comando inicia la grabadora de configuración predeterminada:

```
aws configservice start-configuration-recorder --configuration-recorder-name default
```

Si el comando se ejecuta correctamente, AWS Config no devuelve ningún resultado. Para comprobar que AWS Config está registrando sus recursos, ejecute el comando `get-status`.

- Para API obtener más información, consulte la Referencia [StartConfigurationRecorder](#) de AWS CLI comandos.

## stop-configuration-recorder

En el siguiente ejemplo de código, se muestra cómo usar `stop-configuration-recorder`.

## AWS CLI

Para detener la grabadora de configuración

El siguiente comando detiene la grabadora de configuración predeterminada:

```
aws configservice stop-configuration-recorder --configuration-recorder-name default
```

Si el comando se ejecuta correctamente, AWS Config no devuelve ningún resultado. Para comprobar que AWS Config no graba sus recursos, ejecute el comando `get-status`.

- Para API obtener más información, consulte la Referencia [StopConfigurationRecorder](#) de AWS CLI comandos.

## subscribe

En el siguiente ejemplo de código, se muestra cómo usar `subscribe`.

## AWS CLI

Para suscribirse a AWS Config

El siguiente comando crea el canal de entrega y el grabador de configuración predeterminados. El comando también especifica el bucket de Amazon S3 y el SNS tema de Amazon al que AWS Config entregará la información de configuración:

```
aws configservice subscribe --s3-bucket config-bucket-123456789012  
--sns-topic arn:aws:sns:us-east-1:123456789012:config-topic --iam-  
role arn:aws:iam::123456789012:role/ConfigRole-A1B2C3D4E5F6
```

Salida:

```
Using existing S3 bucket: config-bucket-123456789012  
Using existing SNS topic: arn:aws:sns:us-east-1:123456789012:config-topic  
Subscribe succeeded:
```

```
Configuration Recorders: [  
  {  
    "recordingGroup": {  
      "allSupported": true,  
      "resourceTypes": [],  
      "includeGlobalResourceTypes": false
```

```
    },
    "roleARN": "arn:aws:iam::123456789012:role/ConfigRole-A1B2C3D4E5F6",
    "name": "default"
  }
]

Delivery Channels: [
  {
    "snsTopicARN": "arn:aws:sns:us-east-1:123456789012:config-topic",
    "name": "default",
    "s3BucketName": "config-bucket-123456789012"
  }
]
```

- Para API obtener más información, consulte [Subscribe](#) in AWS CLI Command Reference.

## Ejemplos de Amazon Connect que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface mediante Amazon Connect.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **create-user**

En el siguiente ejemplo de código, se muestra cómo usar `create-user`.

### AWS CLI

Para crear un usuario

En el siguiente `create-user` ejemplo, se añade un usuario con los atributos especificados a la instancia de Amazon Connect especificada.

```
aws connect create-user \  
  --username Mary \  
  --password Pass@Word1 \  
  --identity-info FirstName=Mary,LastName=Major \  
  --phone-  
config PhoneType=DESK_PHONE,AutoAccept=true,AfterContactWorkTimeLimit=60,DeskPhoneNumber=  
+15555551212 \  
  --security-profile-id 12345678-1111-2222-aaaa-a1b2c3d4f5g7 \  
  --routing-profile-id 87654321-9999-3434-abcd-x1y2z3a1b2c3 \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "UserId": "87654321-2222-1234-1234-111234567891",  
  "UserArn": "arn:aws:connect:us-west-2:123456789012:instance/a1b2c3d4-5678-90ab-  
cdef-EXAMPLE11111/agent/87654321-2222-1234-1234-111234567891"  
}
```

Para obtener más información, consulte [Añadir usuarios](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [CreateUser](#) la Referencia de AWS CLI comandos.

## delete-user

En el siguiente ejemplo de código, se muestra cómo usar `delete-user`.

### AWS CLI

Para eliminar un usuario

En el siguiente `delete-user` ejemplo, se elimina el usuario especificado de la instancia de Amazon Connect especificada.

```
aws connect delete-user \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --user-id 87654321-2222-1234-1234-111234567891
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar usuarios](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [DeleteUser](#) la Referencia de AWS CLI comandos.

## describe-user-hierarchy-group

En el siguiente ejemplo de código, se muestra cómo usar `describe-user-hierarchy-group`.

### AWS CLI

Para mostrar los detalles de un grupo jerárquico

En el siguiente `describe-user-hierarchy-group` ejemplo, se muestran los detalles del grupo jerárquico de Amazon Connect especificado.

```
aws connect describe-user-hierarchy-group \  
  --hierarchy-group-id 12345678-1111-2222-800e-aaabbb555gg \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "HierarchyGroup": {  
    "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",  
    "Arn": "arn:aws:connect:us-west-2:123456789012:instance/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group/12345678-1111-2222-800e-a2b3c4d5f6g7",  
    "Name": "Example Corporation",  
    "LevelId": "1",  
    "HierarchyPath": {  
      "LevelOne": {  
        "Id": "abcdefgh-3333-4444-8af3-201123456789",  
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group/abcdefgh-3333-4444-8af3-201123456789",  
        "Name": "Example Corporation"  
      }  
    }  
  }  
}
```

Para obtener más información, consulte [Configurar jerarquías de agentes](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [DescribeUserHierarchyGroup](#) la Referencia de AWS CLI comandos.

## describe-user-hierarchy-structure

En el siguiente ejemplo de código, se muestra cómo usar `describe-user-hierarchy-structure`.

### AWS CLI

Para mostrar los detalles de una estructura jerárquica

En el siguiente `describe-user-hierarchy-structure` ejemplo, se muestran los detalles de la estructura jerárquica de la instancia de Amazon Connect especificada.

```
aws connect describe-user-hierarchy-group \  
--instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "HierarchyStructure": {  
    "LevelOne": {  
      "Id": "12345678-1111-2222-800e-aaabbb555gg",  
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group-level/1",  
      "Name": "Corporation"  
    },  
    "LevelTwo": {  
      "Id": "87654321-2222-3333-ac99-123456789102",  
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group-level/2",  
      "Name": "Services Division"  
    },  
    "LevelThree": {  
      "Id": "abcdefghijkl-3333-4444-8af3-201123456789",  
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group-level/3",  
      "Name": "EU Site"  
    }  
  }  
}
```



```

    }
  }
}

```

Para obtener más información, consulte [Configurar jerarquías de agentes](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [DescribeUserHierarchyStructure](#) la Referencia de AWS CLI comandos.

## describe-user

En el siguiente ejemplo de código, se muestra cómo usar `describe-user`.

### AWS CLI

Para mostrar los detalles de un usuario

En el siguiente `describe-user` ejemplo, se muestran los detalles del usuario de Amazon Connect especificado.

```

aws connect describe-user \
  --user-id 0c245dc0-0cf5-4e37-800e-2a7481cc8a60
  --instance-id 40c83b68-ea62-414c-97bb-d018e39e158e

```

Salida:

```

{
  "User": {
    "Id": "0c245dc0-0cf5-4e37-800e-2a7481cc8a60",
    "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent/0c245dc0-0cf5-4e37-800e-2a7481cc8a60",
    "Username": "Jane",
    "IdentityInfo": {
      "FirstName": "Jane",
      "LastName": "Doe",
      "Email": "example.com"
    },
    "PhoneConfig": {
      "PhoneType": "SOFT_PHONE",
      "AutoAccept": false,
      "AfterContactWorkTimeLimit": 0,

```

```

        "DeskPhoneNumber": ""
    },
    "DirectoryUserId": "8b444cf6-b368-4f29-ba18-07af27405658",
    "SecurityProfileIds": [
        "b6f85a42-1dc5-443b-b621-de0abf70c9cf"
    ],
    "RoutingProfileId": "0be36ee9-2b5f-4ef4-bcf7-87738e5be0e5",
    "Tags": {}
}
}

```

Para obtener más información, consulte [Administrar usuarios](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [DescribeUser](#) la Referencia de AWS CLI comandos.

## get-contact-attributes

En el siguiente ejemplo de código, se muestra cómo usar `get-contact-attributes`.

### AWS CLI

Para recuperar los atributos de un contacto

En el siguiente `get-contact-attributes` ejemplo, se recuperan los atributos que se establecieron para el contacto de Amazon Connect especificado.

```

aws connect get-contact-attributes \
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --initial-contact-id 12345678-1111-2222-800e-a2b3c4d5f6g7

```

Salida:

```

{
  "Attributes": {
    "greetingPlayed": "true"
  }
}

```

Para obtener más información, consulte [Uso de los atributos de contacto de Amazon Connect](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [GetContactAttributes](#) la Referencia de AWS CLI comandos.

## list-contact-flows

En el siguiente ejemplo de código, se muestra cómo usar `list-contact-flows`.

### AWS CLI

Para enumerar los flujos de contactos de una instancia

En el siguiente `list-contact-flows` ejemplo, se enumeran los flujos de contactos de la instancia de Amazon Connect especificada.

```
aws connect list-contact-flows \  
--instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "ContactFlowSummaryList": [  
    {  
      "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",  
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/contact-flow/12345678-1111-2222-800e-  
a2b3c4d5f6g7",  
      "Name": "Default queue transfer",  
      "ContactFlowType": "QUEUE_TRANSFER"  
    },  
    {  
      "Id": "87654321-2222-3333-ac99-123456789102",  
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/contact-flow/87654321-2222-3333-  
ac99-123456789102",  
      "Name": "Default agent hold",  
      "ContactFlowType": "AGENT_HOLD"  
    },  
    {  
      "Id": "abcdefgh-3333-4444-8af3-201123456789",  
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/contact-flow/  
abcdefgh-3333-4444-8af3-201123456789",  
    }  
  ]  
}
```

```

        "Name": "Default customer hold",
        "ContactFlowType": "CUSTOMER_HOLD"
    },
]
}

```

Para obtener más información, consulte [Creación de flujos de contactos de Amazon Connect](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [ListContactFlows](#) la Referencia de AWS CLI comandos.

## list-hours-of-operations

En el siguiente ejemplo de código, se muestra cómo usar `list-hours-of-operations`.

### AWS CLI

Para enumerar las horas de operación de una instancia

En el siguiente `list-hours-of-operations` ejemplo, se enumeran las horas de operación de la instancia de Amazon Connect especificada.

```

aws connect list-hours-of-operations \
  --instance-id 40c83b68-ea62-414c-97bb-d018e39e158e

```

Salida:

```

{
  "HoursOfOperationSummaryList": [
    {
      "Id": "d69f1f84-7457-4924-8fbe-e64875546259",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/operating-hours/d69f1f84-7457-4924-8fbe-e64875546259",
      "Name": "Basic Hours"
    }
  ]
}

```

Para obtener más información, consulte [Establecer el horario de funcionamiento de una cola](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [ListHoursOfOperations](#) la Referencia de AWS CLI comandos.

## list-phone-numbers

En el siguiente ejemplo de código, se muestra cómo usar `list-phone-numbers`.

### AWS CLI

Para enumerar los números de teléfono de una instancia

En el siguiente `list-phone-numbers` ejemplo, se muestran los números de teléfono de la instancia de Amazon Connect especificada.

```
aws connect list-phone-numbers \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "PhoneNumberSummaryList": [  
    {  
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/phone-number/xyz80zxy-xyz1-80zx-  
zx80-11111EXAMPLE",  
      "PhoneNumber": "+17065551212",  
      "PhoneNumberType": "DID",  
      "PhoneNumberCountryCode": "US"  
    },  
    {  
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/phone-number/ccc0ccc-xyz1-80zx-  
zx80-22222EXAMPLE",  
      "PhoneNumber": "+18555551212",  
      "PhoneNumberType": "TOLL_FREE",  
      "PhoneNumberCountryCode": "US"  
    }  
  ]  
}
```

Para obtener más información, consulte [Configurar números de teléfono para su centro de contacto](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [ListPhoneNumbers](#) la Referencia de AWS CLI comandos.

## list-queues

En el siguiente ejemplo de código, se muestra cómo usar `list-queues`.

### AWS CLI

Para enumerar las colas de una instancia

En el siguiente `list-queues` ejemplo, se muestran las colas de la instancia de Amazon Connect especificada.

```
aws connect list-queues \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "QueueSummaryList": [  
    {  
      "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",  
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/queue/agent/12345678-1111-2222-800e-  
a2b3c4d5f6g7",  
      "QueueType": "AGENT"  
    },  
    {  
      "Id": "87654321-2222-3333-ac99-123456789102",  
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/queue/agent/87654321-2222-3333-  
ac99-123456789102",  
      "QueueType": "AGENT"  
    },  
    {  
      "Id": "abcdefgh-3333-4444-8af3-201123456789",  
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/queue/agent/  
abcdefgh-3333-4444-8af3-201123456789",  
    }  
  ]  
}
```

```

        "QueueType": "AGENT"
    },
    {
        "Id": "hgfedcba-4444-5555-a31f-123456789102",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/queue/hgfedcba-4444-5555-a31f-123456789102",
        "Name": "BasicQueue",
        "QueueType": "STANDARD"
    },
]
}

```

Para obtener más información, consulte [Crear una cola](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [ListQueues](#) la Referencia de AWS CLI comandos.

## list-routing-profiles

En el siguiente ejemplo de código, se muestra cómo usar `list-routing-profiles`.

### AWS CLI

Para enumerar los perfiles de enrutamiento de una instancia

En el siguiente `list-routing-profiles` ejemplo, se enumeran los perfiles de enrutamiento de la instancia de Amazon Connect especificada.

```

aws connect list-routing-profiles \
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

```

Salida:

```

{
  "RoutingProfileSummaryList": [
    {
      "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/routing-profile/12345678-1111-2222-800e-
a2b3c4d5f6g7",
      "Name": "Basic Routing Profile"
    },
  ],
}

```

```
]
}
```

Para obtener más información, consulte [Creación de un perfil de enrutamiento](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [ListRoutingProfiles](#) la Referencia de AWS CLI comandos.

## list-security-profiles

En el siguiente ejemplo de código, se muestra cómo usar `list-security-profiles`.

### AWS CLI

Para enumerar los perfiles de seguridad de una instancia

En el siguiente `list-security-profiles` ejemplo, se enumeran los perfiles de seguridad de la instancia de Amazon Connect especificada.

```
aws connect list-security-profiles \
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{
  "SecurityProfileSummaryList": [
    {
      "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/security-profile/12345678-1111-2222-800e-
a2b3c4d5f6g7",
      "Name": "CallCenterManager"
    },
    {
      "Id": "87654321-2222-3333-ac99-123456789102",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/security-profile/87654321-2222-3333-
ac99-123456789102",
      "Name": "QualityAnalyst"
    },
  ],
}
```



```

    {
      "Id": "abcdefgh-3333-4444-8af3-201123456789",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/security-profile/
abcdefgh-3333-4444-8af3-201123456789",
      "Name": "Agent"
    },
    {
      "Id": "12345678-1111-2222-800e-x2y3c4d5fzzzz",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/security-profile/12345678-1111-2222-800e-
x2y3c4d5fzzzz",
      "Name": "Admin"
    }
  ]
}

```

Para obtener más información, consulte [Asignar permisos: perfiles de seguridad](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [ListSecurityProfiles](#) la Referencia de AWS CLI comandos.

## list-user-hierarchy-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-user-hierarchy-groups`.

### AWS CLI

Para enumerar los grupos jerárquicos de usuarios de una instancia

En el siguiente `list-user-hierarchy-groups` ejemplo, se enumeran los grupos jerárquicos de usuarios de la instancia de Amazon Connect especificada.

```

aws connect list-user-hierarchy-groups \
  --instance-id 40c83b68-ea62-414c-97bb-d018e39e158e

```

Salida:

```

{
  "UserHierarchyGroupSummaryList": [
    {

```

```

        "Id": "0e2f6d1d-b3ca-494b-8dbc-ba81d9f8182a",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent-group/0e2f6d1d-b3ca-494b-8dbc-ba81d9f8182a",
        "Name": "Example Corporation"
    },
]
}

```

Para obtener más información, consulte [Configurar jerarquías de agentes](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [ListUserHierarchyGroups](#) la Referencia de AWS CLI comandos.

## list-users

En el siguiente ejemplo de código, se muestra cómo usar `list-users`.

### AWS CLI

Para enumerar los grupos jerárquicos de usuarios de una instancia

En el siguiente `list-users` ejemplo, se enumeran los usuarios de la instancia de Amazon Connect especificada.

```

aws connect list-users \
  --instance-id 40c83b68-ea62-414c-97bb-d018e39e158e

```

Salida:

```

{
  "UserSummaryList": [
    {
      "Id": "0c245dc0-0cf5-4e37-800e-2a7481cc8a60",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent/0c245dc0-0cf5-4e37-800e-2a7481cc8a60",
      "Username": "Jane"
    },
    {
      "Id": "46f0c67c-3fc7-4806-ac99-403798788c14",
      "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent/46f0c67c-3fc7-4806-ac99-403798788c14",

```

```

    "Username": "Paulo"
  },
  {
    "Id": "55a83578-95e1-4710-8af3-2b7afe310e48",
    "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent/55a83578-95e1-4710-8af3-2b7afe310e48",
    "Username": "JohnD"
  },
  {
    "Id": "703e27b5-c9f0-4f1f-a239-64ccbb160125",
    "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent/703e27b5-c9f0-4f1f-a239-64ccbb160125",
    "Username": "JohnS"
  }
]
}

```

Para obtener más información, consulte [Añadir usuarios](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [ListUsers](#) la Referencia de AWS CLI comandos.

## update-contact-attributes

En el siguiente ejemplo de código, se muestra cómo usar `update-contact-attributes`.

### AWS CLI

Para actualizar el atributo de un contacto

En el siguiente `update-contact-attributes` ejemplo, se actualiza el `greetingPlayed` atributo del usuario de Amazon Connect especificado.

```

aws connect update-contact-attributes \
  --initial-contact-id 11111111-2222-3333-4444-12345678910 \
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --attributes greetingPlayed=false

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de los atributos de contacto de Amazon Connect](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [UpdateContactAttributes](#) la Referencia de AWS CLI comandos.

## update-user-hierarchy

En el siguiente ejemplo de código, se muestra cómo usar `update-user-hierarchy`.

### AWS CLI

Para actualizar la jerarquía de un usuario

En el siguiente `update-user-hierarchy` ejemplo, se actualiza la jerarquía de agentes del usuario de Amazon Connect especificado.

```
aws connect update-user-hierarchy \  
  --hierarchy-group-id 12345678-a1b2-c3d4-e5f6-123456789abc \  
  --user-id 87654321-2222-1234-1234-111234567891 \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configurar los ajustes del agente](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [UpdateUserHierarchy](#) la Referencia de AWS CLI comandos.

## update-user-identity-info

En el siguiente ejemplo de código, se muestra cómo usar `update-user-identity-info`.

### AWS CLI

Para actualizar la información de identidad de un usuario

En el siguiente `update-user-identity-info` ejemplo, se actualiza la información de identidad del usuario de Amazon Connect especificado.

```
aws connect update-user-identity-info \  
  --identity-info FirstName=Mary,LastName=Major,Email=marym@example.com \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

```
--user-id 87654321-2222-1234-1234-111234567891 \  
--instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configurar los ajustes del agente](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [UpdateUserIdentityInfo](#) la Referencia de AWS CLI comandos.

## update-user-phone-config

En el siguiente ejemplo de código, se muestra cómo usar `update-user-phone-config`.

AWS CLI

Para actualizar la configuración del teléfono de un usuario

En el siguiente `update-user-phone-config` ejemplo, se actualiza la configuración del teléfono del usuario especificado.

```
aws connect update-user-phone-config \  
  --phone-  
config PhoneType=SOFT_PHONE,AutoAccept=false,AfterContactWorkTimeLimit=60,DeskPhoneNumber=  
+18005551212 \  
  --user-id 12345678-4444-3333-2222-111122223333 \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configurar los ajustes del agente](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [UpdateUserPhoneConfig](#) la Referencia de AWS CLI comandos.

## update-user-routing-profile

En el siguiente ejemplo de código, se muestra cómo usar `update-user-routing-profile`.

## AWS CLI

Para actualizar el perfil de enrutamiento de un usuario

El siguiente `update-user-routing-profile` ejemplo actualiza el perfil de enrutamiento del usuario de Amazon Connect especificado.

```
aws connect update-user-routing-profile \  
  --routing-profile-id 12345678-1111-3333-2222-4444EXAMPLE \  
  --user-id 87654321-2222-1234-1234-111234567891 \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configurar los ajustes del agente](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [UpdateUserRoutingProfile](#) la Referencia de AWS CLI comandos.

## update-user-security-profiles

En el siguiente ejemplo de código, se muestra cómo usar `update-user-security-profiles`.

## AWS CLI

Para actualizar los perfiles de seguridad de un usuario

El siguiente `update-user-security-profiles` ejemplo actualiza el perfil de seguridad del usuario de Amazon Connect especificado.

```
aws connect update-user-security-profiles \  
  --security-profile-ids 12345678-1234-1234-1234-1234567892111 \  
  --user-id 87654321-2222-1234-1234-111234567891 \  
  --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Asignar permisos: perfiles de seguridad](#) en la Guía del administrador de Amazon Connect.

- Para API obtener más información, consulte [UpdateUserSecurityProfiles](#) la Referencia de AWS CLI comandos.

## AWS Cost and Usage Report ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Cost and Usage Report.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **delete-report-definition**

En el siguiente ejemplo de código, se muestra cómo usar `delete-report-definition`.

AWS CLI

Para eliminar un informe de AWS costos y uso

En este ejemplo, se elimina un informe de AWS costo y uso.

Comando:

```
aws cur --region us-east-1 delete-report-definition --report-name "ExampleReport"
```

- Para API obtener más información, consulte [DeleteReportDefinition](#) la Referencia de AWS CLI comandos.

#### **describe-report-definitions**

En el siguiente ejemplo de código, se muestra cómo usar `describe-report-definitions`.

## AWS CLI

Para recuperar una lista de informes de AWS costos y uso

En este ejemplo, se describe una lista de informes de AWS costo y uso propiedad de una cuenta.

Comando:

```
aws cur --region us-east-1 describe-report-definitions --max-items 5
```

Salida:

```
{
  "ReportDefinitions": [
    {
      "ReportName": "ExampleReport",
      "Compression": "ZIP",
      "S3Region": "us-east-1",
      "Format": "textORcsv",
      "S3Prefix": "exampleprefix",
      "S3Bucket": "example-s3-bucket",
      "TimeUnit": "DAILY",
      "AdditionalArtifacts": [
        "REDSHIFT",
        "QUICKSIGHT"
      ],
      "AdditionalSchemaElements": [
        "RESOURCES"
      ]
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeReportDefinitions](#) la Referencia de AWS CLI comandos.

## put-report-definition

En el siguiente ejemplo de código, se muestra cómo usar `put-report-definition`.

## AWS CLI

Para crear un informe de AWS costo y uso



En el siguiente `put-report-definition` ejemplo, se crea un informe diario de AWS costes y uso que puede cargar en Amazon Redshift o Amazon. QuickSight

```
aws cur put-report-definition --report-definition file://report-definition.json
```

Contenidos de `report-definition.json`:

```
{
  "ReportName": "ExampleReport",
  "TimeUnit": "DAILY",
  "Format": "textORcsv",
  "Compression": "ZIP",
  "AdditionalSchemaElements": [
    "RESOURCES"
  ],
  "S3Bucket": "example-s3-bucket",
  "S3Prefix": "exampleprefix",
  "S3Region": "us-east-1",
  "AdditionalArtifacts": [
    "REDSHIFT",
    "QUICKSIGHT"
  ]
}
```

- Para API obtener más información, consulte [PutReportDefinition](#) la Referencia de AWS CLI comandos.

## Ejemplos del servicio Cost Explorer utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el AWS Command Line Interface servicio Cost Explorer.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### get-cost-and-usage

En el siguiente ejemplo de código, se muestra cómo usar `get-cost-and-usage`.

#### AWS CLI

Para recuperar el uso de una cuenta en S3 durante el mes de septiembre de 2017

En el siguiente `get-cost-and-usage` ejemplo, se recupera el uso de S3 de una cuenta durante el mes de septiembre de 2017.

```
aws ce get-cost-and-usage \  
  --time-period Start=2017-09-01,End=2017-10-01 \  
  --granularity MONTHLY \  
  --metrics "BlendedCost" "UnblendedCost" "UsageQuantity" \  
  --group-by Type=DIMENSION,Key=SERVICE Type=TAG,Key=Environment \  
  --filter file://filters.json
```

Contenidos de `filters.json`:

```
{  
  "Dimensions": {  
    "Key": "SERVICE",  
    "Values": [  
      "Amazon Simple Storage Service"  
    ]  
  }  
}
```

Salida:

```
{  
  "GroupDefinitions": [  
    {  
      "Type": "DIMENSION",  
      "Key": "SERVICE"  
    },  
    {
```

```
        "Type": "TAG",
        "Key": "Environment"
    }
],
"ResultsByTime": [
    {
        "Estimated": false,
        "TimePeriod": {
            "Start": "2017-09-01",
            "End": "2017-10-01"
        },
        "Total": {},
        "Groups": [
            {
                "Keys": [
                    "Amazon Simple Storage Service",
                    "Environment$"
                ],
                "Metrics": {
                    "BlendedCost": {
                        "Amount": "40.3527508453",
                        "Unit": "USD"
                    },
                    "UnblendedCost": {
                        "Amount": "40.3543773134",
                        "Unit": "USD"
                    },
                    "UsageQuantity": {
                        "Amount": "9312771.098461578",
                        "Unit": "N/A"
                    }
                }
            }
        ],
        {
            "Keys": [
                "Amazon Simple Storage Service",
                "Environment$Dev"
            ],
            "Metrics": {
                "BlendedCost": {
                    "Amount": "0.2682364644",
                    "Unit": "USD"
                },
                "UnblendedCost": {
```

```

        "Amount": "0.2682364644",
        "Unit": "USD"
      },
      "UsageQuantity": {
        "Amount": "22403.4395271182",
        "Unit": "N/A"
      }
    }
  ]
}

```

- Para API obtener más información, consulte [GetCostAndUsage](#) la Referencia de AWS CLI comandos.

## get-dimension-values

En el siguiente ejemplo de código, se muestra cómo usar `get-dimension-values`.

### AWS CLI

Para recuperar las etiquetas de la dimensión `SERVICE`, con el valor «Elastic»

En este ejemplo, se recuperan las etiquetas de la dimensión `SERVICE`, con un valor de «Elastic», del 1 de enero al 18 de mayo de 2017.

Comando:

```
aws ce get-dimension-values --search-string Elastic --time-period Start=2017-01-01,End=2017-05-18 --dimension SERVICE
```

Salida:

```

{
  "TotalSize": 6,
  "DimensionValues": [
    {
      "Attributes": {},
      "Value": "Amazon ElastiCache"
    },
  ],
}

```

```

    {
      "Attributes": {},
      "Value": "EC2 - Other"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elastic Compute Cloud - Compute"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elastic Load Balancing"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elastic MapReduce"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elasticsearch Service"
    }
  ],
  "ReturnSize": 6
}

```

- Para API obtener más información, consulte [GetDimensionValues](#) la Referencia de AWS CLI comandos.

## get-reservation-coverage

En el siguiente ejemplo de código, se muestra cómo usar `get-reservation-coverage`.

### AWS CLI

Para recuperar la cobertura de reservas para las instancias EC2 t2.nano en la región us-east-1

En este ejemplo, se recupera la cobertura de reservas para las instancias EC2 t2.nano en la región us-east-1 de julio a septiembre de 2017.

Comando:

```
aws ce get-reservation-coverage --time-period Start=2017-07-01,End=2017-10-01 --
group-by Type=Dimension,Key=REGION --filter file://filters.json
```

filters.json:

```
{
  "And": [
    {
      "Dimensions": {
        "Key": "INSTANCE_TYPE",
        "Values": [
          "t2.nano"
        ]
      },
      "Dimensions": {
        "Key": "REGION",
        "Values": [
          "us-east-1"
        ]
      }
    }
  ]
}
```

Salida:

```
{
  "TotalSize": 6,
  "DimensionValues": [
    {
      "Attributes": {},
      "Value": "Amazon ElastiCache"
    },
    {
      "Attributes": {},
      "Value": "EC2 - Other"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elastic Compute Cloud - Compute"
    },
    {
      "Attributes": {},
      "Value": "Amazon Elastic Load Balancing"
    },
    {
```

```

        "Attributes": {},
        "Value": "Amazon Elastic MapReduce"
    },
    {
        "Attributes": {},
        "Value": "Amazon Elasticsearch Service"
    }
],
"ReturnSize": 6
}

```

- Para API obtener más información, consulte la Referencia de [GetReservationCoverage](#) comandos AWS CLI .

## get-reservation-purchase-recommendation

En el siguiente ejemplo de código, se muestra cómo usar `get-reservation-purchase-recommendation`.

### AWS CLI

Para obtener las recomendaciones de reserva para un pago anticipado parcial EC2 RIs con un plazo de tres años

En el siguiente `get-reservation-purchase-recommendation` ejemplo, se recuperan las recomendaciones para las EC2 instancias de Partial Upfront con un plazo de tres años, en función de los últimos 60 días de uso. EC2

```

aws ce get-reservation-purchase-recommendation \
  --service "Amazon Redshift" \
  --lookback-period-in-days SIXTY_DAYS \
  --term-in-years THREE_YEARS \
  --payment-option PARTIAL_UPFRONT

```

Salida:

```

{
  "Recommendations": [],
  "Metadata": {
    "GenerationTimestamp": "2018-08-08T15:20:57Z",
    "RecommendationId": "00d59dde-a1ad-473f-8ff2-iexample3330b"
  }
}

```

```
}  
}
```

- Para API obtener más información, consulte la Referencia [GetReservationPurchaseRecommendation](#) de AWS CLI comandos.

## get-reservation-utilization

En el siguiente ejemplo de código, se muestra cómo usar `get-reservation-utilization`.

### AWS CLI

Para recuperar la reserva, utilice su cuenta

En el siguiente `get-reservation-utilization` ejemplo, se recupera la utilización de la RI en todos los tipos de instancias `t2.nano` entre el 1 de marzo de 2018 y el 1 de agosto de 2018 de la cuenta.

```
aws ce get-reservation-utilization \  
  --time-period Start=2018-03-01,End=2018-08-01 \  
  --filter file://filters.json
```

Contenidos de `filters.json`:

```
{  
  "Dimensions": {  
    "Key": "INSTANCE_TYPE",  
    "Values": [  
      "t2.nano"  
    ]  
  }  
}
```

Salida:

```
{  
  "Total": {  
    "TotalAmortizedFee": "0",  
    "UtilizationPercentage": "0",  
    "PurchasedHours": "0",
```



```

    "NetRISavings": "0",
    "TotalActualHours": "0",
    "AmortizedRecurringFee": "0",
    "UnusedHours": "0",
    "TotalPotentialRISavings": "0",
    "OnDemandCostOfRIHoursUsed": "0",
    "AmortizedUpfrontFee": "0"
  },
  "UtilizationsByTime": []
}

```

- Para API obtener más información, consulte la Referencia de comandos.

[GetReservationUtilization](#) AWS CLI

## get-tags

En el siguiente ejemplo de código, se muestra cómo usar `get-tags`.

### AWS CLI

Para recuperar las claves y los valores de una etiqueta de asignación de costes

En este ejemplo, se recuperan todas las etiquetas de asignación de costes con la clave «Proyecto» y un valor que contiene «secretProject».

Comando:

```
aws ce get-tags --search-string secretProject --time-period Start=2017-01-01,End=2017-05-18 --tag-key Project
```

Salida:

```

{
  "ReturnSize": 2,
  "Tags": [
    "secretProject1",
    "secretProject2"
  ],
  "TotalSize": 2
}

```

- Para API obtener más información, consulte [GetTags](#) la Referencia de AWS CLI comandos.

## Ejemplos de Firehose usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Firehose.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde encontrarás instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **list-delivery-streams**

En el siguiente ejemplo de código, se muestra cómo usar `list-delivery-streams`.

AWS CLI

Para enumerar los flujos de entrega disponibles

En el siguiente `list-delivery-streams` ejemplo, se enumeran los flujos de entrega disponibles en su AWS cuenta.

```
aws firehose list-delivery-streams
```

Salida:

```
{
  "DeliveryStreamNames": [
    "my-stream"
  ],
  "HasMoreDeliveryStreams": false
}
```

Para obtener más información, consulte la sección sobre [creación de una secuencia de entrega de Amazon Kinesis Data Firehose](#) en la guía para desarrolladores de Amazon Kinesis Data Firehose.

- Para API obtener más información, consulte [ListDeliveryStreams](#) la Referencia de AWS CLI comandos.

## put-record-batch

En el siguiente ejemplo de código, se muestra cómo usar `put-record-batch`.

### AWS CLI

Escritura de varios registros en una secuencia

En el siguiente ejemplo `put-record-batch`, se escriben tres registros en una secuencia. Los datos se codifican en formato Base64.

```
aws firehose put-record-batch \
  --delivery-stream-name my-stream \
  --records file://records.json
```

Contenidos de `myfile.json`:

```
[
  {"Data": "Rmlyc3QgdGhpbmci="},
  {"Data": "U2Vjb25kIHRoaW5n"},
  {"Data": "VGhpcmQgdGhpbmci="}
]
```

Salida:

```
{
  "FailedPutCount": 0,
  "Encrypted": false,
  "RequestResponses": [
    {
      "RecordId": "9D20J6t2EqCTZTXwGzeSv/EVHxRoRCw89xd+o3+sXg8DhY0aWKPSmZy/
CGlRVEys1u1xbeKh6VofEYKkoeiDrcjrxhQp9iF7sUW7pujiMEQ5LzlrzCkGosxQn
+3boDnURDEaD42V7Giixp0yLJkYZcae1i7HzlCEoy9LJhMr8EjDSi40m/9Vc2uhwwuAtGE0XKpxJ2WD7ZRWtAnY1KAnv"
    },
    {
```

```

    "RecordId": "jFirejqxCLlK5xjH/UNm1MVcjkTEN76I7916X9PaZ
+PVa0SXDFU1WG0qEZhxq2js7xcZ552eoeDxsuTU1MSq9nZTbVfb6cQTIXnm/GsuF37Uhg67GkmR5z9016XKJ
+/+pD1oFv7Hh9a3oUS6wYm3DcNRLTHHAimANp1PhkQvWpvLRfzbuCUkBphR2QVzhP90iHLbzGwy8/
DfH8sqWEUYASNJKS8GXP5s"
  },
  {
    "RecordId":
      "oy0amQ40o5Y2YV4vxzufdcM00w6n3EPi3tpPJGoYVNKH4APPVqNcbUgefo1stEFRg4hTLrf2k6eliHu/9+YJ5R3iie
DTBt3qBlmTj7Xq8SKVb01S7YvMTpWkMKA86f8JfmT8BMKoMb4XZS/s0kQLe+qh0sYKXW1"
  }
]
}

```

Para obtener más información, consulte [Envío de una secuencia de entrega de Amazon Kinesis Data Firehose](#) en la Guía para desarrolladores de Amazon Kinesis Data Firehose.

- Para API obtener más información, consulte [PutRecordBatch](#) la Referencia de AWS CLI comandos.

## put-record

En el siguiente ejemplo de código, se muestra cómo usar `put-record`.

### AWS CLI

Para escribir un registro en una transmisión

El siguiente `put-record` ejemplo escribe datos en una transmisión. Los datos se codifican en formato Base64.

```

aws firehose put-record \
  --delivery-stream-name my-stream \
  --record '{"Data": "SGVsbG8gd29ybGQ="}'

```

Salida:

```

{
  "RecordId": "RjB5K/nnoGFHqwTsZ1Nd/
TTqvjE8V5dsyXZTQn2JXrdpMT0wssyEb6nfC8fwf1whhwnItt4mvrn+gsqeK5jB7QjuLg283+Ps4Sz/
j1Xujv31iDhnPdaLw4B0yM9Amv7PcCuB2079RuM0NhoakbyUymLwY8yt20G8X2420wu1j1Fafhci4erAt7QhDEvpwuK8
  "Encrypted": false
}

```

Para obtener más información, consulte [Envío de una secuencia de entrega de Amazon Kinesis Data Firehose](#) en la Guía para desarrolladores de Amazon Kinesis Data Firehose.

- Para API obtener más información, consulte [PutRecord](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon Data Lifecycle Manager que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Amazon Data Lifecycle Manager.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-default-role**

En el siguiente ejemplo de código, se muestra cómo usar `create-default-role`.

AWS CLI

Para crear el IAM rol necesario para Amazon DLM

En el siguiente `d1m create-default-role` ejemplo, se crea el rol AWS `DataLifecycleManagerDefaultRole` predeterminado para gestionar las instantáneas.

```
aws d1m create-default-role \  
  --resource-type snapshot
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Funciones de servicio predeterminadas para Amazon Data Lifecycle Manager](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte [CreateDefaultRole](#) la Referencia de AWS CLI comandos.

## create-lifecycle-policy

En el siguiente ejemplo de código, se muestra cómo usar `create-lifecycle-policy`.

### AWS CLI

Para crear una política de ciclo de vida

En el siguiente `create-lifecycle-policy` ejemplo, se crea una política de ciclo de vida que crea una instantánea diaria de los volúmenes a la hora especificada. Las etiquetas especificadas se agregan a las instantáneas y las etiquetas también se copian del volumen y se agregan a las instantáneas. Si al crear una nueva instantánea se supera el recuento máximo especificado, se elimina la instantánea más antigua.

```
aws dlm create-lifecycle-policy \  
  --description "My first policy" \  
  --state ENABLED \  
  --execution-role-arn arn:aws:iam::12345678910:role/  
AWSDataLifecycleManagerDefaultRole \  
  --policy-details file://policyDetails.json
```

Contenidos de `policyDetails.json`:

```
{  
  "ResourceTypes": [  
    "VOLUME"  
  ],  
  "TargetTags": [  
    {  
      "Key": "costCenter",  
      "Value": "115"  
    }  
  ],  
  "Schedules": [  
    {  
      "Name": "DailySnapshots",  
      "CopyTags": true,  
      "TagsToAdd": [  

```

```
    {
      "Key": "type",
      "Value": "myDailySnapshot"
    }
  ],
  "CreateRule": {
    "Interval": 24,
    "IntervalUnit": "HOURS",
    "Times": [
      "03:00"
    ]
  },
  "RetainRule": {
    "Count": 5
  }
}
]
```

Salida:

```
{
  "PolicyId": "policy-0123456789abcdef0"
}
```

- Para API obtener más información, consulte [CreateLifecyclePolicy](#) la Referencia de AWS CLI comandos.

## delete-lifecycle-policy

En el siguiente ejemplo de código, se muestra cómo usar delete-lifecycle-policy.

AWS CLI

Para eliminar una política de ciclo de vida

En el siguiente ejemplo, se elimina la política de ciclo de vida especificada. :

```
aws dlm delete-lifecycle-policy --policy-id policy-0123456789abcdef0
```

- Para API obtener más información, consulte [DeleteLifecyclePolicy](#) la Referencia de AWS CLI comandos.

## get-lifecycle-policies

En el siguiente ejemplo de código, se muestra cómo usar `get-lifecycle-policies`.

### AWS CLI

Para obtener un resumen de sus políticas de ciclo de vida

En el siguiente `get-lifecycle-policies` ejemplo, se enumeran todas las políticas del ciclo de vida.

```
aws dlm get-lifecycle-policies
```

Salida:

```
{
  "Policies": [
    {
      "PolicyId": "policy-0123456789abcdef0",
      "Description": "My first policy",
      "State": "ENABLED"
    }
  ]
}
```

- Para API obtener más información, consulte [GetLifecyclePolicies](#) la Referencia de AWS CLI comandos.

## get-lifecycle-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-lifecycle-policy`.

### AWS CLI

Para describir una política de ciclo de vida

En el siguiente `get-lifecycle-policy` ejemplo, se muestran los detalles de la política de ciclo de vida especificada.

```
aws dlm get-lifecycle-policy \
  --policy-id policy-0123456789abcdef0
```



**Salida:**

```
{
  "Policy": {
    "PolicyId": "policy-0123456789abcdef0",
    "Description": "My policy",
    "State": "ENABLED",
    "ExecutionRoleArn": "arn:aws:iam::123456789012:role/
AWSDataLifecycleManagerDefaultRole",
    "DateCreated": "2019-08-08T17:45:42Z",
    "DateModified": "2019-08-08T17:45:42Z",
    "PolicyDetails": {
      "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
      "ResourceTypes": [
        "VOLUME"
      ],
      "TargetTags": [
        {
          "Key": "costCenter",
          "Value": "115"
        }
      ],
      "Schedules": [
        {
          "Name": "DailySnapshots",
          "CopyTags": true,
          "TagsToAdd": [
            {
              "Key": "type",
              "Value": "myDailySnapshot"
            }
          ],
          "CreateRule": {
            "Interval": 24,
            "IntervalUnit": "HOURS",
            "Times": [
              "03:00"
            ]
          },
          "RetainRule": {
            "Count": 5
          }
        }
      ]
    }
  }
}
```

```
    }  
  }  
}
```

- Para API obtener más información, consulte [GetLifecyclePolicy](#) la Referencia de AWS CLI comandos.

## update-lifecycle-policy

En el siguiente ejemplo de código, se muestra cómo usar `update-lifecycle-policy`.

### AWS CLI

Ejemplo 1: Para habilitar una política de ciclo de vida

El siguiente `update-lifecycle-policy` ejemplo habilita la política de ciclo de vida especificada.

```
aws dlm update-lifecycle-policy \  
  --policy-id policy-0123456789abcdef0 \  
  --state ENABLED
```

Ejemplo 2: Para deshabilitar una política de ciclo de vida

En el siguiente `update-lifecycle-policy` ejemplo, se inhabilita la política de ciclo de vida especificada.

```
aws dlm update-lifecycle-policy \  
  --policy-id policy-0123456789abcdef0 \  
  --state DISABLED
```

Ejemplo 3: Para actualizar los detalles de la política de ciclo de vida

En el siguiente `update-lifecycle-policy` ejemplo, se actualizan las etiquetas de destino de la política de ciclo de vida especificada.

```
aws dlm update-lifecycle-policy \  
  --policy-id policy-0123456789abcdef0 \  
  --policy-details file://policyDetails.json
```

Contenido de `policyDetails.json`. El comando no modifica otros detalles a los que no se hace referencia en este archivo.

```
{
  "TargetTags": [
    {
      "Key": "costCenter",
      "Value": "120"
    },
    {
      "Key": "project",
      "Value": "lima"
    }
  ]
}
```

- Para API obtener más información, consulte [UpdateLifecyclePolicy](#) la Referencia de AWS CLI comandos.

## AWS Data Pipeline ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Data Pipeline.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **activate-pipeline**

En el siguiente ejemplo de código, se muestra cómo usar `activate-pipeline`.

## AWS CLI

Para activar una canalización

En este ejemplo, se activa la canalización especificada:

```
aws datapipeline activate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE
```

Para activar la canalización en una fecha y hora específicas, usa el siguiente comando:

```
aws datapipeline activate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE --start-timestamp 2015-04-07T00:00:00Z
```

- Para API obtener más información, consulte [ActivatePipeline](#) la Referencia de AWS CLI comandos.

## add-tags

En el siguiente ejemplo de código, se muestra cómo usar add-tags.

## AWS CLI

Para añadir una etiqueta a una canalización

En este ejemplo, se añade la etiqueta especificada a la canalización especificada:

```
aws datapipeline add-tags --pipeline-id df-00627471S0VYZEXAMPLE --tags key=environment,value=production key=owner,value=sales
```

Para ver las etiquetas, utilice el comando describe-pipelines. Por ejemplo, las etiquetas agregadas en el comando example aparecen de la siguiente manera en el resultado de describe-pipelines:

```
{
  ...
  "tags": [
    {
      "value": "production",
      "key": "environment"
    },
    {
```

```
        "value": "sales",
        "key": "owner"
      }
    ]
    ...
  }
```

- Para obtener API más información, consulte la Referencia de comandos [AddTags](#).AWS CLI

## create-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `create-pipeline`.

### AWS CLI

Para crear una canalización

En este ejemplo se crea una canalización:

```
aws datapipeline create-pipeline --name my-pipeline --unique-id my-pipeline-token
```

A continuación, se muestra un ejemplo de la salida:

```
{
  "pipelineId": "df-00627471S0VYZEXAMPLE"
}
```

- Para API obtener más información, consulte [CreatePipeline](#) la Referencia de AWS CLI comandos.

## deactivate-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `deactivate-pipeline`.

### AWS CLI

Para desactivar una canalización

En este ejemplo se desactiva la canalización especificada:

```
aws datapipeline deactivate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE
```

Para desactivar la canalización solo después de que finalicen todas las actividades en ejecución, usa el siguiente comando:

```
aws datapipeline deactivate-pipeline --pipeline-id df-00627471S0VYZEXAMPLE --no-cancel-active
```

- Para API obtener más información, consulte [DeactivatePipeline](#) la Referencia de AWS CLI comandos.

## delete-pipeline

En el siguiente ejemplo de código, se muestra cómo usar delete-pipeline.

### AWS CLI

Para eliminar una canalización

En este ejemplo, se elimina la canalización especificada:

```
aws datapipeline delete-pipeline --pipeline-id df-00627471S0VYZEXAMPLE
```

- Para API obtener más información, consulte [DeletePipeline](#) la Referencia de AWS CLI comandos.

## describe-pipelines

En el siguiente ejemplo de código, se muestra cómo usar describe-pipelines.

### AWS CLI

Para describir sus canalizaciones

En este ejemplo se describe la canalización especificada:

```
aws datapipeline describe-pipelines --pipeline-ids df-00627471S0VYZEXAMPLE
```

A continuación, se muestra un ejemplo de la salida:

```
{
  "pipelineDescriptionList": [
    {
```

```
    "fields": [  
      {  
        "stringValue": "PENDING",  
        "key": "@pipelineState"  
      },  
      {  
        "stringValue": "my-pipeline",  
        "key": "name"  
      },  
      {  
        "stringValue": "2015-04-07T16:05:58",  
        "key": "@creationTime"  
      },  
      {  
        "stringValue": "df-00627471SOVYZEXAMPLE",  
        "key": "@id"  
      },  
      {  
        "stringValue": "123456789012",  
        "key": "pipelineCreator"  
      },  
      {  
        "stringValue": "PIPELINE",  
        "key": "@sphere"  
      },  
      {  
        "stringValue": "123456789012",  
        "key": "@userId"  
      },  
      {  
        "stringValue": "123456789012",  
        "key": "@accountId"  
      },  
      {  
        "stringValue": "my-pipeline-token",  
        "key": "uniqueId"  
      }  
    ],  
    "pipelineId": "df-00627471SOVYZEXAMPLE",  
    "name": "my-pipeline",  
    "tags": []  
  }  
]
```

```
}
```

- Para API obtener más información, consulte [DescribePipelines](#) la Referencia de AWS CLI comandos.

## get-pipeline-definition

En el siguiente ejemplo de código, se muestra cómo usar `get-pipeline-definition`.

### AWS CLI

Para obtener una definición de canalización

En este ejemplo se obtiene la definición de canalización de la canalización especificada:

```
aws datapipeline get-pipeline-definition --pipeline-id df-00627471S0VYZEXAMPLE
```

A continuación, se muestra un ejemplo de la salida:

```
{
  "parameters": [
    {
      "type": "AWS::S3::ObjectKey",
      "id": "myS3OutputLoc",
      "description": "S3 output folder"
    },
    {
      "default": "s3://us-east-1.elasticmapreduce.samples/pig-apache-logs/data",
      "type": "AWS::S3::ObjectKey",
      "id": "myS3InputLoc",
      "description": "S3 input folder"
    },
    {
      "default": "grep -rc \"GET\" ${INPUT1_STAGING_DIR}/* >
${OUTPUT1_STAGING_DIR}/output.txt",
      "type": "String",
      "id": "myShellCmd",
      "description": "Shell command to run"
    }
  ],
  "objects": [
    {
```



```

    "type": "Ec2Resource",
    "terminateAfter": "20 Minutes",
    "instanceType": "t1.micro",
    "id": "EC2ResourceObj",
    "name": "EC2ResourceObj"
  },
  {
    "name": "Default",
    "failureAndRerunMode": "CASCADE",
    "resourceRole": "DataPipelineDefaultResourceRole",
    "schedule": {
      "ref": "DefaultSchedule"
    },
    "role": "DataPipelineDefaultRole",
    "scheduleType": "cron",
    "id": "Default"
  },
  {
    "directoryPath": "#{myS3OutputLoc}/#{format(@scheduledStartTime, 'YYYY-MM-dd-HH-mm-ss')}",
    "type": "S3DataNode",
    "id": "S3OutputLocation",
    "name": "S3OutputLocation"
  },
  {
    "directoryPath": "#{myS3InputLoc}",
    "type": "S3DataNode",
    "id": "S3InputLocation",
    "name": "S3InputLocation"
  },
  {
    "startAt": "FIRST_ACTIVATION_DATE_TIME",
    "name": "Every 15 minutes",
    "period": "15 minutes",
    "occurrences": "4",
    "type": "Schedule",
    "id": "DefaultSchedule"
  },
  {
    "name": "ShellCommandActivityObj",
    "command": "#{myShellCmd}",
    "output": {
      "ref": "S3OutputLocation"
    }
  },

```

```

        "input": {
            "ref": "S3InputLocation"
        },
        "stage": "true",
        "type": "ShellCommandActivity",
        "id": "ShellCommandActivityObj",
        "runsOn": {
            "ref": "EC2ResourceObj"
        }
    }
],
"values": {
    "myS3OutputLoc": "s3://my-s3-bucket/",
    "myS3InputLoc": "s3://us-east-1.elasticmapreduce.samples/pig-apache-logs/
data",
    "myShellCmd": "grep -rc \"GET\" ${INPUT1_STAGING_DIR}/* >
${OUTPUT1_STAGING_DIR}/output.txt"
}
}

```

- Para API obtener más información, consulte [GetPipelineDefinition](#) la Referencia de AWS CLI comandos.

## list-pipelines

En el siguiente ejemplo de código, se muestra cómo usar `list-pipelines`.

### AWS CLI

Para enumerar tus canalizaciones

En este ejemplo, se muestran tus canalizaciones:

```
aws datapipeline list-pipelines
```

A continuación, se muestra un ejemplo de la salida:

```

{
  "pipelineIdList": [
    {
      "id": "df-00627471S0VYZEXAMPLE",
      "name": "my-pipeline"
    }
  ]
}

```

```

    },
    {
      "id": "df-09028963KNVMREXAMPLE",
      "name": "ImportDDB"
    },
    {
      "id": "df-0870198233ZYVEXAMPLE",
      "name": "CrossRegionDDB"
    },
    {
      "id": "df-00189603TB4MZEXAMPLE",
      "name": "CopyRedshift"
    }
  ]
}

```

- Para API obtener más información, consulta [ListPipelines](#) la Referencia de AWS CLI comandos.

## list-runs

En el siguiente ejemplo de código, se muestra cómo usar `list-runs`.

### AWS CLI

Ejemplo 1: Para enumerar las ejecuciones de tu canalización

En el siguiente `list-runs` ejemplo, se enumeran las ejecuciones de la canalización especificada.

```
aws datapipeline list-runs --pipeline-id df-00627471S0VYZEXAMPLE
```

Salida:

Name	Scheduled Start	Status	ID
	Started		Ended
1. EC2ResourceObj	2015-04-12T17:33:02	CREATING	
@EC2ResourceObj_2015-04-12T17:33:02			2015-04-12T17:33:10
2. S3InputLocation	2015-04-12T17:33:02	FINISHED	
@S3InputLocation_2015-04-12T17:33:02			2015-04-12T17:33:09
2015-04-12T17:33:09			

```

3. S3OutputLocation          2015-04-12T17:33:02    WAITING_ON_DEPENDENCIES
   @S3OutputLocation_2015-04-12T17:33:02    2015-04-12T17:33:09
4. ShellCommandActivityObj   2015-04-12T17:33:02    WAITING_FOR_RUNNER
   @ShellCommandActivityObj_2015-04-12T17:33:02    2015-04-12T17:33:09

```

Ejemplo 2: Para enumerar los recorridos de la canalización entre las fechas especificadas

En el siguiente `list-runs` ejemplo, se utiliza `--start-interval` para especificar las fechas que se van a incluir en la salida.

```
aws datapipeline list-runs --pipeline-id df-01434553B58A2SHZUK05 --start-
interval 2017-10-07T00:00:00,2017-10-08T00:00:00
```

- Para API obtener más información, consulte [ListRuns](#) la Referencia de AWS CLI comandos.

## put-pipeline-definition

En el siguiente ejemplo de código, se muestra cómo usar `put-pipeline-definition`.

### AWS CLI

Para cargar una definición de canalización

En este ejemplo, se carga la definición de canalización especificada en la canalización especificada:

```
aws datapipeline put-pipeline-definition --pipeline-id df-00627471S0VYZEXAMPLE --
pipeline-definition file://my-pipeline-definition.json
```

A continuación, se muestra un ejemplo de la salida:

```
{
  "validationErrors": [],
  "errored": false,
  "validationWarnings": []
}
```

- Para API obtener más información, consulte [PutPipelineDefinition](#) la Referencia de AWS CLI comandos.

## remove-tags

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags`.

### AWS CLI

Para eliminar una etiqueta de una canalización

En este ejemplo, se elimina la etiqueta especificada de la canalización especificada:

```
aws datapipeline remove-tags --pipeline-id df-00627471S0VYZEXAMPLE --tag-  
keys environment
```

- Para API obtener más información, consulte [RemoveTags](#) la Referencia de AWS CLI comandos.

## DataSync ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with DataSync.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

## update-location-azure-blob

En el siguiente ejemplo de código, se muestra cómo usar `update-location-azure-blob`.

### AWS CLI

Para actualizar tu ubicación de transferencia con un agente nuevo

En el siguiente `update-location-object-storage` ejemplo, se actualiza la DataSync ubicación de Microsoft Azure Blob Storage con un agente nuevo.

```
aws datasync update-location-azure-blob \  
  --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-  
abcdef01234567890 \  
  --agent-arns arn:aws:datasync:us-west-2:123456789012:agent/  
agent-1234567890abcdef0 \  
  --sas-configuration '{ \  
    "Token": "sas-token-for-azure-blob-storage-access" \  
  }'
```

Este comando no genera ninguna salida.

Para obtener más información, [consulte Sustitución de un agente](#) en la Guía del AWS DataSync usuario.

- Para API obtener más información, consulte [UpdateLocationAzureBlob](#) la Referencia de AWS CLI comandos.

## update-location-hdfs

En el siguiente ejemplo de código, se muestra cómo usar `update-location-hdfs`.

### AWS CLI

Para actualizar tu ubicación de transferencia con un agente nuevo

En el siguiente `update-location-hdfs` ejemplo, se actualiza tu DataSync HDFS ubicación con un agente nuevo. Solo necesitará las `--kerberos-krb5-conf` opciones `--kerberos-keytab` y si el HDFS clúster utiliza la autenticación Kerberos.

```
aws datasync update-location-hdfs \  
  --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-  
abcdef01234567890 \  
  --agent-arns arn:aws:datasync:us-west-2:123456789012:agent/  
agent-1234567890abcdef0 \  
  --kerberos-keytab file://hdfs.keytab \  
  --kerberos-krb5-conf file://krb5.conf
```

Contenidos de `hdfs.keytab`:

N/A. The content of this file is encrypted and not human readable.

## Contenidos de krb5.conf:

```
[libdefaults]
    default_realm = EXAMPLE.COM
    dns_lookup_realm = false
    dns_lookup_kdc = false
    rdns = true
    ticket_lifetime = 24h
    forwardable = true
    udp_preference_limit = 1000000
    default_tkt_etypes = aes256-cts-hmac-sha1-96 aes128-cts-hmac-sha1-96 des3-cbc-sha1
    default_tgs_etypes = aes256-cts-hmac-sha1-96 aes128-cts-hmac-sha1-96 des3-cbc-sha1
    permitted_etypes = aes256-cts-hmac-sha1-96 aes128-cts-hmac-sha1-96 des3-cbc-sha1

[realms]
    EXAMPLE.COM = {
        kdc = kdc1.example.com
        admin_server = krbadmin.example.com
        default_domain = example.com
    }

[domain_realm]
    .example.com = EXAMPLE.COM
    example.com = EXAMPLE.COM

[logging]
    kdc = FILE:/var/log/krb5kdc.log
    admin_server = FILE:/var/log/kerberos/kadmin.log
    default = FILE:/var/log/krb5libs.log
```

Este comando no genera ninguna salida.

Para obtener más información, [consulte Sustitución del agente](#) en la Guía del AWS DataSync usuario.

- Para API obtener más información, consulte [UpdateLocationHdfs](#) la Referencia de AWS CLI comandos.

## update-location-nfs

En el siguiente ejemplo de código, se muestra cómo usar `update-location-nfs`.

### AWS CLI

Para actualizar tu ubicación de transferencia con un agente nuevo

En el siguiente `update-location-nfs` ejemplo, se actualiza tu DataSync NFS ubicación con un agente nuevo.

```
aws datasync update-location-nfs \  
  --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-  
abcdef01234567890 \  
  --on-prem-config AgentArns=arn:aws:datasync:us-west-2:123456789012:agent/  
agent-1234567890abcdef0
```

Este comando no genera ninguna salida.

Para obtener más información, [consulte Sustitución de un agente](#) en la Guía del AWS DataSync usuario.

- Para API obtener más información, consulte [UpdateLocationNfs](#) la Referencia de AWS CLI comandos.

## update-location-object-storage

En el siguiente ejemplo de código, se muestra cómo usar `update-location-object-storage`.

### AWS CLI

Para actualizar tu ubicación de transferencia con un agente nuevo

En el siguiente `update-location-object-storage` ejemplo, se actualiza la ubicación de almacenamiento de DataSync objetos con un agente nuevo.

```
aws datasync update-location-object-storage \  
  --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-  
abcdef01234567890 \  
  --agent-arns arn:aws:datasync:us-west-2:123456789012:agent/  
agent-1234567890abcdef0 \  
  --secret-key secret-key-for-object-storage
```



Este comando no genera ninguna salida.

Para obtener más información, [consulte Sustitución de un agente](#) en la Guía del AWS DataSync usuario.

- Para API obtener más información, consulte [UpdateLocationObjectStorage](#) la Referencia de AWS CLI comandos.

## **update-location-smb**

En el siguiente ejemplo de código, se muestra cómo usar `update-location-smb`.

### AWS CLI

Para actualizar tu ubicación de transferencia con un agente nuevo

En el siguiente `update-location-smb` ejemplo, se actualiza tu DataSync SMB ubicación con un agente nuevo.

```
aws datasync update-location-smb \  
  --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-  
abcdef01234567890 \  
  --agent-arns arn:aws:datasync:us-west-2:123456789012:agent/  
agent-1234567890abcdef0 \  
  --password smb-file-server-password
```

Este comando no genera ninguna salida.

Para obtener más información, [consulte Sustitución de un agente](#) en la Guía del AWS DataSync usuario.

- Para API obtener más información, consulte [UpdateLocationSmb](#) la Referencia de AWS CLI comandos.

## DAXejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with DAX.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **create-cluster**

En el siguiente ejemplo de código, se muestra cómo usar `create-cluster`.

#### AWS CLI

Para crear un DAX clúster

En el siguiente `create-cluster` ejemplo, se crea un DAX clúster con la configuración especificada.

```
aws dax create-cluster \  
  --cluster-name daxcluster \  
  --node-type dax.r4.large \  
  --replication-factor 3 \  
  --iam-role-arn roleARN \  
  --sse-specification Enabled=true
```

Salida:

```
{  
  "Cluster": {  
    "ClusterName": "daxcluster",  
    "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",  
    "TotalNodes": 3,  
    "ActiveNodes": 0,  
    "NodeType": "dax.r4.large",  
    "Status": "creating",  
    "ClusterDiscoveryEndpoint": {  
      "Port": 8111  
    },  
    "PreferredMaintenanceWindow": "thu:13:00-thu:14:00",  
    "SubnetGroup": "default",  
    "SecurityGroups": [  

```

```

        {
            "SecurityGroupIdentifier": "sg-1af6e36e",
            "Status": "active"
        }
    ],
    "IamRoleArn": "arn:aws:iam::123456789012:role/
DAXServiceRoleForDynamoDBAccess",
    "ParameterGroup": {
        "ParameterGroupName": "default.dax1.0",
        "ParameterApplyStatus": "in-sync",
        "NodeIdsToReboot": []
    },
    "SSEDescription": {
        "Status": "ENABLED"
    }
}
}

```

Para obtener más información, consulte el [paso 3: Creación de un DAX clúster](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [CreateCluster](#) de AWS CLI comandos.

## create-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `create-parameter-group`.

### AWS CLI

Para crear un grupo de parámetros

El siguiente ejemplo de `create-parameter-group` crea un grupo de parámetros con los ajustes especificados.

```

aws dax create-parameter-group \
  --parameter-group-name daxparametergroup \
  --description "A new parameter group"

```

Salida:

```

{
  "ParameterGroup": {

```

```
    "ParameterGroupName": "daxparametergroup",
    "Description": "A new parameter group"
  }
}
```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [CreateParameterGroup](#) de AWS CLI comandos.

## create-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `create-subnet-group`.

### AWS CLI

Para crear un grupo de DAX subredes

En el siguiente `create-subnet-group` ejemplo, se crea un grupo de subredes con la configuración especificada.

```
aws dax create-subnet-group \
  --subnet-group-name daxSubnetGroup \
  --subnet-ids subnet-11111111 subnet-22222222
```

Salida:

```
{
  "SubnetGroup": {
    "SubnetGroupName": "daxSubnetGroup",
    "VpcId": "vpc-05a1fa8e00c325226",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-11111111",
        "SubnetAvailabilityZone": "us-west-2b"
      },
      {
        "SubnetIdentifier": "subnet-22222222",
        "SubnetAvailabilityZone": "us-west-2c"
      }
    ]
  }
}
```

```
}
```

Para obtener más información, consulte el [paso 2: Crear un grupo de subredes](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia de [CreateSubnetGroup](#) comandos AWS CLI .

## decrease-replication-factor

En el siguiente ejemplo de código, se muestra cómo usar `decrease-replication-factor`.

### AWS CLI

Para eliminar uno o más nodos del clúster

En el siguiente `decrease-replication-factor` ejemplo, se reduce a uno el número de nodos del DAX clúster especificado.

```
aws dax decrease-replication-factor \  
  --cluster-name daxcluster \  
  --new-replication-factor 1
```

Salida:

```
{  
  "Cluster": {  
    "ClusterName": "daxcluster",  
    "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",  
    "TotalNodes": 3,  
    "ActiveNodes": 3,  
    "NodeType": "dax.r4.large",  
    "Status": "modifying",  
    "ClusterDiscoveryEndpoint": {  
      "Address": "daxcluster.ey3o9d.clustercfg.dax.usw2.cache.amazonaws.com",  
      "Port": 8111  
    },  
    "Nodes": [  
      {  
        "NodeId": "daxcluster-a",  
        "Endpoint": {  
          "Address": "daxcluster-  
a.ey3o9d.0001.dax.usw2.cache.amazonaws.com",
```

```
        "Port": 8111
      },
      "NodeCreateTime": 1576625059.509,
      "AvailabilityZone": "us-west-2c",
      "NodeStatus": "available",
      "ParameterGroupStatus": "in-sync"
    },
    {
      "NodeId": "daxcluster-b",
      "Endpoint": {
        "Address": "daxcluster-
b.eyJ3o9d.0001.dax.usw2.cache.amazonaws.com",
        "Port": 8111
      },
      "NodeCreateTime": 1576625059.509,
      "AvailabilityZone": "us-west-2a",
      "NodeStatus": "available",
      "ParameterGroupStatus": "in-sync"
    },
    {
      "NodeId": "daxcluster-c",
      "Endpoint": {
        "Address": "daxcluster-
c.eyJ3o9d.0001.dax.usw2.cache.amazonaws.com",
        "Port": 8111
      },
      "NodeCreateTime": 1576625059.509,
      "AvailabilityZone": "us-west-2b",
      "NodeStatus": "available",
      "ParameterGroupStatus": "in-sync"
    }
  ],
  "PreferredMaintenanceWindow": "thu:13:00-thu:14:00",
  "SubnetGroup": "default",
  "SecurityGroups": [
    {
      "SecurityGroupIdentifier": "sg-1af6e36e",
      "Status": "active"
    }
  ],
  "IamRoleArn": "arn:aws:iam::123456789012:role/
DAXServiceRoleForDynamoDBAccess",
  "ParameterGroup": {
    "ParameterGroupName": "default.dax1.0",
```

```

        "ParameterApplyStatus": "in-sync",
        "NodeIdsToReboot": []
    },
    "SSEDescription": {
        "Status": "ENABLED"
    }
}
}
}

```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [DecreaseReplicationFactor](#) de AWS CLI comandos.

## delete-cluster

En el siguiente ejemplo de código, se muestra cómo usar `delete-cluster`.

### AWS CLI

Para eliminar un DAX clúster

En el siguiente `delete-cluster` ejemplo, se elimina el DAX clúster especificado.

```

aws dax delete-cluster \
  --cluster-name daxcluster

```

Salida:

```

{
  "Cluster": {
    "ClusterName": "daxcluster",
    "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",
    "TotalNodes": 3,
    "ActiveNodes": 0,
    "NodeType": "dax.r4.large",
    "Status": "deleting",
    "ClusterDiscoveryEndpoint": {
      "Address": "dd.ey3o9d.clustercfg.dax.usw2.cache.amazonaws.com",
      "Port": 8111
    }
  },
}

```

```

    "PreferredMaintenanceWindow": "fri:06:00-fri:07:00",
    "SubnetGroup": "default",
    "SecurityGroups": [
      {
        "SecurityGroupIdentifier": "sg-1af6e36e",
        "Status": "active"
      }
    ],
    "IamRoleArn": "arn:aws:iam::123456789012:role/
DAXServiceRoleForDynamoDBAccess",
    "ParameterGroup": {
      "ParameterGroupName": "default.dax1.0",
      "ParameterApplyStatus": "in-sync",
      "NodeIdsToReboot": []
    },
    "SSEDescription": {
      "Status": "ENABLED"
    }
  }
}

```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [DeleteCluster](#) de AWS CLI comandos.

## delete-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar delete-parameter-group.

### AWS CLI

Para eliminar un grupo de parámetros

En el siguiente delete-parameter-group ejemplo, se elimina el grupo de DAX parámetros especificado.

```

aws dax delete-parameter-group \
  --parameter-group-name daxparametergroup

```

Salida:



```
{
  "DeletionMessage": "Parameter group daxparametergroup has been deleted."
}
```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [DeleteParameterGroup](#) de AWS CLI comandos.

## delete-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-subnet-group`.

### AWS CLI

Para eliminar un grupo de subredes

En el siguiente `delete-subnet-group` ejemplo, se elimina el grupo de DAX subredes especificado.

```
aws dax delete-subnet-group \
  --subnet-group-name daxSubnetGroup
```

Salida:

```
{
  "DeletionMessage": "Subnet group daxSubnetGroup has been deleted."
}
```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [DeleteSubnetGroup](#) de AWS CLI comandos.

## describe-clusters

En el siguiente ejemplo de código, se muestra cómo usar `describe-clusters`.

## AWS CLI

Para devolver información sobre todos los clústeres aprovisionados DAX

En el siguiente `describe-clusters` ejemplo, se muestran detalles sobre todos los clústeres aprovisionados DAX.

```
aws dax describe-clusters
```

Salida:

```
{
  "Clusters": [
    {
      "ClusterName": "daxcluster",
      "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",
      "TotalNodes": 1,
      "ActiveNodes": 1,
      "NodeType": "dax.r4.large",
      "Status": "available",
      "ClusterDiscoveryEndpoint": {
        "Address":
"daxcluster.ey3o9d.clustercfg.dax.usw2.cache.amazonaws.com",
        "Port": 8111
      },
      "Nodes": [
        {
          "NodeId": "daxcluster-a",
          "Endpoint": {
            "Address": "daxcluster-
a.ey3o9d.0001.dax.usw2.cache.amazonaws.com",
            "Port": 8111
          },
          "NodeCreateTime": 1576625059.509,
          "AvailabilityZone": "us-west-2c",
          "NodeStatus": "available",
          "ParameterGroupStatus": "in-sync"
        }
      ],
      "PreferredMaintenanceWindow": "thu:13:00-thu:14:00",
      "SubnetGroup": "default",
      "SecurityGroups": [
        {
```

```

        "SecurityGroupIdentifier": "sg-1af6e36e",
        "Status": "active"
    }
],
    "IamRoleArn": "arn:aws:iam::123456789012:role/
DAXServiceRoleForDynamoDBAccess",
    "ParameterGroup": {
        "ParameterGroupName": "default.dax1.0",
        "ParameterApplyStatus": "in-sync",
        "NodeIdsToReboot": []
    },
    "SSEDescription": {
        "Status": "ENABLED"
    }
}
]
}

```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [DescribeClusters](#) de AWS CLI comandos.

## describe-default-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-default-parameters`.

### AWS CLI

Para devolver la información de los parámetros del sistema por defecto para DAX

En el siguiente `describe-default-parameters` ejemplo, se muestra la información de los parámetros del sistema por defecto de DAX.

```
aws dax describe-default-parameters
```

Salida:

```
{
  "Parameters": [
    {
```

```

    "ParameterName": "query-ttl-millis",
    "ParameterType": "DEFAULT",
    "ParameterValue": "300000",
    "NodeTypeSpecificValues": [],
    "Description": "Duration in milliseconds for queries to remain cached",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": "TRUE",
    "ChangeType": "IMMEDIATE"
  },
  {
    "ParameterName": "record-ttl-millis",
    "ParameterType": "DEFAULT",
    "ParameterValue": "300000",
    "NodeTypeSpecificValues": [],
    "Description": "Duration in milliseconds for records to remain valid in
cache (Default: 0 = infinite)",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": "TRUE",
    "ChangeType": "IMMEDIATE"
  }
]
}

```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [DescribeDefaultParameters](#) de AWS CLI comandos.

## describe-events

En el siguiente ejemplo de código, se muestra cómo usar `describe-events`.

### AWS CLI

Para devolver todos los eventos relacionados con los DAX clústeres y grupos de parámetros

En el siguiente `describe-events` ejemplo, se muestran los detalles de los eventos relacionados con DAX los clústeres y los grupos de parámetros.

**aws dax describe-events**

Salida:

```
{
  "Events": [
    {
      "SourceName": "daxcluster",
      "SourceType": "CLUSTER",
      "Message": "Cluster deleted.",
      "Date": 1576702736.706
    },
    {
      "SourceName": "daxcluster",
      "SourceType": "CLUSTER",
      "Message": "Removed node daxcluster-b.",
      "Date": 1576702691.738
    },
    {
      "SourceName": "daxcluster",
      "SourceType": "CLUSTER",
      "Message": "Removed node daxcluster-a.",
      "Date": 1576702633.498
    },
    {
      "SourceName": "daxcluster",
      "SourceType": "CLUSTER",
      "Message": "Removed node daxcluster-c.",
      "Date": 1576702631.329
    },
    {
      "SourceName": "daxcluster",
      "SourceType": "CLUSTER",
      "Message": "Cluster created.",
      "Date": 1576626560.057
    }
  ]
}
```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [DescribeEvents](#) de AWS CLI comandos.

## describe-parameter-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-parameter-groups`.

AWS CLI

Para describir los grupos de parámetros definidos en DAX

El siguiente `describe-parameter-groups` ejemplo recupera detalles sobre los grupos de parámetros que se definen en DAX.

```
aws dax describe-parameter-groups
```

Salida:

```
{
  "ParameterGroups": [
    {
      "ParameterGroupName": "default.dax1.0",
      "Description": "Default parameter group for dax1.0"
    }
  ]
}
```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [DescribeParameterGroups](#) de AWS CLI comandos.

## describe-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-parameters`.

AWS CLI

Para describir los parámetros definidos en un grupo de DAX parámetros

En el siguiente describe-parameters ejemplo, se recuperan detalles sobre los parámetros que se definen en el grupo de DAX parámetros especificado.

```
aws dax describe-parameters \  
  --parameter-group-name default.dax1.0
```

Salida:

```
{  
  "Parameters": [  
    {  
      "ParameterName": "query-ttl-millis",  
      "ParameterType": "DEFAULT",  
      "ParameterValue": "300000",  
      "NodeTypeSpecificValues": [],  
      "Description": "Duration in milliseconds for queries to remain cached",  
      "Source": "user",  
      "DataType": "integer",  
      "AllowedValues": "0-",  
      "IsModifiable": "TRUE",  
      "ChangeType": "IMMEDIATE"  
    },  
    {  
      "ParameterName": "record-ttl-millis",  
      "ParameterType": "DEFAULT",  
      "ParameterValue": "300000",  
      "NodeTypeSpecificValues": [],  
      "Description": "Duration in milliseconds for records to remain valid in  
cache (Default: 0 = infinite)",  
      "Source": "user",  
      "DataType": "integer",  
      "AllowedValues": "0-",  
      "IsModifiable": "TRUE",  
      "ChangeType": "IMMEDIATE"  
    }  
  ]  
}
```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [DescribeParameters](#) de AWS CLI comandos.

## describe-subnet-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-subnet-groups`.

### AWS CLI

Para describir los grupos de subredes definidos en DAX

En el siguiente `describe-subnet-groups` ejemplo, se recuperan los detalles de los grupos de subredes definidos en. DAX

```
aws dax describe-subnet-groups
```

Salida:

```
{
  "SubnetGroups": [
    {
      "SubnetGroupName": "default",
      "Description": "Default CacheSubnetGroup",
      "VpcId": "vpc-ee70a196",
      "Subnets": [
        {
          "SubnetIdentifier": "subnet-874953af",
          "SubnetAvailabilityZone": "us-west-2d"
        },
        {
          "SubnetIdentifier": "subnet-bd3d1fc4",
          "SubnetAvailabilityZone": "us-west-2a"
        },
        {
          "SubnetIdentifier": "subnet-72c2ff28",
          "SubnetAvailabilityZone": "us-west-2c"
        },
        {
          "SubnetIdentifier": "subnet-09e6aa42",
          "SubnetAvailabilityZone": "us-west-2b"
        }
      ]
    }
  ]
}
```



Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [DescribeSubnetGroups](#) de AWS CLI comandos.

## increase-replication-factor

En el siguiente ejemplo de código, se muestra cómo usar `increase-replication-factor`.

### AWS CLI

Para aumentar el factor de replicación de un DAX clúster

El siguiente `increase-replication-factor` ejemplo aumenta el factor de replicación del DAX clúster especificado a 3.

```
aws dax increase-replication-factor \  
  --cluster-name daxcluster \  
  --new-replication-factor 3
```

Salida:

```
{  
  "Cluster": {  
    "ClusterName": "daxcluster",  
    "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",  
    "TotalNodes": 3,  
    "ActiveNodes": 1,  
    "NodeType": "dax.r4.large",  
    "Status": "modifying",  
    "ClusterDiscoveryEndpoint": {  
      "Address": "daxcluster.ey3o9d.clustercfg.dax.usw2.cache.amazonaws.com",  
      "Port": 8111  
    },  
    "Nodes": [  
      {  
        "NodeId": "daxcluster-a",  
        "Endpoint": {  
          "Address": "daxcluster-  
a.ey3o9d.0001.dax.usw2.cache.amazonaws.com",  
          "Port": 8111  
        }  
      }  
    ]  
  }  
}
```

```

    },
    "NodeCreateTime": 1576625059.509,
    "AvailabilityZone": "us-west-2c",
    "NodeStatus": "available",
    "ParameterGroupStatus": "in-sync"
  },
  {
    "NodeId": "daxcluster-b",
    "NodeStatus": "creating"
  },
  {
    "NodeId": "daxcluster-c",
    "NodeStatus": "creating"
  }
],
"PreferredMaintenanceWindow": "thu:13:00-thu:14:00",
"SubnetGroup": "default",
"SecurityGroups": [
  {
    "SecurityGroupIdentifier": "sg-1af6e36e",
    "Status": "active"
  }
],
"IamRoleArn": "arn:aws:iam::123456789012:role/
DAXServiceRoleForDynamoDBAccess",
"ParameterGroup": {
  "ParameterGroupName": "default.dax1.0",
  "ParameterApplyStatus": "in-sync",
  "NodeIdsToReboot": []
},
"SSEDescription": {
  "Status": "ENABLED"
}
}
}

```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [IncreaseReplicationFactor](#) de AWS CLI comandos.

## list-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-tags`.

### AWS CLI

Para enumerar las etiquetas de un DAX recurso

En el siguiente `list-tags` ejemplo, se enumeran las claves y los valores de las etiquetas adjuntos al DAX clúster especificado.

```
aws dax list-tags \  
  --resource-name arn:aws:dax:us-west-2:123456789012:cache/daxcluster
```

Salida:

```
{  
  "Tags": [  
    {  
      "Key": "ClusterUsage",  
      "Value": "prod"  
    }  
  ]  
}
```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [ListTags](#) de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar un DAX recurso

En el siguiente `tag-resource` ejemplo, se adjunta el nombre de la clave de etiqueta especificada y el valor asociado al DAX clúster especificado para describir el uso del clúster.

```
aws dax tag-resource \  
  --resource-name arn:aws:dax:us-west-2:123456789012:cache/daxcluster
```

```
--resource-name arn:aws:dax:us-west-2:123456789012:cache/daxcluster \  
--tags="Key=ClusterUsage,Value=prod"
```

Salida:

```
{  
  "Tags": [  
    {  
      "Key": "ClusterUsage",  
      "Value": "prod"  
    }  
  ]  
}
```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [TagResource](#) de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

AWS CLI

Para eliminar etiquetas de un DAX recurso

En el siguiente `untag-resource` ejemplo, se quita la etiqueta con el nombre de clave especificado de un DAX clúster.

```
aws dax untag-resource \  
--resource-name arn:aws:dax:us-west-2:123456789012:cache/daxcluster \  
--tag-keys="ClusterUsage"
```

Salida:

```
{  
  "Tags": []  
}
```

Para obtener más información, consulte [Administración de DAX clústeres](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte la Referencia [UntagResource](#) de AWS CLI comandos.

## Ejemplos de Detectives que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface with Detective.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde encontrará instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **accept-invitation**

En el siguiente ejemplo de código, se muestra cómo usar `accept-invitation`.

AWS CLI

Para aceptar una invitación para convertirse en miembro de una cuenta en un gráfico de comportamiento

En el siguiente `accept-invitation` ejemplo, se acepta una invitación para convertirse en una cuenta miembro en Behavior Graph `arn:aws:detective:us-east-1:111122223333:graph:123412341234`.

```
aws detective accept-invitation \  
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Cómo responder a una invitación con un gráfico de comportamiento](#) en la Guía de administración de Amazon Detective.

- Para API obtener más información, consulte [AcceptInvitation](#) la Referencia de AWS CLI comandos.

## create-graph

En el siguiente ejemplo de código, se muestra cómo usar `create-graph`.

### AWS CLI

Para activar Amazon Detective y crear un nuevo gráfico de comportamiento

El siguiente `create-graph` ejemplo habilita Detective para la AWS cuenta que ejecuta el comando en la región en la que se ejecuta el comando. Se crea un nuevo gráfico de comportamiento que tiene esa cuenta como cuenta de administrador. El comando también asigna el valor Finanzas a la etiqueta Departamento.

```
aws detective create-graph \  
  --tags '{"Department": "Finance"}
```

Salida:

```
{  
  "GraphArn": "arn:aws:detective:us-  
east-1:111122223333:graph:027c7c4610ea4aacaf0b883093cab899"  
}
```

Para obtener más información, consulte [Activación de Amazon Detective](#) en la Guía de administración de Amazon Detective.

- Para API obtener más información, consulte [CreateGraph](#) la Referencia de AWS CLI comandos.

## create-members

En el siguiente ejemplo de código, se muestra cómo usar `create-members`.

### AWS CLI

Para invitar a las cuentas de los miembros a un gráfico de comportamiento

El siguiente `create-members` ejemplo invita a dos AWS cuentas a convertirse en cuentas de miembros en el gráfico de comportamiento `arn:aws:detective:us-east-1:111122223333:graph:123412341234`. Para cada cuenta, la solicitud proporciona el ID de la cuenta y la dirección de correo electrónico del usuario raíz de la cuenta. AWS La solicitud incluye un mensaje personalizado para insertarlo en el correo electrónico de invitación.

```
aws detective create-members \
  --
accounts AccountId=444455556666,EmailAddress=mmajor@example.com AccountId=123456789012,Email
\
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234 \
  --message "This is Paul Santos. I need to add your account to the data we use
for security investigation in Amazon Detective. If you have any questions, contact
me at psantos@example.com."
```

Salida:

```
{
  "Members": [
    {
      "AccountId": "444455556666",
      "AdministratorId": "111122223333",
      "EmailAddress": "mmajor@example.com",
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
      "InvitedTime": 1579826107000,
      "MasterId": "111122223333",
      "Status": "INVITED",
      "UpdatedTime": 1579826107000
    },
    {
      "AccountId": "123456789012",
      "AdministratorId": "111122223333",
      "EmailAddress": "jstiles@example.com",
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
      "InvitedTime": 1579826107000,
      "MasterId": "111122223333",
      "Status": "VERIFICATION_IN_PROGRESS",
      "UpdatedTime": 1579826107000
    }
  ],
  "UnprocessedAccounts": [ ]
}
```

Para obtener más información, consulta [Cómo invitar a las cuentas de los miembros a un gráfico de comportamiento](https://docs.aws.amazon.com/detective/latest/adminguide/graph-admin-add-member-accounts.html) en la Guía de administración de Amazon Detective.

Para invitar a cuentas de miembros sin enviar correos electrónicos de invitación

El siguiente `create-members` ejemplo invita a dos AWS cuentas a convertirse en cuentas de miembros en el gráfico de comportamiento `arn:aws:detective:us-east-1:111122223333:graph:123412341234`. Para cada cuenta, la solicitud proporciona el ID de la cuenta y la dirección de correo electrónico del usuario raíz de la cuenta. AWS Las cuentas de los miembros no reciben correos electrónicos de invitación.

```
aws detective create-members \
  --
accounts AccountId=444455556666,EmailAddress=mmajor@example.com AccountId=123456789012,Email
\
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234 \
  --disable-email-notification
```

Salida:

```
{
  "Members": [
    {
      "AccountId": "444455556666",
      "AdministratorId": "111122223333",
      "EmailAddress": "mmajor@example.com",
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
      "InvitedTime": 1579826107000,
      "MasterId": "111122223333",
      "Status": "INVITED",
      "UpdatedTime": 1579826107000
    },
    {
      "AccountId": "123456789012",
      "AdministratorId": "111122223333",
      "EmailAddress": "jstiles@example.com",
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
      "InvitedTime": 1579826107000,
      "MasterId": "111122223333",
      "Status": "VERIFICATION_IN_PROGRESS",
      "UpdatedTime": 1579826107000
    }
  ]
}
```



```
    }  
  ],  
  "UnprocessedAccounts": [ ]  
}
```

Para obtener más información, consulta [Cómo invitar a las cuentas de los miembros a un gráfico de comportamiento](https://docs.aws.amazon.com/detective/latest/adminguide/graph-admin-add-member-accounts.html) en la Guía de administración de Amazon Detective.

- Para obtener API más información, [CreateMembers](#) consulte AWS CLI la Referencia de comandos.

## delete-graph

En el siguiente ejemplo de código, se muestra cómo usar `delete-graph`.

### AWS CLI

Para deshabilitar Detective y eliminar el gráfico de comportamiento

En el siguiente `delete-graph` ejemplo, se desactiva Detective y se elimina el gráfico de comportamiento especificado.

```
aws detective delete-graph \  
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Desactivación de Amazon Detective](#) en la Guía de administración de Amazon Detective.

- Para API obtener más información, consulte [DeleteGraph](#) la Referencia de AWS CLI comandos.

## delete-members

En el siguiente ejemplo de código, se muestra cómo usar `delete-members`.

### AWS CLI

Para eliminar las cuentas de los miembros de un gráfico de comportamiento

En el siguiente `delete-members` ejemplo, se eliminan dos cuentas de miembros del gráfico de comportamiento `arn:aws:detective:us-east-1:111122223333:graph:123412341234`. Para identificar las AWS IDs cuentas, la solicitud proporciona la cuenta.

```
aws detective delete-members \  
  --account-ids 444455556666 123456789012 \  
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Salida:

```
{  
  "AccountIds": [ "444455556666", "123456789012" ],  
  "UnprocessedAccounts": [ ]  
}
```

Para obtener más información, consulta [Eliminar cuentas de miembros de un gráfico de comportamiento](https://docs.aws.amazon.com/detective/latest/adminguide/graph-admin-remove-member-accounts.html) en la Guía de administración de Amazon Detective.

- Para obtener API más información, [DeleteMembers](#) consulte AWS CLI la Referencia de comandos.

## **disassociate-membership**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-membership`.

AWS CLI

Para renunciar a la membresía de un gráfico de comportamiento

El siguiente ejemplo de disociar la membresía elimina la AWS cuenta que ejecuta el comando del gráfico de comportamiento `arn:aws:detective:us-east-1:111122223333:graph:123412341234`.

```
aws detective disassociate-membership \  
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Para obtener más información, consulta [Eliminar tu cuenta de un gráfico de comportamiento](https://docs.aws.amazon.com/detective/latest/adminguide/member-remove-self-from-graph.html) en la Guía de administración de Amazon Detective.

- Para obtener API más información, [DisassociateMembership](#) consulte AWS CLI la Referencia de comandos.

## get-members

En el siguiente ejemplo de código, se muestra cómo usar `get-members`.

### AWS CLI

Para recuperar información sobre las cuentas de los miembros seleccionadas, grafique su comportamiento

El siguiente `get-members` ejemplo recupera información sobre dos cuentas de miembros en el gráfico de comportamiento `arn:aws:detective:us-east-1:111122223333:graph:123412341234`.

Para las dos cuentas AWS , IDs la solicitud proporciona la cuenta.

```
aws detective get-members \  
  --account-ids 444455556666 123456789012 \  
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Salida:

```
{  
  "MemberDetails": [  
    {  
      "AccountId": "444455556666",  
      "AdministratorId": "111122223333",  
      "EmailAddress": "mmajor@example.com",  
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",  
      "InvitedTime": 1579826107000,  
      "MasterId": "111122223333",  
      "Status": "INVITED",  
      "UpdatedTime": 1579826107000  
    }  
    {  
      "AccountId": "123456789012",  
      "AdministratorId": "111122223333",  
      "EmailAddress": "jstiles@example.com",  
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",  
      "InvitedTime": 1579826107000,  
      "MasterId": "111122223333",  
      "Status": "INVITED",
```

```
    "UpdateTime": 1579826107000
  }
],
  "UnprocessedAccounts": [ ]
}
```

Para obtener más información, consulte [Visualización de la lista de cuentas en un gráfico de comportamiento](https://docs.aws.amazon.com/detective/latest/adminguide/graph-admin-view-accounts.html) en la Guía de administración de Amazon Detective.

- Para obtener más información, consulte la Referencia de comandos. API [GetMembers](#) AWS CLI

## list-graphs

En el siguiente ejemplo de código, se muestra cómo usar `list-graphs`.

### AWS CLI

Para ver una lista de gráficos de comportamiento de los que su cuenta es administradora

En el siguiente `list-graphs` ejemplo, se recuperan los gráficos de comportamiento de los que la cuenta que realiza la llamada es la administradora en la región actual.

```
aws detective list-graphs
```

Salida:

```
{
  "GraphList": [
    {
      "Arn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
      "CreatedTime": 1579736111000
    }
  ]
}
```

- Para API obtener más información, consulte [ListGraphs](#) la Referencia de AWS CLI comandos.

## list-invitations

En el siguiente ejemplo de código, se muestra cómo usar `list-invitations`.

## AWS CLI

Para ver una lista de gráficos de comportamiento de los que una cuenta es miembro o a los que está invitada

En el siguiente `list-invitations` ejemplo, se recuperan los gráficos de comportamiento a los que se ha invitado a la cuenta que realiza la llamada. Los resultados incluyen solo las invitaciones abiertas y aceptadas. No incluyen las invitaciones rechazadas ni las membresías eliminadas.

```
aws detective list-invitations
```

Salida:

```
{
  "Invitations": [
    {
      "AccountId": "444455556666",
      "AdministratorId": "111122223333",
      "EmailAddress": "mmajor@example.com",
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
      "InvitedTime": 1579826107000,
      "MasterId": "111122223333",
      "Status": "INVITED",
      "UpdatedTime": 1579826107000
    }
  ]
}
```

Para obtener más información, consulte [Visualización de la lista de invitaciones a gráficos de comportamiento](https://docs.aws.amazon.com/detective/latest/adminguide/member-view-graph-invitations) < <https://docs.aws.amazon.com/detective/latest/adminguide/member-view-graph-invitations> en la Guía de administración de Amazon Detective.

- Para obtener API más información, [ListInvitations](#) consulte AWS CLI la Referencia de comandos.

## **list-members**

En el siguiente ejemplo de código, se muestra cómo usar `list-members`.

## AWS CLI

Para enumerar las cuentas de los miembros en un gráfico de comportamiento

En el siguiente `list-members` ejemplo, se recuperan las cuentas de los miembros invitados y habilitados para el gráfico `arn:aws:detective:us-east-1:111122223333:graph:123412341234` de comportamiento. Los resultados no incluyen las cuentas de los miembros que se eliminaron.

```
aws detective list-members \  
--graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Salida:

```
{  
  "MemberDetails": [  
    {  
      "AccountId": "444455556666",  
      "AdministratorId": "111122223333",  
      "EmailAddress": "mmajor@example.com",  
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",  
      "InvitedTime": 1579826107000,  
      "MasterId": "111122223333",  
      "Status": "INVITED",  
      "UpdateTime": 1579826107000  
    },  
    {  
      "AccountId": "123456789012",  
      "AdministratorId": "111122223333",  
      "EmailAddress": "jstiles@example.com",  
      "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",  
      "InvitedTime": 1579826107000,  
      "MasterId": "111122223333",  
      "PercentOfGraphUtilization": 2,  
      "PercentOfGraphUtilizationUpdateTime": 1586287843,  
      "Status": "ENABLED",  
      "UpdateTime": 1579973711000,  
      "VolumeUsageInBytes": 200,  
      "VolumeUsageUpdateTime": 1586287843  
    }  
  ]  
}
```

Para obtener más información, consulte [Visualización de la lista de cuentas en un gráfico de comportamiento](#) en la Guía de administración de Amazon Detective.

- Para API obtener más información, consulte [ListMembers](#) la Referencia de AWS CLI comandos.

## **list-tags-for-resource**

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para recuperar las etiquetas asignadas a un gráfico de comportamiento

El siguiente `list-tags-for-resource` ejemplo devuelve las etiquetas asignadas al gráfico de comportamiento especificado.

```
aws detective list-tags-for-resource \  
--resource-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Salida:

```
{  
  "Tags": {  
    "Department" : "Finance"  
  }  
}
```

Para obtener más información, consulte [Gestión de etiquetas para un gráfico de comportamiento](#) en la Guía de administración de Amazon Detective.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## **reject-invitation**

En el siguiente ejemplo de código, se muestra cómo usar `reject-invitation`.

### AWS CLI

Para rechazar una invitación a convertirse en miembro de una cuenta en un gráfico de comportamiento

En el siguiente `reject-invitation` ejemplo, se rechaza una invitación para convertirse en una cuenta miembro del gráfico de comportamiento `arn:aws:detective:us-east-1:111122223333:graph:123412341234`.

```
aws detective reject-invitation \  
  --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Responder a una invitación a un gráfico de comportamiento](https://docs.aws.amazon.com/detective/latest/adminguide/member-invitation-response) < <https://docs.aws.amazon.com/detective/latest/adminguide/member-invitation-response> en la Guía de administración de Amazon Detective.

- Para obtener API más información, [RejectInvitation](#) consulte AWS CLI la Referencia de comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para asignar una etiqueta a un recurso

En el siguiente `tag-resource` ejemplo, se asigna un valor para la etiqueta `Department` al gráfico de comportamiento especificado.

```
aws detective tag-resource \  
  --resource-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234 \  
  --tags '{"Department": "Finance"}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión de etiquetas para un gráfico de comportamiento](#) en la Guía de administración de Amazon Detective.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.



## AWS CLI

Para eliminar el valor de una etiqueta de un recurso

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta `Department` del gráfico de comportamiento especificado.

```
aws detective untag-resource \  
  --resource-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234 \  
  --tag-keys "Department"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión de etiquetas para un gráfico de comportamiento](#) en la Guía de administración de Amazon Detective.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## Ejemplos de Device Farm que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Device Farm.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-device-pool**

En el siguiente ejemplo de código, se muestra cómo usar `create-device-pool`.

## AWS CLI

Para crear un grupo de dispositivos

El siguiente comando crea un grupo de dispositivos Android para un proyecto:

```
aws devicefarm create-device-pool --name pool1 --rules file://  
device-pool-rules.json --project-arn "arn:aws:devicefarm:us-  
west-2:123456789012:project:070fc3ca-7ec1-4741-9c1f-d3e044efc506"
```

Puede obtener el proyecto ARN a partir de la salida de `create-project` o `list-projects`. El archivo `device-pool-rules.json` es un JSON documento de la carpeta actual que especifica la plataforma del dispositivo:

```
[  
  {  
    "attribute": "PLATFORM",  
    "operator": "EQUALS",  
    "value": "\"ANDROID\""  
  }  
]
```

Salida:

```
{  
  "devicePool": {  
    "rules": [  
      {  
        "operator": "EQUALS",  
        "attribute": "PLATFORM",  
        "value": "\"ANDROID\""  
      }  
    ],  
    "type": "PRIVATE",  
    "name": "pool1",  
    "arn": "arn:aws:devicefarm:us-  
west-2:123456789012:devicepool:070fc3ca-7ec1-4741-9c1f-  
d3e044efc506/2aa8d2a9-5e73-47ca-b929-659cb34b7dcd"  
  }  
}
```

- Para API obtener más información, consulte [CreateDevicePool](#) la Referencia de AWS CLI comandos.

## create-project

En el siguiente ejemplo de código, se muestra cómo usar `create-project`.

AWS CLI

Para crear un proyecto

El siguiente comando crea un nuevo proyecto denominado `my-project`:

```
aws devicefarm create-project --name my-project
```

Salida:

```
{
  "project": {
    "name": "myproject",
    "arn": "arn:aws:devicefarm:us-
west-2:123456789012:project:070fc3ca-7ec1-4741-9c1f-d3e044efc506",
    "created": 1503612890.057
  }
}
```

- Para API obtener más información, consulte [CreateProject](#) la Referencia de AWS CLI comandos.

## create-upload

En el siguiente ejemplo de código, se muestra cómo usar `create-upload`.

AWS CLI

Para crear una carga

El siguiente comando crea una carga para una aplicación de Android:

```
aws devicefarm create-upload --project-arn "arn:aws:devicefarm:us-west-2:123456789012:project:070fc3ca-7ec1-4741-9c1f-d3e044efc506" --name app.apk --type ANDROID_APP
```

Puedes obtener el proyecto ARN desde el resultado de `create-project` o `list-projects`.

Salida:

```
{
  "upload": {
    "status": "INITIALIZED",
    "name": "app.apk",
    "created": 1503614408.769,
    "url": "https://prod-us-west-2-uploads.s3-us-west-2.amazonaws.com/arn%3Aaws%3Adevicefarm%3Aus-west-2%3A123456789012%3Aproject%3A070fc3ca-c7e1-4471-91cf-d3e4efc50604/uploads/arn%3Aaws%3Adevicefarm%3Aus-west-2%3A123456789012%3Aupload%3A070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-ae9e-4087-09e6-f4cea3599514/app.apk?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20170824T224008Z&X-Amz-SignedHeaders=host&X-Amz-Expires=86400&X-Amz-Credential=AKIAEXAMPLEPBUMBC3GA%2F20170824%2Fus-west-2%2Fs%32Faws4_request&X-Amz-Signature=05050370c38894ef5bd09f5d009f36fc8f96fa4bb04e1bba9aca71b8dbe49a0f",
    "type": "ANDROID_APP",
    "arn": "arn:aws:devicefarm:us-west-2:123456789012:upload:070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-ae9e-4087-09e6-f4cea3599514"
  }
}
```

Usa la información URL de salida iniciada para subir un archivo a Device Farm:

```
curl -T app.apk "https://prod-us-west-2-uploads.s3-us-west-2.amazonaws.com/arn%3Aaws%3Adevicefarm%3Aus-west-2%3A123456789012%3Aproject%3A070fc3ca-c7e1-4471-91cf-d3e4efc50604/uploads/arn%3Aaws%3Adevicefarm%3Aus-west-2%3A123456789012%3Aupload%3A070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-ae9e-4087-09e6-f4cea3599514/app.apk?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20170824T224008Z&X-Amz-SignedHeaders=host&X-Amz-Expires=86400&X-Amz-Credential=AKIAEXAMPLEPBUMBC3GA%2F20170824%2Fus-west-2%2Fs%32Faws4_request&X-Amz-Signature=05050370c38894ef5bd09f5d009f36fc8f96fa4bb04e1bba9aca71b8dbe49a0f"
```

- Para API obtener más información, consulte [CreateUpload](#) la Referencia de AWS CLI comandos.

## get-upload

En el siguiente ejemplo de código, se muestra cómo usar `get-upload`.

### AWS CLI

Para ver y subir

El siguiente comando recupera información sobre una carga:

```
aws devicefarm get-upload --arn "arn:aws:devicefarm:us-west-2:123456789012:upload:070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-ae9e-4087-09e6-f4cea3599514"
```

Puede obtener la carga a ARN partir de la salida de `create-upload`.

Salida:

```
{
  "upload": {
    "status": "SUCCEEDED",
    "name": "app.apk",
    "created": 1505262773.186,
    "type": "ANDROID_APP",
    "arn": "arn:aws:devicefarm:us-west-2:123456789012:upload:070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-ae9e-4087-09e6-f4cea3599514",
    "metadata": "{\"device_admin\":false,\"activity_name\": \"com.example.client.LauncherActivity\", \"version_name\": \"1.0.2.94\", \"screens\": [\"small\", \"normal\", \"large\", \"xlarge\"], \"error_type\": null, \"sdk_version\": \"16\", \"package_name\": \"com.example.client\", \"version_code\": \"20994\", \"native_code\": [\"armeabi-v7a\"], \"target_sdk_version\": \"25\"}"
  }
}
```

- Para API obtener más información, consulte [GetUpload](#) la Referencia de AWS CLI comandos.

## list-projects

En el siguiente ejemplo de código, se muestra cómo usar `list-projects`.

## AWS CLI

Para enumerar los proyectos

A continuación, se obtiene una lista de proyectos:

```
aws devicefarm list-projects
```

Salida:

```
{
  "projects": [
    {
      "name": "myproject",
      "arn": "arn:aws:devicefarm:us-west-2:123456789012:project:070fc3ca-7ec1-4741-9c1f-d3e044efc506",
      "created": 1503612890.057
    },
    {
      "name": "otherproject",
      "arn": "arn:aws:devicefarm:us-west-2:123456789012:project:a5f5b752-8098-49d1-86bf-5f7682c1c77e",
      "created": 1505257519.337
    }
  ]
}
```

- Para API obtener más información, consulte [ListProjects](#) la Referencia de AWS CLI comandos.

## AWS Direct Connect ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Direct Connect.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **accept-direct-connect-gateway-association-proposal**

En el siguiente ejemplo de código, se muestra cómo usar `accept-direct-connect-gateway-association-proposal`.

#### AWS CLI

Para aceptar una propuesta de asociación de pasarelas

Lo siguiente `accept-direct-connect-gateway-association-proposal` acepta la propuesta especificada.

```
aws directconnect accept-direct-connect-gateway-association-proposal \  
  --direct-connect-gateway-id 11460968-4ac1-4fd3-bdb2-00599EXAMPLE \  
  --proposal-id cb7f41cb-8128-43a5-93b1-dcaedEXAMPLE \  
  --associated-gateway-owner-account 111122223333  
  
{  
  "directConnectGatewayAssociation": {  
    "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",  
    "directConnectGatewayOwnerAccount": "111122223333",  
    "associationState": "associating",  
    "associatedGateway": {  
      "id": "tgw-02f776b1a7EXAMPLE",  
      "type": "transitGateway",  
      "ownerAccount": "111122223333",  
      "region": "us-east-1"  
    },  
    "associationId": "6441f8bf-5917-4279-ade1-9708bEXAMPLE",  
    "allowedPrefixesToDirectConnectGateway": [  
      {  
        "cidr": "192.168.1.0/30"  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [Aceptar o rechazar una propuesta de Transit Gateway Association](#) en la Guía del usuario de AWS Direct Connect.

- Para API obtener más información, consulte [AcceptDirectConnectGatewayAssociationProposalla](#) Referencia de AWS CLI comandos.

## allocate-connection-on-interconnect

En el siguiente ejemplo de código, se muestra cómo usar `allocate-connection-on-interconnect`.

### AWS CLI

Para crear una conexión alojada en una interconexión

El siguiente `allocate-connection-on-interconnect` comando crea una conexión alojada en una interconexión:

```
aws directconnect allocate-connection-on-interconnect --bandwidth 500Mbps --  
connection-name mydcinterconnect --owner-account 123456789012 --interconnect-  
id dxcon-fgktov66 --vlan 101
```

Salida:

```
{  
  "partnerName": "TIVIT",  
  "vlan": 101,  
  "ownerAccount": "123456789012",  
  "connectionId": "dxcon-ffzc51m1",  
  "connectionState": "ordering",  
  "bandwidth": "500Mbps",  
  "location": "TIVIT",  
  "connectionName": "mydcinterconnect",  
  "region": "sa-east-1"  
}
```

- Para API obtener más información, consulte [AllocateConnectionOnInterconnect](#)la Referencia de AWS CLI comandos.



## allocate-hosted-connection

En el siguiente ejemplo de código, se muestra cómo usar `allocate-hosted-connection`.

### AWS CLI

Para crear una conexión alojada en una interconexión

El siguiente `allocate-hosted-connection` ejemplo crea una conexión alojada en la interconexión especificada.

```
aws directconnect allocate-hosted-connection \  
  --bandwidth 500Mbps \  
  --connection-name mydcinterconnect \  
  --owner-account 123456789012 \  
  -connection-id dxcon-fgktov66 \  
  -vlan 101
```

Salida:

```
{  
  "partnerName": "TIVIT",  
  "vlan": 101,  
  "ownerAccount": "123456789012",  
  "connectionId": "dxcon-ffzc51m1",  
  "connectionState": "ordering",  
  "bandwidth": "500Mbps",  
  "location": "TIVIT",  
  "connectionName": "mydcinterconnect",  
  "region": "sa-east-1"  
}
```

- Para API obtener más información, consulte [AllocateHostedConnection](#) la Referencia de AWS CLI comandos.

## allocate-private-virtual-interface

En el siguiente ejemplo de código, se muestra cómo usar `allocate-private-virtual-interface`.

## AWS CLI

Para aprovisionar una interfaz virtual privada

El siguiente `allocate-private-virtual-interface` comando proporciona una interfaz virtual privada que será propiedad de otro cliente:

```
aws directconnect allocate-private-virtual-interface --connection-id dxcon-ffjrnx17 --owner-account 123456789012 --new-private-virtual-interface-allocation virtualInterfaceName=PrivateVirtualInterface,vlan=1000,asn=65000,authKey=asdf34ex
```

Salida:

```
{
  "virtualInterfaceState": "confirming",
  "asn": 65000,
  "vlan": 1000,
  "customerAddress": "192.168.1.2/30",
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-ffjrnx17",
  "virtualInterfaceId": "dxvif-fgy8orxu",
  "authKey": "asdf34example",
  "routeFilterPrefixes": [],
  "location": "TIVIT",
  "customerRouterConfig": "<?xml version='1.0' encoding='UTF-8'?>
>\n <logical_connection id='dxvif-fgy8orxu'>\n <vlan>1000</
vlan>\n <customer_address>192.168.1.2/30</customer_address>\n
<amazon_address>192.168.1.1/30</amazon_address>\n <bgp_asn>65000</bgp_asn>\n
<bgp_auth_key>asdf34example</bgp_auth_key>\n <amazon_bgp_asn>7224</amazon_bgp_asn>
\n <connection_type>private</connection_type>\n</logical_connection>\n",
  "amazonAddress": "192.168.1.1/30",
  "virtualInterfaceType": "private",
  "virtualInterfaceName": "PrivateVirtualInterface"
}
```

- Para API obtener más información, consulte [AllocatePrivateVirtualInterface](#) la Referencia de AWS CLI comandos.

## `allocate-public-virtual-interface`

En el siguiente ejemplo de código, se muestra cómo usar `allocate-public-virtual-interface`.

## AWS CLI

Para aprovisionar una interfaz virtual pública

El siguiente `allocate-public-virtual-interface` comando proporciona una interfaz virtual pública que será propiedad de otro cliente:

```
aws directconnect allocate-public-virtual-interface --connection-id dxcon-ffjrnx17 --owner-account 123456789012 --new-public-virtual-interface-allocation virtualInterfaceName=PublicVirtualInterface,vlan=2000,asn=65000,authKey=asdf34example,cidr=203.0.113.4/30}
```

Salida:

```
{
  "virtualInterfaceState": "confirming",
  "asn": 65000,
  "vlan": 2000,
  "customerAddress": "203.0.113.2/30",
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-ffjrnx17",
  "virtualInterfaceId": "dxvif-fg9xo9vp",
  "authKey": "asdf34example",
  "routeFilterPrefixes": [
    {
      "cidr": "203.0.113.0/30"
    },
    {
      "cidr": "203.0.113.4/30"
    }
  ],
  "location": "TIVIT",
  "customerRouterConfig": "<?xml version='1.0' encoding='UTF-8'?>\n<logical_connection id='dxvif-fg9xo9vp'>\n  <vlan>2000</vlan>\n  <customer_address>203.0.113.2/30</customer_address>\n  <amazon_address>203.0.113.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n  <bgp_auth_key>asdf34example</bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>\n  <connection_type>public</connection_type>\n</logical_connection>\n",
  "amazonAddress": "203.0.113.1/30",
  "virtualInterfaceType": "public",
  "virtualInterfaceName": "PublicVirtualInterface"
}
```

- Para API obtener más información, consulte [AllocatePublicVirtualInterface](#) la Referencia de AWS CLI comandos.

## allocate-transit-virtual-interface

En el siguiente ejemplo de código, se muestra cómo usar `allocate-transit-virtual-interface`.

### AWS CLI

Para aprovisionar una interfaz virtual de tránsito que sea propiedad de la AWS cuenta especificada

El siguiente `allocate-transit-virtual-interface` ejemplo proporciona una interfaz virtual de tránsito para la cuenta especificada.

```
aws directconnect allocate-transit-virtual-interface \  
  --connection-id dxlag-fEXAMPLE \  
  --owner-account 123456789012 \  
  --new-transit-virtual-interface-allocation "virtualInterfaceName=Example Transit  
Virtual  
Interface,vlan=126,asn=65110,mtu=1500,authKey=0xzxgA9YoW9h58u8SEXAMPLE,amazonAddress=192.168.1.1/30"
```

Salida:

```
{  
  "virtualInterface": {  
    "ownerAccount": "123456789012",  
    "virtualInterfaceId": "dxvif-fEXAMPLE",  
    "location": "loc1",  
    "connectionId": "dxlag-fEXAMPLE",  
    "virtualInterfaceType": "transit",  
    "virtualInterfaceName": "Example Transit Virtual Interface",  
    "vlan": 126,  
    "asn": 65110,  
    "amazonSideAsn": 7224,  
    "authKey": "0xzxgA9YoW9h58u8SEXAMPLE",  
    "amazonAddress": "192.168.1.1/30",  
    "customerAddress": "192.168.1.2/30",  
    "addressFamily": "ipv4",  
    "virtualInterfaceState": "confirming",
```

```

    "customerRouterConfig": "<?xml version=\"1.0\" encoding=
    \"UTF-8\"?>\n<logical_connection id=\"dxvif-fEXAMPLE\">\n  <vlan>126</
    vlan>\n  <customer_address>192.168.1.2/30</customer_address>\n
    <amazon_address>192.168.1.1/30</amazon_address>\n  <bgp_asn>65110</bgp_asn>\n
    <bgp_auth_key>0xzxcgA9YoW9h58u8SEXAMPLE</bgp_auth_key>\n  <amazon_bgp_asn>7224</
    amazon_bgp_asn>\n  <connection_type>transit</connection_type>\n</logical_connection>
    \n",
    "mtu": 1500,
    "jumboFrameCapable": true,
    "virtualGatewayId": "",
    "directConnectGatewayId": "",
    "routeFilterPrefixes": [],
    "bgpPeers": [
      {
        "bgpPeerId": "dxpeer-fEXAMPLE",
        "asn": 65110,
        "authKey": "0xzxcgA9YoW9h58u8SEXAMPLE",
        "addressFamily": "ipv4",
        "amazonAddress": "192.168.1.1/30",
        "customerAddress": "192.168.1.2/30",
        "bgpPeerState": "pending",
        "bgpStatus": "down",
        "awsDeviceV2": "loc1-26wz6vEXAMPLE"
      }
    ],
    "region": "sa-east-1",
    "awsDeviceV2": "loc1-26wz6vEXAMPLE",
    "tags": [
      {
        "key": "Tag",
        "value": "Example"
      }
    ]
  }
}

```

Para obtener más información, consulte [Creación de una interfaz virtual de tránsito alojada](#) en la Guía del usuario de AWS Direct Connect.

- Para API obtener más información, consulte [AllocateTransitVirtualInterface](#) la Referencia de AWS CLI comandos.

## associate-connection-with-lag

En el siguiente ejemplo de código, se muestra cómo usar `associate-connection-with-lag`.

### AWS CLI

Para asociar una conexión a un LAG

El siguiente ejemplo asocia la conexión especificada a la especificada LAG.

Comando:

```
aws directconnect associate-connection-with-lag --lag-id dxlag-fhccu14t --  
connection-id dxcon-fg9607vm
```

Salida:

```
{  
  "ownerAccount": "123456789012",  
  "connectionId": "dxcon-fg9607vm",  
  "lagId": "dxlag-fhccu14t",  
  "connectionState": "requested",  
  "bandwidth": "1Gbps",  
  "location": "EqDC2",  
  "connectionName": "Con2ForLag",  
  "region": "us-east-1"  
}
```

- Para API obtener más información, consulte [AssociateConnectionWithLag](#) la Referencia de AWS CLI comandos.

## associate-hosted-connection

En el siguiente ejemplo de código, se muestra cómo usar `associate-hosted-connection`.

### AWS CLI

Para asociar una conexión hospedada a un LAG

El siguiente ejemplo asocia la conexión hospedada especificada a la especificada LAG.

Comando:

```
aws directconnect associate-hosted-connection --parent-connection-id dxlag-fhccu14t
--connection-id dxcon-fg9607vm
```

Salida:

```
{
  "partnerName": "TIVIT",
  "vlan": 101,
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-fg9607vm",
  "lagId": "dxlag-fhccu14t",
  "connectionState": "ordering",
  "bandwidth": "500Mbps",
  "location": "TIVIT",
  "connectionName": "mydcinterconnect",
  "region": "sa-east-1"
}
```

- Para API obtener más información, consulte [AssociateHostedConnection](#) la Referencia de AWS CLI comandos.

## associate-virtual-interface

En el siguiente ejemplo de código, se muestra cómo usar `associate-virtual-interface`.

### AWS CLI

Para asociar una interfaz virtual a una conexión

El siguiente ejemplo asocia la interfaz virtual especificada a la especificada LAG. Como alternativa, para asociar la interfaz virtual a una conexión, especifique el ID de una conexión de AWS Direct Connect para `--connection-id`, por ejemplo, `dxcon-ffnikghc`.

Comando:

```
aws directconnect associate-virtual-interface --connection-id dxlag-ffjhj9lx --
virtual-interface-id dxvif-fgputw0j
```

Salida:

```
{
```

```

"virtualInterfaceState": "pending",
"asn": 65000,
"vlan": 123,
"customerAddress": "169.254.255.2/30",
"ownerAccount": "123456789012",
"connectionId": "dxlag-ffjhj9lx",
"addressFamily": "ipv4",
"virtualGatewayId": "vgw-38e90b51",
"virtualInterfaceId": "dxvif-fgputw0j",
"authKey": "0x123pK5_VBqv.UQ3kJ4123_",
"routeFilterPrefixes": [],
"location": "CSVA1",
"bgpPeers": [
  {
    "bgpStatus": "down",
    "customerAddress": "169.254.255.2/30",
    "addressFamily": "ipv4",
    "authKey": "0x123pK5_VBqv.UQ3kJ4123_",
    "bgpPeerState": "deleting",
    "amazonAddress": "169.254.255.1/30",
    "asn": 65000
  },
  {
    "bgpStatus": "down",
    "customerAddress": "169.254.255.2/30",
    "addressFamily": "ipv4",
    "authKey": "0x123pK5_VBqv.UQ3kJ4123_",
    "bgpPeerState": "pending",
    "amazonAddress": "169.254.255.1/30",
    "asn": 65000
  }
],
"customerRouterConfig": "<?xml version=\"1.0\" encoding=\"UTF-8\"?
>\n<logical_connection id=\"dxvif-fgputw0j\">\n  <vlan>123</vlan>
\n  <customer_address>169.254.255.2/30</customer_address>\n
  <amazon_address>169.254.255.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
  <bgp_auth_key>0x123pK5_VBqv.UQ3kJ4123_</bgp_auth_key>\n  <amazon_bgp_asn>7224</
amazon_bgp_asn>\n  <connection_type>private</connection_type>\n</logical_connection>
\n",
"amazonAddress": "169.254.255.1/30",
"virtualInterfaceType": "private",
"virtualInterfaceName": "VIF1A"
}

```



- Para API obtener más información, consulte [AssociateVirtualInterface](#) la Referencia de AWS CLI comandos.

## **confirm-connection**

En el siguiente ejemplo de código, se muestra cómo usar `confirm-connection`.

### AWS CLI

Para confirmar la creación de una conexión alojada en una interconexión

El siguiente `confirm-connection` comando confirma la creación de una conexión hospedada en una interconexión:

```
aws directconnect confirm-connection --connection-id dxcon-fg2wi7hy
```

Salida:

```
{
  "connectionState": "pending"
}
```

- Para API obtener más información, consulte [ConfirmConnection](#) la Referencia de AWS CLI comandos.

## **confirm-private-virtual-interface**

En el siguiente ejemplo de código, se muestra cómo usar `confirm-private-virtual-interface`.

### AWS CLI

Para aceptar la propiedad de una interfaz virtual privada

El siguiente `confirm-private-virtual-interface` comando acepta la propiedad de una interfaz virtual privada creada por otro cliente:

```
aws directconnect confirm-private-virtual-interface --virtual-interface-id dxvif-fgy8orxu --virtual-gateway-id vgw-e4a47df9
```

Salida:

```
{
  "virtualInterfaceState": "pending"
}
```

- Para API obtener más información, consulte [ConfirmPrivateVirtualInterface](#) la Referencia de AWS CLI comandos.

## confirm-public-virtual-interface

En el siguiente ejemplo de código, se muestra cómo usar `confirm-public-virtual-interface`.

### AWS CLI

Para aceptar la propiedad de una interfaz virtual pública

El siguiente `confirm-public-virtual-interface` comando acepta la propiedad de una interfaz virtual pública creada por otro cliente:

```
aws directconnect confirm-public-virtual-interface --virtual-interface-id dxvif-  
fg9xo9vp
```

Salida:

```
{
  "virtualInterfaceState": "verifying"
}
```

- Para API obtener más información, consulte [ConfirmPublicVirtualInterface](#) la Referencia de AWS CLI comandos.

## confirm-transit-virtual-interface

En el siguiente ejemplo de código, se muestra cómo usar `confirm-transit-virtual-interface`.

### AWS CLI

Para aceptar la propiedad de una interfaz virtual de tránsito

Lo siguiente `confirm-transit-virtual-interface` acepta la propiedad de una interfaz virtual de tránsito creada por otro cliente.

```
aws directconnect confirm-transit-virtual-interface \  
  --virtual-interface-id dxvif-fEXAMPLE \  
  --direct-connect-gateway-id 4112ccf9-25e9-4111-8237-b6c5dEXAMPLE
```

Salida:

```
{  
  "virtualInterfaceState": "pending"  
}
```

Para obtener más información, consulte [Aceptar una interfaz virtual alojada](#) en la Guía del usuario de AWS Direct Connect.

- Para API obtener más información, consulte [ConfirmTransitVirtualInterface](#) la Referencia de AWS CLI comandos.

## create-bgp-peer

En el siguiente ejemplo de código, se muestra cómo usar `create-bgp-peer`.

### AWS CLI

Para crear una sesión de IPv6 BGP intercambio de pares

El siguiente ejemplo crea una sesión de IPv6 BGP emparejamiento en una interfaz virtual privada. `dxvif-fg1vuj3d` Amazon asigna automáticamente IPv6 las direcciones homólogas.

Comando:

```
aws directconnect create-bgp-peer --virtual-interface-id dxvif-fg1vuj3d --new-bgp-peer asn=64600,addressFamily=ipv6
```

Salida:

```
{  
  "virtualInterface": {  
    "virtualInterfaceState": "available",  
    "asn": 65000,  
    "vlan": 125,  
  }  
}
```

```

"customerAddress": "169.254.255.2/30",
"ownerAccount": "123456789012",
"connectionId": "dxcon-fguhmq1c",
"addressFamily": "ipv4",
"virtualGatewayId": "vgw-f9eb0c90",
"virtualInterfaceId": "dxvif-fg1vuj3d",
"authKey": "0xC_ukbCer16EYA0example",
"routeFilterPrefixes": [],
"location": "EqDC2",
"bgpPeers": [
  {
    "bgpStatus": "down",
    "customerAddress": "169.254.255.2/30",
    "addressFamily": "ipv4",
    "authKey": "0xC_ukbCer16EYA0uexample",
    "bgpPeerState": "available",
    "amazonAddress": "169.254.255.1/30",
    "asn": 65000
  },
  {
    "bgpStatus": "down",
    "customerAddress": "2001:db8:1100:2f0:0:1:9cb4:4216/125",
    "addressFamily": "ipv6",
    "authKey": "0xS27kAIU_VHPjjAexample",
    "bgpPeerState": "pending",
    "amazonAddress": "2001:db8:1100:2f0:0:1:9cb4:4211/125",
    "asn": 64600
  }
],
"customerRouterConfig": "<?xml version=\"1.0\" encoding=
\"UTF-8\"?>\n<logical_connection id=\"dxvif-fg1vuj3d\">\n  <vlan>125</
vlan>\n  <customer_address>169.254.255.2/30</customer_address>\n
  <amazon_address>169.254.255.1/30</amazon_address>\n  <bgp_asn>65000</
bgp_asn>\n  <bgp_auth_key>0xC_ukbCer16EYA0uexample</bgp_auth_key>\n
  <ipv6_customer_address>2001:db8:1100:2f0:0:1:9cb4:4216/125</ipv6_customer_address>
\n  <ipv6_amazon_address>2001:db8:1100:2f0:0:1:9cb4:4211/125</ipv6_amazon_address>\n
  <ipv6_bgp_asn>64600</ipv6_bgp_asn>\n  <ipv6_bgp_auth_key>0xS27kAIU_VHPjjAexample</
ipv6_bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>\n
  <connection_type>private</connection_type>\n</logical_connection>\n",
  "amazonAddress": "169.254.255.1/30",
  "virtualInterfaceType": "private",
  "virtualInterfaceName": "Test"
}

```

```
}
```

- Para API obtener más información, consulte [CreateBgpPeer](#) la Referencia de AWS CLI comandos.

## create-connection

En el siguiente ejemplo de código, se muestra cómo usar `create-connection`.

### AWS CLI

Para crear una conexión desde la red a una ubicación de AWS Direct Connect

El siguiente `create-connection` comando crea una conexión desde la red a una ubicación de AWS Direct Connect:

```
aws directconnect create-connection --location TIVIT --bandwidth 1Gbps --connection-name "Connection to AWS"
```

Salida:

```
{
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-fg31dyv6",
  "connectionState": "requested",
  "bandwidth": "1Gbps",
  "location": "TIVIT",
  "connectionName": "Connection to AWS",
  "region": "sa-east-1"
}
```

- Para API obtener más información, consulte [CreateConnection](#) la Referencia de AWS CLI comandos.

## create-direct-connect-gateway-association-proposal

En el siguiente ejemplo de código, se muestra cómo usar `create-direct-connect-gateway-association-proposal`.

## AWS CLI

Para crear una propuesta para asociar la puerta de enlace de tránsito especificada con la puerta de enlace Direct Connect especificada

El siguiente `create-direct-connect-gateway-association-proposal` ejemplo crea una propuesta que asocia la puerta de enlace de tránsito especificada con la puerta de enlace Direct Connect especificada.

```
aws directconnect create-direct-connect-gateway-association-proposal \  
  --direct-connect-gateway-id 11460968-4ac1-4fd3-bdb2-00599EXAMPLE \  
  --direct-connect-gateway-owner-account 111122223333 \  
  --gateway-id tgw-02f776b1a7EXAMPLE \  
  --add-allowed-prefixes-to-direct-connect-gateway cidr=192.168.1.0/30
```

Salida:

```
{  
  "directConnectGatewayAssociationProposal": {  
    "proposalId": "cb7f41cb-8128-43a5-93b1-dcaedEXAMPLE",  
    "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",  
    "directConnectGatewayOwnerAccount": "111122223333",  
    "proposalState": "requested",  
    "associatedGateway": {  
      "id": "tgw-02f776b1a7EXAMPLE",  
      "type": "transitGateway",  
      "ownerAccount": "111122223333",  
      "region": "us-east-1"  
    },  
    "requestedAllowedPrefixesToDirectConnectGateway": [  
      {  
        "cidr": "192.168.1.0/30"  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [Creación de una propuesta de asociación de Transit Gateway](#) en la Guía del usuario de AWS Direct Connect.

- Para API obtener más información, consulte [CreateDirectConnectGatewayAssociationProposal](#) la Referencia de AWS CLI comandos.

## create-direct-connect-gateway-association

En el siguiente ejemplo de código, se muestra cómo usar `create-direct-connect-gateway-association`.

### AWS CLI

Para asociar una puerta de enlace privada virtual a una puerta de enlace Direct Connect

El siguiente ejemplo asocia la puerta de enlace privada virtual `vgw-6efe725e` con la puerta de enlace Direct Connect `5f294f92-bafb-4011-916d-9b0bexample`. Debe ejecutar el comando en la región en la que se encuentra la puerta de enlace privada virtual.

Comando:

```
aws directconnect create-direct-connect-gateway-association --direct-connect-gateway-id 5f294f92-bafb-4011-916d-9b0bexample --virtual-gateway-id vgw-6efe725e
```

Salida:

```
{
  "directConnectGatewayAssociation": {
    "associationState": "associating",
    "virtualGatewayOwnerAccount": "123456789012",
    "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
    "virtualGatewayId": "vgw-6efe725e",
    "virtualGatewayRegion": "us-east-2"
  }
}
```

- Para API obtener más información, consulte [CreateDirectConnectGatewayAssociation](#) la Referencia de AWS CLI comandos.

## create-direct-connect-gateway

En el siguiente ejemplo de código, se muestra cómo usar `create-direct-connect-gateway`.

### AWS CLI

Para crear una puerta de enlace Direct Connect

En el siguiente ejemplo, se crea una puerta de enlace Direct Connect con el nombre `DxGateway1`.

Comando:

```
aws directconnect create-direct-connect-gateway --direct-connect-gateway-name "DxGateway1"
```

Salida:

```
{
  "directConnectGateway": {
    "amazonSideAsn": 64512,
    "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bdexample",
    "ownerAccount": "123456789012",
    "directConnectGatewayName": "DxGateway1",
    "directConnectGatewayState": "available"
  }
}
```

- Para API obtener más información, consulte [CreateDirectConnectGateway](#) la Referencia de AWS CLI comandos.

## **create-interconnect**

En el siguiente ejemplo de código, se muestra cómo usar `create-interconnect`.

AWS CLI

Para crear una interconexión entre la red de un socio y AWS

El siguiente `create-interconnect` comando crea una interconexión entre la red de un socio de AWS Direct Connect y una ubicación específica de AWS Direct Connect:

```
aws directconnect create-interconnect --interconnect-name "1G Interconnect to AWS" --bandwidth 1Gbps --location TIVIT
```

Salida:

```
{
  "region": "sa-east-1",
```



```
"bandwidth": "1Gbps",
"location": "TIVIT",
"interconnectName": "1G Interconnect to AWS",
"interconnectId": "dxcon-fgktov66",
"interconnectState": "requested"
}
```

- Para API obtener más información, consulte [CreateInterconnect](#) la Referencia de AWS CLI comandos.

## create-lag

En el siguiente ejemplo de código, se muestra cómo usar `create-lag`.

### AWS CLI

Para crear un LAG con nuevas conexiones

El siguiente ejemplo crea LAG y solicita dos nuevas conexiones AWS Direct Connect LAG con un ancho de banda de 1 Gbps.

Comando:

```
aws directconnect create-lag --location CSVA1 --number-of-connections 2 --
connections-bandwidth 1Gbps --lag-name 1GBLag
```

Salida:

```
{
  "awsDevice": "CSVA1-23u8t1paz8iks",
  "numberOfConnections": 2,
  "lagState": "pending",
  "ownerAccount": "123456789012",
  "lagName": "1GBLag",
  "connections": [
    {
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-ffqr6x5q",
      "lagId": "dxlabel-ffjhj9lx",
      "connectionState": "requested",
      "bandwidth": "1Gbps",
      "location": "CSVA1",
```

```

        "connectionName": "Requested Connection 1 for Lag dxlag-ffjhj91x",
        "region": "us-east-1"
    },
    {
        "ownerAccount": "123456789012",
        "connectionId": "dxcon-fflqyj95",
        "lagId": "dxlag-ffjhj91x",
        "connectionState": "requested",
        "bandwidth": "1Gbps",
        "location": "CSVA1",
        "connectionName": "Requested Connection 2 for Lag dxlag-ffjhj91x",
        "region": "us-east-1"
    }
],
"lagId": "dxlag-ffjhj91x",
"minimumLinks": 0,
"connectionsBandwidth": "1Gbps",
"region": "us-east-1",
"location": "CSVA1"
}

```

Para crear una LAG utilizando una conexión existente

En el siguiente ejemplo, se crea una LAG a partir de una conexión existente en tu cuenta y se solicita una segunda conexión nueva LAG con el mismo ancho de banda y la misma ubicación que la conexión existente.

Comando:

```

aws directconnect create-lag --location EqDC2 --number-of-connections 2 --
connections-bandwidth 1Gbps --lag-name 2ConnLAG --connection-id dxcon-fgk145dr

```

Salida:

```

{
  "awsDevice": "EqDC2-4h6ce2r1bes6",
  "numberOfConnections": 2,
  "lagState": "pending",
  "ownerAccount": "123456789012",
  "lagName": "2ConnLAG",
  "connections": [
    {

```

```

    "ownerAccount": "123456789012",
    "connectionId": "dxcon-fh6ljcvo",
    "lagId": "dxlag-fhccu14t",
    "connectionState": "requested",
    "bandwidth": "1Gbps",
    "location": "EqDC2",
    "connectionName": "Requested Connection 1 for Lag dxlag-fhccu14t",
    "region": "us-east-1"
  },
  {
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-fgk145dr",
    "lagId": "dxlag-fhccu14t",
    "connectionState": "down",
    "bandwidth": "1Gbps",
    "location": "EqDC2",
    "connectionName": "VAConn1",
    "region": "us-east-1"
  }
],
"lagId": "dxlag-fhccu14t",
"minimumLinks": 0,
"connectionsBandwidth": "1Gbps",
"region": "us-east-1",
"location": "EqDC2"
}

```

- Para API obtener más información, consulte [CreateLag](#) la Referencia de AWS CLI comandos.

## create-private-virtual-interface

En el siguiente ejemplo de código, se muestra cómo usar `create-private-virtual-interface`.

### AWS CLI

Para crear una interfaz virtual privada

El siguiente `create-private-virtual-interface` comando crea una interfaz virtual privada:

```
aws directconnect create-private-virtual-interface --connection-id dxcon-ffjrkx17 --
new-private-virtual-
```

```
interface virtualInterfaceName=PrivateVirtualInterface,vlan=101,asn=65000,authKey=asdf34example37db6
```

Salida:

```
{
  "virtualInterfaceState": "pending",
  "asn": 65000,
  "vlan": 101,
  "customerAddress": "192.168.1.2/30",
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-ffjrnx17",
  "virtualGatewayId": "vgw-aba37db6",
  "virtualInterfaceId": "dxvif-ffhkh74f",
  "authKey": "asdf34example",
  "routeFilterPrefixes": [],
  "location": "TIVIT",
  "customerRouterConfig": "<?xml version=\"1.0\" encoding=
  \"UTF-8\"?>\n<logical_connection id=\"dxvif-ffhkh74f\">\n  <vlan>101</
  vlan>\n  <customer_address>192.168.1.2/30</customer_address>\n
  <amazon_address>192.168.1.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
  <bgp_auth_key>asdf34example</bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
  \n  <connection_type>private</connection_type>\n</logical_connection>\n",
  "amazonAddress": "192.168.1.1/30",
  "virtualInterfaceType": "private",
  "virtualInterfaceName": "PrivateVirtualInterface"
}
```

- Para API obtener más información, consulte [CreatePrivateVirtualInterface](#) la Referencia de AWS CLI comandos.

## create-public-virtual-interface

En el siguiente ejemplo de código, se muestra cómo usar `create-public-virtual-interface`.

### AWS CLI

Para crear una interfaz virtual pública

El siguiente `create-public-virtual-interface` comando crea una interfaz virtual pública:

```
aws directconnect create-public-virtual-interface --connection-id dxcon-ffjrkrx17 --
new-public-virtual-
interface virtualInterfaceName=PublicVirtualInterface,vlan=2000,asn=65000,authKey=asdf34exam
{cidr=203.0.113.4/30}]
```

Salida:

```
{
  "virtualInterfaceState": "verifying",
  "asn": 65000,
  "vlan": 2000,
  "customerAddress": "203.0.113.2/30",
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-ffjrkrx17",
  "virtualInterfaceId": "dxvif-fgh0hcrk",
  "authKey": "asdf34example",
  "routeFilterPrefixes": [
    {
      "cidr": "203.0.113.0/30"
    },
    {
      "cidr": "203.0.113.4/30"
    }
  ],
  "location": "TIVIT",
  "customerRouterConfig": "<?xml version=\"1.0\" encoding=\"UTF-8\"?
>\n<logical_connection id=\"dxvif-fgh0hcrk\">\n  <vlan>2000</
vlan>\n  <customer_address>203.0.113.2/30</customer_address>\n
  <amazon_address>203.0.113.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
  <bgp_auth_key>asdf34example</bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\n  <connection_type>public</connection_type>\n</logical_connection>\n",
  "amazonAddress": "203.0.113.1/30",
  "virtualInterfaceType": "public",
  "virtualInterfaceName": "PublicVirtualInterface"
}
```

- Para API obtener más información, consulte [CreatePublicVirtualInterface](#) la Referencia de AWS CLI comandos.

## create-transit-virtual-interface

En el siguiente ejemplo de código, se muestra cómo usar `create-transit-virtual-interface`.

### AWS CLI

Para crear una interfaz virtual de tránsito

El siguiente `create-transit-virtual-interface` ejemplo crea una interfaz virtual de tránsito para la conexión especificada.

```
aws directconnect create-transit-virtual-interface \
  --connection-id dxlag-fEXAMPLE \
  --new-transit-virtual-interface "virtualInterfaceName=Example Transit Virtual
  Interface,vlan=126,asn=65110,mtu=1500,authKey=0xzxcgA9YoW9h58u8SvEXAMPLE,amazonAddress=192.1
  aada-5a1baEXAMPLE,tags=[{key=Tag,value=Example}]"
```

Salida:

```
{
  "virtualInterface": {
    "ownerAccount": "1111222233333",
    "virtualInterfaceId": "dxvif-fEXAMPLE",
    "location": "loc1",
    "connectionId": "dxlag-fEXAMPLE",
    "virtualInterfaceType": "transit",
    "virtualInterfaceName": "Example Transit Virtual Interface",
    "vlan": 126,
    "asn": 65110,
    "amazonSideAsn": 4200000000,
    "authKey": "0xzxcgA9YoW9h58u8SEXAMPLE",
    "amazonAddress": "192.168.1.1/30",
    "customerAddress": "192.168.1.2/30",
    "addressFamily": "ipv4",
    "virtualInterfaceState": "pending",
    "customerRouterConfig": "<?xml version='1.0' encoding=
  \<UTF-8'>\n<logical_connection id='dxvif-fEXAMPLE'>\n  <vlan>126</
  vlan>\n  <customer_address>192.168.1.2/30</customer_address>\n
  <amazon_address>192.168.1.1/30</amazon_address>\n  <bgp_asn>65110</
  bgp_asn>\n  <bgp_auth_key>0xzxcgA9YoW9h58u8Sv0mXRTw</bgp_auth_key>\n
  <amazon_bgp_asn>4200000000</amazon_bgp_asn>\n  <connection_type>transit</
  connection_type>\n</logical_connection>\n",
```

```
"mtu": 1500,
"jumboFrameCapable": true,
"virtualGatewayId": "",
"directConnectGatewayId": "8384da05-13ce-4a91-aada-5a1baEXAMPLE",
"routeFilterPrefixes": [],
"bgpPeers": [
  {
    "bgpPeerId": "dxpeer-EXAMPLE",
    "asn": 65110,
    "authKey": "0xzxgA9YoW9h58u8SEXAMPLE",
    "addressFamily": "ipv4",
    "amazonAddress": "192.168.1.1/30",
    "customerAddress": "192.168.1.2/30",
    "bgpPeerState": "pending",
    "bgpStatus": "down",
    "awsDeviceV2": "loc1-26wz6vEXAMPLE"
  }
],
"region": "sa-east-1",
"awsDeviceV2": "loc1-26wz6vEXAMPLE",
"tags": [
  {
    "key": "Tag",
    "value": "Example"
  }
]
}
```

Para obtener más información, consulte [Creación de una interfaz virtual de tránsito para la puerta de enlace Direct Connect](#) en la Guía del usuario de AWS Direct Connect.

- Para API obtener más información, consulte [CreateTransitVirtualInterface](#) la Referencia de AWS CLI comandos.

## delete-bgp-peer

En el siguiente ejemplo de código, se muestra cómo usar `delete-bgp-peer`.

### AWS CLI

Para eliminar un BGP par de una interfaz virtual

En el siguiente ejemplo, se elimina el IPv6 BGP par de la interfaz `dxvif-fg1vuj3d` virtual.

Comando:

```
aws directconnect delete-bgp-peer --virtual-interface-id dxvif-fg1vuj3d --asn 64600
--customer-address 2001:db8:1100:2f0:0:1:9cb4:4216/125
```

Salida:

```
{
  "virtualInterface": {
    "virtualInterfaceState": "available",
    "asn": 65000,
    "vlan": 125,
    "customerAddress": "169.254.255.2/30",
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-fguhmq1c",
    "addressFamily": "ipv4",
    "virtualGatewayId": "vgw-f9eb0c90",
    "virtualInterfaceId": "dxvif-fg1vuj3d",
    "authKey": "0xC_ukbCer16EYA0example",
    "routeFilterPrefixes": [],
    "location": "EqDC2",
    "bgpPeers": [
      {
        "bgpStatus": "down",
        "customerAddress": "169.254.255.2/30",
        "addressFamily": "ipv4",
        "authKey": "0xC_ukbCer16EYA0uexample",
        "bgpPeerState": "available",
        "amazonAddress": "169.254.255.1/30",
        "asn": 65000
      },
      {
        "bgpStatus": "down",
        "customerAddress": "2001:db8:1100:2f0:0:1:9cb4:4216/125",
        "addressFamily": "ipv6",
        "authKey": "0xS27kAIU_VHPjjAexample",
        "bgpPeerState": "deleting",
        "amazonAddress": "2001:db8:1100:2f0:0:1:9cb4:4211/125",
        "asn": 64600
      }
    ]
  },
}
```



```

    "customerRouterConfig": "<?xml version=\"1.0\" encoding=
    \<UTF-8\"?>\n<logical_connection id=\"dxvif-fg1vuj3d\">\n  <vlan>125</
    vlan>\n  <customer_address>169.254.255.2/30</customer_address>\n
    <amazon_address>169.254.255.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
    <bgp_auth_key>0xC_ukbCer16EYA0example</bgp_auth_key>\n  <amazon_bgp_asn>7224</
    amazon_bgp_asn>\n  <connection_type>private</connection_type>\n</logical_connection>
    \n",
    "amazonAddress": "169.254.255.1/30",
    "virtualInterfaceType": "private",
    "virtualInterfaceName": "Test"
  }
}

```

- Para API obtener más información, consulte [DeleteBgpPeer](#) la Referencia de AWS CLI comandos.

## delete-connection

En el siguiente ejemplo de código, se muestra cómo usar `delete-connection`.

### AWS CLI

Para eliminar una conexión

El siguiente `delete-connection` comando elimina la conexión especificada:

```
aws directconnect delete-connection --connection-id dxcon-fg31dyv6
```

Salida:

```

{
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-fg31dyv6",
  "connectionState": "deleted",
  "bandwidth": "1Gbps",
  "location": "TIVIT",
  "connectionName": "Connection to AWS",
  "region": "sa-east-1"
}

```

- Para API obtener más información, consulte [DeleteConnection](#) la Referencia de AWS CLI comandos.

## delete-direct-connect-gateway-association

En el siguiente ejemplo de código, se muestra cómo usar `delete-direct-connect-gateway-association`.

### AWS CLI

Para eliminar una asociación de puerta de enlace de Direct Connect

El siguiente `delete-direct-connect-gateway-association` ejemplo elimina la asociación de la puerta de enlace Direct Connect con una puerta de enlace de tránsito que tiene el ID de asociación especificado.

```
aws directconnect delete-direct-connect-gateway-association --association-id
be85116d-46eb-4b43-a27a-da0c2ad648de
```

Salida:

```
{
  "directConnectGatewayAssociation": {
    "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
    "directConnectGatewayOwnerAccount": "123456789012",
    "associationState": "disassociating",
    "associatedGateway": {
      "id": "tgw-095b3b0b54EXAMPLE",
      "type": "transitGateway",
      "ownerAccount": "123456789012",
      "region": "us-east-1"
    },
    "associationId": " be85116d-46eb-4b43-a27a-da0c2ad648deEXAMPLE ",
    "allowedPrefixesToDirectConnectGateway": [
      {
        "cidr": "192.0.1.0/28"
      }
    ]
  }
}
```

Para obtener más información, consulte [Asociación y desasociación de pasarelas de tránsito en la Guía del usuario](#) de Direct AWS Connect.

- Para obtener API más información, consulte la Referencia de comandos [DeleteDirectConnectGatewayAssociation](#).AWS CLI

## delete-direct-connect-gateway

En el siguiente ejemplo de código, se muestra cómo usar `delete-direct-connect-gateway`.

### AWS CLI

Para eliminar una puerta de enlace de Direct Connect

En el siguiente ejemplo, se elimina la puerta `5f294f92-bafb-4011-916d-9b0bexample` de enlace Direct Connect.

Comando:

```
aws directconnect delete-direct-connect-gateway --direct-connect-gateway-id 5f294f92-bafb-4011-916d-9b0bexample
```

Salida:

```
{
  "directConnectGateway": {
    "amazonSideAsn": 64512,
    "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
    "ownerAccount": "123456789012",
    "directConnectGatewayName": "DxGateway1",
    "directConnectGatewayState": "deleting"
  }
}
```

- Para API obtener más información, consulte [DeleteDirectConnectGateway](#) la Referencia de AWS CLI comandos.

## delete-interconnect

En el siguiente ejemplo de código, se muestra cómo usar `delete-interconnect`.

### AWS CLI

Para eliminar una interconexión

El siguiente `delete-interconnect` comando elimina la interconexión especificada:

```
aws directconnect delete-interconnect --interconnect-id dxcon-fgktov66
```

Salida:

```
{
  "interconnectState": "deleted"
}
```

- Para API obtener más información, consulte la Referencia [DeleteInterconnect](#) de AWS CLI comandos.

## delete-lag

En el siguiente ejemplo de código, se muestra cómo usar `delete-lag`.

AWS CLI

Para eliminar un LAG

En el siguiente ejemplo, se elimina lo especificado LAG.

Comando:

```
aws directconnect delete-lag --lag-id dxlag-ffrhowd9
```

Salida:

```
{
  "awsDevice": "EqDC2-4h6ce2r1bes6",
  "numberOfConnections": 0,
  "lagState": "deleted",
  "ownerAccount": "123456789012",
  "lagName": "TestLAG",
  "connections": [],
  "lagId": "dxlag-ffrhowd9",
  "minimumLinks": 0,
  "connectionsBandwidth": "1Gbps",
  "region": "us-east-1",
  "location": "EqDC2"
}
```

- Para API obtener más información, consulte [DeleteLag](#) la Referencia de AWS CLI comandos.

## delete-virtual-interface

En el siguiente ejemplo de código, se muestra cómo usar `delete-virtual-interface`.

### AWS CLI

Para eliminar una interfaz virtual

El siguiente `delete-virtual-interface` comando elimina la interfaz virtual especificada:

```
aws directconnect delete-virtual-interface --virtual-interface-id dxvif-ffhkh74f
```

Salida:

```
{  
  "virtualInterfaceState": "deleting"  
}
```

- Para API obtener más información, consulte [DeleteVirtualInterface](#) la Referencia de AWS CLI comandos.

## describe-connection-loa

En el siguiente ejemplo de código, se muestra cómo usar `describe-connection-loa`.

### AWS CLI

Para describir suLOA: CFA para una conexión mediante Linux o Mac OS X

En el siguiente ejemplo se describe su conexión LOA - CFA `fordxcon-fh6ayh1d`. El contenido de LOA - CFA está codificado en base64. Este comando usa los `--query` parámetros `--output` y para controlar la salida y extraer el contenido de la estructura. `loaContent` La parte final del comando decodifica el contenido mediante la `base64` utilidad y envía el resultado a un PDF archivo.

```
aws directconnect describe-connection-loa --connection-id dxcon-fh6ayh1d --  
output text --query loa.loaContent|base64 --decode > myLoaCfa.pdf
```

Para describir suLOA: CFA para una conexión mediante Windows

El ejemplo anterior requiere el uso de la base64 utilidad para decodificar la salida. En una computadora con Windows, puede usarla `certutil` en su lugar. En el ejemplo siguiente, el primer comando describe la conexión LOA - CFA for `dxcon-fh6ayh1d` y utiliza los `--query` parámetros `--output` y para controlar la salida y extraer el contenido de la `loaContent` estructura en un archivo llamado `myLoaCfa.base64`. El segundo comando utiliza la `certutil` utilidad para decodificar el archivo y enviar el resultado a un PDF archivo.

```
aws directconnect describe-connection-loa --connection-id dxcon-fh6ayh1d --  
output text --query loa.loaContent > myLoaCfa.base64
```

```
certutil -decode myLoaCfa.base64 myLoaCfa.pdf
```

Para obtener más información sobre el control de la AWS CLI salida, consulte [Control de la salida de comandos desde la interfaz de línea de AWS comandos](#) en la Guía del usuario de la interfaz de línea de AWS comandos.

- Para API obtener más información, consulte [DescribeConnectionLoa](#) la Referencia de AWS CLI comandos.

## describe-connections-on-interconnect

En el siguiente ejemplo de código, se muestra cómo usar `describe-connections-on-interconnect`.

### AWS CLI

Para enumerar las conexiones de una interconexión

El siguiente `describe-connections-on-interconnect` comando muestra las conexiones que se han aprovisionado en la interconexión dada:

```
aws directconnect describe-connections-on-interconnect --interconnect-id dxcon-  
fgktov66
```

Salida:

```
{  
  "connections": [  
    {
```

```
    "partnerName": "TIVIT",
    "vlan": 101,
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-ffzc51m1",
    "connectionState": "ordering",
    "bandwidth": "500Mbps",
    "location": "TIVIT",
    "connectionName": "mydcinterconnect",
    "region": "sa-east-1"
  }
]
```

- Para API obtener más información, consulte la Referencia [DescribeConnectionsOnInterconnect](#) de AWS CLI comandos.

## describe-connections

En el siguiente ejemplo de código, se muestra cómo usar `describe-connections`.

### AWS CLI

Para ver una lista de todas las conexiones de la región actual

El siguiente `describe-connections` comando muestra todas las conexiones de la región actual:

```
aws directconnect describe-connections
```

Salida:

```
{
  "connections": [
    {
      "awsDevice": "EqDC2-123h49s71dabc",
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-fguhmq1c",
      "lagId": "dxlag-ffrz71kw",
      "connectionState": "down",
      "bandwidth": "1Gbps",
      "location": "EqDC2",
      "connectionName": "My_Connection",
```

```
        "loaIssueTime": 1491568964.0,  
        "region": "us-east-1"  
    }  
]  
}
```

- Para API obtener más información, consulte [DescribeConnections](#) la Referencia de AWS CLI comandos.

## describe-direct-connect-gateway-association-proposals

En el siguiente ejemplo de código, se muestra cómo usar `describe-direct-connect-gateway-association-proposals`.

### AWS CLI

Para describir sus propuestas de asociación de pasarelas Direct Connect

En el siguiente `describe-direct-connect-gateway-association-proposals` ejemplo, se muestran detalles sobre sus propuestas de asociación de pasarelas de Direct Connect.

```
aws directconnect describe-direct-connect-gateway-association-proposals
```

Salida:

```
{  
  "directConnectGatewayAssociationProposals": [  
    {  
      "proposalId": "c2ede9b4-bbc6-4d33-923c-bc4feEXAMPLE",  
      "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",  
      "directConnectGatewayOwnerAccount": "111122223333",  
      "proposalState": "requested",  
      "associatedGateway": {  
        "id": "tgw-02f776b1a7EXAMPLE",  
        "type": "transitGateway",  
        "ownerAccount": "111122223333",  
        "region": "us-east-1"  
      },  
      "existingAllowedPrefixesToDirectConnectGateway": [  
        {  
          "cidr": "192.168.2.0/30"  
        }  
      ]  
    }  
  ]  
}
```



```

        },
        {
            "cidr": "192.168.1.0/30"
        }
    ],
    "requestedAllowedPrefixesToDirectConnectGateway": [
        {
            "cidr": "192.168.1.0/30"
        }
    ]
},
{
    "proposalId": "cb7f41cb-8128-43a5-93b1-dcaedEXAMPLE",
    "directConnectGatewayId": "11560968-4ac1-4fd3-bcb2-00599EXAMPLE",
    "directConnectGatewayOwnerAccount": "111122223333",
    "proposalState": "accepted",
    "associatedGateway": {
        "id": "tgw-045776b1a7EXAMPLE",
        "type": "transitGateway",
        "ownerAccount": "111122223333",
        "region": "us-east-1"
    },
    "existingAllowedPrefixesToDirectConnectGateway": [
        {
            "cidr": "192.168.4.0/30"
        },
        {
            "cidr": "192.168.5.0/30"
        }
    ],
    "requestedAllowedPrefixesToDirectConnectGateway": [
        {
            "cidr": "192.168.5.0/30"
        }
    ]
}
]
}

```

Para obtener más información, consulte [Asociación y desasociación de pasarelas de tránsito en la Guía del usuario](#) de Direct AWS Connect.

- Para obtener API más información, consulte la Referencia de comandos [DescribeDirectConnectGatewayAssociationProposals](#).AWS CLI

## describe-direct-connect-gateway-associations

En el siguiente ejemplo de código, se muestra cómo usar `describe-direct-connect-gateway-associations`.

### AWS CLI

Para describir las asociaciones de pasarelas de Direct Connect

El siguiente ejemplo describe todas las asociaciones con la puerta de enlace Direct Connect `5f294f92-bafb-4011-916d-9b0bexample`.

Comando:

```
aws directconnect describe-direct-connect-gateway-associations --direct-connect-gateway-id 5f294f92-bafb-4011-916d-9b0bexample
```

Salida:

```
{
  "nextToken":
  "eyJ2IjoxLCJzIjoxLCJpIjoiOU83OTFodzdycnZCbkN4MEExHeHVwQT09IiwieyYyI6InIwIiwiaXN0UEVHVH0I1UF1kaWFnNl",
  "directConnectGatewayAssociations": [
    {
      "associationState": "associating",
      "virtualGatewayOwnerAccount": "123456789012",
      "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
      "virtualGatewayId": "vgw-6efe725e",
      "virtualGatewayRegion": "us-east-2"
    },
    {
      "associationState": "disassociating",
      "virtualGatewayOwnerAccount": "123456789012",
      "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
      "virtualGatewayId": "vgw-ebaa27db",
      "virtualGatewayRegion": "us-east-2"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeDirectConnectGatewayAssociations](#) la Referencia de AWS CLI comandos.

## describe-direct-connect-gateway-attachments

En el siguiente ejemplo de código, se muestra cómo usar `describe-direct-connect-gateway-attachments`.

### AWS CLI

Para describir los adjuntos de la puerta de enlace Direct Connect

El siguiente ejemplo describe las interfaces virtuales que están conectadas a la puerta de enlace Direct Connect `5f294f92-bafb-4011-916d-9b0bexample`.

Comando:

```
aws directconnect describe-direct-connect-gateway-attachments --direct-connect-gateway-id 5f294f92-bafb-4011-916d-9b0bexample
```

Salida:

```
{
  "directConnectGatewayAttachments": [
    {
      "virtualInterfaceOwnerAccount": "123456789012",
      "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
      "virtualInterfaceRegion": "us-east-2",
      "attachmentState": "attaching",
      "virtualInterfaceId": "dxvif-fg9zyabc"
    }
  ],
  "nextToken":
  "eyJ2IjoxLCJzIjoxLCJpIjoibEhXd1NpUXF5RzhoL1JyUW52S1V2QT09IiwieYyI6Im5wQjFHQ0RyQUdRS3puNnNXcU"
}
```

- Para API obtener más información, consulte [DescribeDirectConnectGatewayAttachments](#) la Referencia de AWS CLI comandos.

## describe-direct-connect-gateways

En el siguiente ejemplo de código, se muestra cómo usar `describe-direct-connect-gateways`.

## AWS CLI

Para describir sus pasarelas de Direct Connect

El siguiente ejemplo describe todas las pasarelas de Direct Connect.

Comando:

```
aws directconnect describe-direct-connect-gateways
```

Salida:

```
{
  "directConnectGateways": [
    {
      "amazonSideAsn": 64512,
      "directConnectGatewayId": "cf68415c-f4ae-48f2-87a7-3b52cexample",
      "ownerAccount": "123456789012",
      "directConnectGatewayName": "DxGateway2",
      "directConnectGatewayState": "available"
    },
    {
      "amazonSideAsn": 64512,
      "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bdexample",
      "ownerAccount": "123456789012",
      "directConnectGatewayName": "DxGateway1",
      "directConnectGatewayState": "available"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeDirectConnectGateways](#) la Referencia de AWS CLI comandos.

## describe-hosted-connections

En el siguiente ejemplo de código, se muestra cómo usar `describe-hosted-connections`.

## AWS CLI

Para enumerar las conexiones de una interconexión

En el siguiente ejemplo, se enumeran las conexiones que se han aprovisionado en la interconexión dada.

Comando:

```
aws directconnect describe-hosted-connections --connection-id dxcon-fgktov66
```

Salida:

```
{
  "connections": [
    {
      "partnerName": "TIVIT",
      "vlan": 101,
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-ffzc51m1",
      "connectionState": "ordering",
      "bandwidth": "500Mbps",
      "location": "TIVIT",
      "connectionName": "mydcinterconnect",
      "region": "sa-east-1"
    }
  ]
}
```

- Para API obtener más información, consulte la Referencia [DescribeHostedConnections](#) de AWS CLI comandos.

## **describe-interconnect-loa**

En el siguiente ejemplo de código, se muestra cómo usar `describe-interconnect-loa`.

AWS CLI

Para describir su LOA... CFA para una interconexión mediante Linux o Mac OS X

El siguiente ejemplo describe su interconexión LOA `dxcon-fh6ayh1d` - CFA for. El contenido de LOA - CFA está codificado en base64. Este comando usa los `--query` parámetros `--output` y para controlar la salida y extraer el contenido de la estructura. `loaContent` La parte final del comando decodifica el contenido mediante la `base64` utilidad y envía el resultado a un PDF archivo.

```
aws directconnect describe-interconnect-loa --interconnect-id dxcon-fh6ayh1d --  
output text --query Loa.LoaContent|base64 --decode > myLoaCfa.pdf
```

Para describir suLOA... CFA para una interconexión mediante Windows

El ejemplo anterior requiere el uso de la base64 utilidad para decodificar la salida. En una computadora con Windows, puede usarla `certutil` en su lugar. En el ejemplo siguiente, el primer comando describe la interconexión LOA - CFA for `dxcon-fh6ayh1d` y utiliza los `--query` parámetros `--output` y para controlar la salida y extraer el contenido de la `LoaContent` estructura en un archivo llamado `myLoaCfa.base64`. El segundo comando utiliza la `certutil` utilidad para decodificar el archivo y enviar el resultado a un PDF archivo.

```
aws directconnect describe-interconnect-loa --interconnect-id dxcon-fh6ayh1d --  
output text --query Loa.LoaContent > myLoaCfa.base64
```

```
certutil -decode myLoaCfa.base64 myLoaCfa.pdf
```

Para obtener más información sobre el control de la AWS CLI salida, consulte [Control de la salida de comandos desde la interfaz de línea de AWS comandos](#) en la Guía del usuario de la interfaz de línea de AWS comandos.

- Para API obtener más información, consulte [DescribeInterconnectLoa](#) la Referencia de AWS CLI comandos.

## describe-interconnects

En el siguiente ejemplo de código, se muestra cómo usar `describe-interconnects`.

### AWS CLI

Para enumerar las interconexiones

El siguiente `describe-interconnects` comando muestra las interconexiones que pertenecen a su AWS cuenta:

```
aws directconnect describe-interconnects
```

Salida:

```
{
  "interconnects": [
    {
      "region": "sa-east-1",
      "bandwidth": "1Gbps",
      "location": "TIVIT",
      "interconnectName": "1G Interconnect to AWS",
      "interconnectId": "dxcon-fgktov66",
      "interconnectState": "down"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeInterconnects](#) la Referencia de AWS CLI comandos.

## describe-lags

En el siguiente ejemplo de código, se muestra cómo usar `describe-lags`.

### AWS CLI

Para describir su LAGs

El siguiente comando describe todos los LAGs de la región actual.

Comando:

```
aws directconnect describe-lags
```

Salida:

```
{
  "lags": [
    {
      "awsDevice": "EqDC2-19y7z3m17xpuz",
      "numberOfConnections": 2,
      "lagState": "down",
      "ownerAccount": "123456789012",
      "lagName": "DA-LAG",
      "connections": [
```

```

    {
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-ffnikghc",
      "lagId": "dxlag-fgsu9erb",
      "connectionState": "requested",
      "bandwidth": "10Gbps",
      "location": "EqDC2",
      "connectionName": "Requested Connection 1 for Lag dxlag-fgsu9erb",
      "region": "us-east-1"
    },
    {
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-fglgbdea",
      "lagId": "dxlag-fgsu9erb",
      "connectionState": "requested",
      "bandwidth": "10Gbps",
      "location": "EqDC2",
      "connectionName": "Requested Connection 2 for Lag dxlag-fgsu9erb",
      "region": "us-east-1"
    }
  ],
  "lagId": "dxlag-fgsu9erb",
  "minimumLinks": 0,
  "connectionsBandwidth": "10Gbps",
  "region": "us-east-1",
  "location": "EqDC2"
}
]
}

```

- Para API obtener más información, consulte [DescribeLags](#) la Referencia de AWS CLI comandos.

## describe-loa

En el siguiente ejemplo de código, se muestra cómo usar `describe-loa`.

### AWS CLI

Para describir suLOA: CFA para una conexión mediante Linux o Mac OS X

En el siguiente ejemplo se describe su conexión LOA - CFA `fordxcon-fh6ayh1d`. El contenido de LOA - CFA está codificado en base64. Este comando usa los `--query` parámetros `--output`



y para controlar la salida y extraer el contenido de la estructura. `loaContent` La parte final del comando decodifica el contenido mediante la `base64` utilidad y envía el resultado a un PDF archivo.

```
aws directconnect describe-loa --connection-id dxcon-fh6ayh1d --output text --  
query Loa.LoaContent|base64 --decode > myLoaCfa.pdf
```

Para describir suLOA: CFA para una conexión mediante Windows

El ejemplo anterior requiere el uso de la `base64` utilidad para decodificar la salida. En una computadora con Windows, puede usarla `certutil` en su lugar. En el ejemplo siguiente, el primer comando describe la conexión LOA - CFA for `dxcon-fh6ayh1d` y utiliza los `--query` parámetros `--output` y para controlar la salida y extraer el contenido de la `loaContent` estructura en un archivo llamado `myLoaCfa.base64`. El segundo comando utiliza la `certutil` utilidad para decodificar el archivo y enviar el resultado a un PDF archivo.

```
aws directconnect describe-loa --connection-id dxcon-fh6ayh1d --output text --  
query Loa.LoaContent > myLoaCfa.base64
```

```
certutil -decode myLoaCfa.base64 myLoaCfa.pdf
```

Para obtener más información sobre el control de la AWS CLI salida, consulte [Control de la salida de comandos desde la interfaz de línea de AWS comandos](#) en la Guía del usuario de la interfaz de línea de AWS comandos.

- Para API obtener más información, consulte [DescribeLoa](#) la Referencia de AWS CLI comandos.

## describe-locations

En el siguiente ejemplo de código, se muestra cómo usar `describe-locations`.

### AWS CLI

Para enumerar los socios y las ubicaciones de AWS Direct Connect

El siguiente `describe-locations` comando muestra los socios y las ubicaciones de AWS Direct Connect en la región actual:

```
aws directconnect describe-locations
```

Salida:

```
{
  "locations": [
    {
      "locationName": "NAP do Brasil, Barueri, Sao Paulo",
      "locationCode": "TNDB"
    },
    {
      "locationName": "Tivit - Site Transamerica (Sao Paulo)",
      "locationCode": "TIVIT"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeLocations](#) la Referencia de AWS CLI comandos.

## describe-tags

En el siguiente ejemplo de código, se muestra cómo usar `describe-tags`.

AWS CLI

Para describir las etiquetas de sus recursos de AWS Direct Connect

El siguiente comando describe las etiquetas de la conexión `dxcon-abcabc12`.

Comando:

```
aws directconnect describe-tags --resource-arns arn:aws:directconnect:us-east-1:123456789012:dxcon/dxcon-abcabc12
```

Salida:

```
{
  "resourceTags": [
    {
      "resourceArn": "arn:aws:directconnect:us-east-1:123456789012:dxcon/dxcon-abcabc12",
      "tags": [
        {
```

```

        "value": "VAConnection",
        "key": "Name"
      }
    ]
  }
}

```

- Para API obtener más información, consulte [DescribeTags](#) la Referencia de AWS CLI comandos.

## describe-virtual-gateways

En el siguiente ejemplo de código, se muestra cómo usar `describe-virtual-gateways`.

### AWS CLI

Para enumerar las puertas de enlace privadas virtuales

El siguiente `describe-virtual-gateways` comando muestra las puertas de enlace privadas virtuales que pertenecen a su AWS cuenta:

```
aws directconnect describe-virtual-gateways
```

Salida:

```

{
  "virtualGateways": [
    {
      "virtualGatewayId": "vgw-aba37db6",
      "virtualGatewayState": "available"
    }
  ]
}

```

- Para API obtener más información, consulte [DescribeVirtualGateways](#) la Referencia de AWS CLI comandos.

## describe-virtual-interfaces

En el siguiente ejemplo de código, se muestra cómo usar `describe-virtual-interfaces`.

## AWS CLI

Para enumerar todas las interfaces virtuales

El siguiente describe-virtual-interfaces comando muestra la información sobre todas las interfaces virtuales asociadas a su AWS cuenta:

```
aws directconnect describe-virtual-interfaces --connection-id dxcon-ffjrnx17
```

Salida:

```
{
  "virtualInterfaces": [
    {
      "virtualInterfaceState": "down",
      "asn": 65000,
      "vlan": 101,
      "customerAddress": "192.168.1.2/30",
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-ffjrnx17",
      "virtualGatewayId": "vgw-aba37db6",
      "virtualInterfaceId": "dxvif-ffhkh74f",
      "authKey": "asdf34example",
      "routeFilterPrefixes": [],
      "location": "TIVIT",
      "customerRouterConfig": "<?xml version=\"1.0\" encoding=
\\\"UTF-8\\\"?>\\n<logical_connection id=\\\"dxvif-ffhkh74f\\\">\\n  <vlan>101</
vlan>\\n  <customer_address>192.168.1.2/30</customer_address>\\n
  <amazon_address>192.168.1.1/30</amazon_address>\\n  <bgp_asn>65000</bgp_asn>\\n
  <bgp_auth_key>asdf34example</bgp_auth_key>\\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\\n  <connection_type>private</connection_type>\\n</logical_connection>\\n",
      "amazonAddress": "192.168.1.1/30",
      "virtualInterfaceType": "private",
      "virtualInterfaceName": "PrivateVirtualInterface"
    },
    {
      "virtualInterfaceState": "verifying",
      "asn": 65000,
      "vlan": 2000,
      "customerAddress": "203.0.113.2/30",
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-ffjrnx17",
      "virtualGatewayId": "",

```

```

    "virtualInterfaceId": "dxvif-fgh0hcrk",
    "authKey": "asdf34example",
    "routeFilterPrefixes": [
      {
        "cidr": "203.0.113.4/30"
      },
      {
        "cidr": "203.0.113.0/30"
      }
    ],
    "location": "TIVIT",
    "customerRouterConfig": "<?xml version=\"1.0\" encoding=
UTF-8\"?>\n<logical_connection id=\"dxvif-fgh0hcrk\">\n  <vlan>2000</
vlan>\n  <customer_address>203.0.113.2/30</customer_address>\n
  <amazon_address>203.0.113.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
  <bgp_auth_key>asdf34example</bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\n  <connection_type>public</connection_type>\n</logical_connection>\n",
    "amazonAddress": "203.0.113.1/30",
    "virtualInterfaceType": "public",
    "virtualInterfaceName": "PublicVirtualInterface"
  }
]
}

```

- Para API obtener más información, consulte [DescribeVirtualInterfaces](#) la Referencia de AWS CLI comandos.

## disassociate-connection-from-lag

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-connection-from-lag`.

### AWS CLI

Para desasociar una conexión de un LAG

El siguiente ejemplo desasocia la conexión especificada de la especificada. LAG

Comando:

```
aws directconnect disassociate-connection-from-lag --lag-id dxlag-fhccu14t --
connection-id dxcon-fg9607vm
```

Salida:

```
{
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-fg9607vm",
  "connectionState": "requested",
  "bandwidth": "1Gbps",
  "location": "EqDC2",
  "connectionName": "Con2ForLag",
  "region": "us-east-1"
}
```

- Para API obtener más información, consulte [DisassociateConnectionFromLag](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para agregar una etiqueta a un recurso de AWS Direct Connect

El siguiente comando agrega una etiqueta con una clave `Name` y un valor de `VACconnection` a la conexión `dxcon-abcabc12`. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws directconnect tag-resource --resource-arn arn:aws:directconnect:us-east-1:123456789012:dxcon/dxcon-abcabc12 --tags "key=Name,value=VACconnection"
```

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

## AWS CLI

Para eliminar una etiqueta de un recurso de AWS Direct Connect

El siguiente comando elimina la etiqueta con la clave Name de la conexión `dxcon-abcabc12`. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws directconnect untag-resource --resource-arn arn:aws:directconnect:us-east-1:123456789012:dxcon/dxcon-abcabc12 --tag-keys Name
```

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-direct-connect-gateway-association

En el siguiente ejemplo de código, se muestra cómo usar `update-direct-connect-gateway-association`.

### AWS CLI

Para actualizar los atributos especificados de la asociación de pasarelas de Direct Connect

En el siguiente `update-direct-connect-gateway-association` ejemplo, se agrega el CIDR bloque especificado a una asociación de puerta de enlace de Direct Connect.

```
aws directconnect update-direct-connect-gateway-association \  
  --association-id 820a6e4f-5374-4004-8317-3f64bEXAMPLE \  
  --add-allowed-prefixes-to-direct-connect-gateway cidr=192.168.2.0/30
```

Salida:

```
{  
  "directConnectGatewayAssociation": {  
    "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",  
    "directConnectGatewayOwnerAccount": "111122223333",  
    "associationState": "updating",  
    "associatedGateway": {  
      "id": "tgw-02f776b1a7EXAMPLE",  
      "type": "transitGateway",
```

```
    "ownerAccount": "111122223333",
    "region": "us-east-1"
  },
  "associationId": "820a6e4f-5374-4004-8317-3f64bEXAMPLE",
  "allowedPrefixesToDirectConnectGateway": [
    {
      "cidr": "192.168.2.0/30"
    },
    {
      "cidr": "192.168.1.0/30"
    }
  ]
}
```

Para obtener más información, consulte [Uso de pasarelas de Direct Connect](#) en la Guía del usuario de AWS Direct Connect.

- Para API obtener más información, consulte [UpdateDirectConnectGatewayAssociation](#) la Referencia de AWS CLI comandos.

## update-lag

En el siguiente ejemplo de código, se muestra cómo usar `update-lag`.

### AWS CLI

Para actualizar un LAG

En el siguiente ejemplo, se cambia el nombre del elemento especificado LAG.

Comando:

```
aws directconnect update-lag --lag-id dxlag-ffjhj9lx --lag-name 2ConnLag
```

Salida:

```
{
  "awsDevice": "CSVA1-23u8t1paz8iks",
  "numberOfConnections": 2,
  "lagState": "down",
  "ownerAccount": "123456789012",
```



```

"lagName": "2ConnLag",
"connections": [
  {
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-fflqyj95",
    "lagId": "dxlag-ffjhj9lx",
    "connectionState": "requested",
    "bandwidth": "1Gbps",
    "location": "CSVA1",
    "connectionName": "Requested Connection 2 for Lag dxlag-ffjhj9lx",
    "region": "us-east-1"
  },
  {
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-ffqr6x5q",
    "lagId": "dxlag-ffjhj9lx",
    "connectionState": "requested",
    "bandwidth": "1Gbps",
    "location": "CSVA1",
    "connectionName": "Requested Connection 1 for Lag dxlag-ffjhj9lx",
    "region": "us-east-1"
  }
],
"lagId": "dxlag-ffjhj9lx",
"minimumLinks": 0,
"connectionsBandwidth": "1Gbps",
"region": "us-east-1",
"location": "CSVA1"
}

```

- Para API obtener más información, consulte [UpdateLag](#) la Referencia de AWS CLI comandos.

## update-virtual-interface-attributes

En el siguiente ejemplo de código, se muestra cómo usar `update-virtual-interface-attributes`.

### AWS CLI

Para actualizar MTU la interfaz virtual

El siguiente `update-virtual-interface-attributes` ejemplo actualiza MTU la interfaz virtual especificada.

```
aws directconnect update-virtual-interface-attributes \
  --virtual-interface-id dxvif-fEXAMPLE \
  --mtu 1500
```

Salida:

```
{
  "ownerAccount": "1111222233333",
  "virtualInterfaceId": "dxvif-fEXAMPLE",
  "location": "loc1",
  "connectionId": "dxlag-fEXAMPLE",
  "virtualInterfaceType": "transit",
  "virtualInterfaceName": "example transit virtual interface",
  "vlan": 125,
  "asn": 650001,
  "amazonSideAsn": 64512,
  "authKey": "0xzxgA9YoW9h58u8SEXAMPLE",
  "amazonAddress": "169.254.248.1/30",
  "customerAddress": "169.254.248.2/30",
  "addressFamily": "ipv4",
  "virtualInterfaceState": "down",
  "customerRouterConfig": "<?xml version='1.0' encoding='UTF-8'?>\n<logical_connection id='dxvif-fEXAMPLE'>\n  <vlan>125</vlan>\n  \n  <customer_address>169.254.248.2/30</customer_address>\n  \n  <amazon_address>169.254.248.1/30</amazon_address>\n  <bgp_asn>650001</bgp_asn>\n  <bgp_auth_key>0xzxgA9YoW9h58u8SEXAMPLE</bgp_auth_key>\n  <amazon_bgp_asn>64512</amazon_bgp_asn>\n  <connection_type>transit</connection_type>\n</logical_connection>\n",
  "mtu": 1500,
  "jumboFrameCapable": true,
  "virtualGatewayId": "",
  "directConnectGatewayId": "879b76a1-403d-4700-8b53-4a56ed85436e",
  "routeFilterPrefixes": [],
  "bgpPeers": [
    {
      "bgpPeerId": "dxpeer-fEXAMPLE",
      "asn": 650001,
      "authKey": "0xzxgA9YoW9h58u8SEXAMPLE",
      "addressFamily": "ipv4",
      "amazonAddress": "169.254.248.1/30",
      "customerAddress": "169.254.248.2/30",
      "bgpPeerState": "available",
      "bgpStatus": "down",
```

```
        "awsDeviceV2": "loc1-26wz6vEXAMPLE"
      }
    ],
    "region": "sa-east-1",
    "awsDeviceV2": "loc1-26wz6vEXAMPLE",
    "tags": []
  }
```

Para obtener más información, consulte [Configuración de red MTU para interfaces virtuales privadas o interfaces virtuales de tránsito](#) en la Guía del usuario de AWS Direct Connect.

- Para API obtener más información, consulte [UpdateVirtualInterfaceAttributes](#) la Referencia de AWS CLI comandos.

## AWS Directory Service ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Directory Service.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **describe-directories**

En el siguiente ejemplo de código, se muestra cómo usar `describe-directories`.

AWS CLI

Para obtener detalles sobre tus directorios

En el siguiente `describe-directories` ejemplo, se muestran detalles sobre el directorio especificado.

```
aws ds describe-directories \  
  --directory-id d-a1b2c3d4e5
```

Salida:

```
{  
  "DirectoryDescriptions": [  
    {  
      "DirectoryId": "d-a1b2c3d4e5",  
      "Name": "mydirectory.example.com",  
      "ShortName": "mydirectory",  
      "Size": "Small",  
      "Edition": "Standard",  
      "Alias": "d-a1b2c3d4e5",  
      "AccessUrl": "d-a1b2c3d4e5.awsapps.com",  
      "Stage": "Active",  
      "ShareStatus": "Shared",  
      "ShareMethod": "HANDSHAKE",  
      "ShareNotes": "These are my share notes",  
      "LaunchTime": "2019-07-08T15:33:46.327000-07:00",  
      "StageLastUpdatedDateTime": "2019-07-08T15:59:12.307000-07:00",  
      "Type": "SharedMicrosoftAD",  
      "SsoEnabled": false,  
      "DesiredNumberOfDomainControllers": 0,  
      "OwnerDirectoryDescription": {  
        "DirectoryId": "d-b2c3d4e5f6",  
        "AccountId": "123456789111",  
        "DnsIpAddrs": [  
          "203.113.0.248",  
          "203.113.0.253"  
        ],  
        "VpcSettings": {  
          "VpcId": "vpc-a1b2c3d4",  
          "SubnetIds": [  
            "subnet-a1b2c3d4",  
            "subnet-d4c3b2a1"  
          ],  
          "AvailabilityZones": [  
            "us-west-2a",  
            "us-west-2c"  
          ]  
        }  
      }  
    ]  
  }  
}
```

```
    }  
  ]  
}
```

- Para API obtener más información, consulte [DescribeDirectories](#) la Referencia de AWS CLI comandos.

## describe-trusts

En el siguiente ejemplo de código, se muestra cómo usar `describe-trusts`.

### AWS CLI

Para obtener detalles sobre sus relaciones de confianza

En el siguiente `describe-trusts` ejemplo, se muestran detalles sobre las relaciones de confianza del directorio especificado.

```
aws ds describe-trusts \  
  --directory-id d-a1b2c3d4e5
```

Salida:

```
{  
  "Trusts": [  
    {  
      "DirectoryId": "d-a1b2c3d4e5",  
      "TrustId": "t-9a8b7c6d5e",  
      "RemoteDomainName": "other.example.com",  
      "TrustType": "Forest",  
      "TrustDirection": "Two-Way",  
      "TrustState": "Verified",  
      "CreatedDateTime": "2017-06-20T18:08:45.614000-07:00",  
      "LastUpdatedDateTime": "2019-06-04T10:52:12.410000-07:00",  
      "StateLastUpdatedDateTime": "2019-06-04T10:52:12.410000-07:00",  
      "SelectiveAuth": "Disabled"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [DescribeTrusts](#) la Referencia de AWS CLI comandos.

## AWS DMS ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS DMS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **add-tags-to-resource**

En el siguiente ejemplo de código, se muestra cómo usar `add-tags-to-resource`.

AWS CLI

Para añadir etiquetas a un recurso

En el siguiente `add-tags-to-resource` ejemplo, se agregan etiquetas a una instancia de replicación.

```
aws dms add-tags-to-resource \  
  --resource-arn arn:aws:dms:us-east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE \  
  \  
  --tags Key=Environment,Value=PROD Key=Project,Value=dbMigration
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de recursos](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [AddTagsToResource](#) la Referencia de AWS CLI comandos.

## create-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `create-endpoint`.

### AWS CLI

Para crear un punto final

El siguiente `create-endpoint` ejemplo crea un punto de conexión para una fuente de Amazon S3.

```
aws dms create-endpoint \  
  --endpoint-type source \  
  --engine-name s3 \  
  --endpoint-identifier src-endpoint \  
  --s3-settings file://s3-settings.json
```

Contenidos de `s3-settings.json`:

```
{  
  "BucketName": "my-corp-data",  
  "BucketFolder": "sourcedata",  
  "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-role"  
}
```

Salida:

```
{  
  "Endpoint": {  
    "EndpointIdentifier": "src-endpoint",  
    "EndpointType": "SOURCE",  
    "EngineName": "s3",  
    "EngineDisplayName": "Amazon S3",  
    "ExtraConnectionAttributes": "bucketFolder=sourcedata;bucketName=my-corp-  
data;compressionType=NONE;csvDelimiter=,;csvRowDelimiter=\\n",  
    "Status": "active",  
    "EndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:GUVAFG34EECU0J6QVZ56DAHT3U",  
    "SslMode": "none",  
    "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-role",  
    "S3Settings": {  
      "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-  
role",
```

```

        "CsvRowDelimiter": "\\n",
        "CsvDelimiter": ",",
        "BucketFolder": "sourcedata",
        "BucketName": "my-corp-data",
        "CompressionType": "NONE",
        "EnableStatistics": true
    }
}
}

```

Para obtener más información, consulte [Trabajar con AWS DMS puntos finales](#) en la Guía del usuario de AWS Database Migration Service.

- Para API obtener más información, consulte [CreateEndpoint](#) la Referencia de AWS CLI comandos.

## create-event-subscription

En el siguiente ejemplo de código, se muestra cómo usar `create-event-subscription`.

### AWS CLI

Para ver una lista de las suscripciones a eventos

En el siguiente `create-event-subscription` ejemplo, se crea una suscripción a un evento a un SNS tema de Amazon (`my-sns-topic`).

```

aws dms create-event-subscription \
  --subscription-name my-dms-events \
  --sns-topic-arn arn:aws:sns:us-east-1:123456789012:my-sns-topic

```

Salida:

```

{
  "EventSubscription": {
    "CustomerAwsId": "123456789012",
    "CustSubscriptionId": "my-dms-events",
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:my-sns-topic",
    "Status": "creating",
    "SubscriptionCreationTime": "2020-05-21 21:58:38.598",
    "Enabled": true
  }
}

```



```
}
```

Para obtener más información, consulte [Trabajo con eventos y notificaciones](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [CreateEventSubscription](#) la Referencia de AWS CLI comandos.

## **create-replication-instance**

En el siguiente ejemplo de código, se muestra cómo usar `create-replication-instance`.

### AWS CLI

Para crear una instancia de replicación

En el siguiente `create-replication-instance` ejemplo, se crea una instancia de replicación.

```
aws dms create-replication-instance \  
  --replication-instance-identifier my-repl-instance \  
  --replication-instance-class dms.t2.micro \  
  --allocated-storage 5
```

Salida:

```
{  
  "ReplicationInstance": {  
    "ReplicationInstanceIdentifier": "my-repl-instance",  
    "ReplicationInstanceClass": "dms.t2.micro",  
    "ReplicationInstanceStatus": "creating",  
    "AllocatedStorage": 5,  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-f839b688",  
        "Status": "active"  
      }  
    ],  
    "ReplicationSubnetGroup": {  
      "ReplicationSubnetGroupIdentifier": "default",  
      "ReplicationSubnetGroupDescription": "default",  
      "VpcId": "vpc-136a4c6a",  
      "SubnetGroupStatus": "Complete",
```

```
"Subnets": [  
  {  
    "SubnetIdentifier": "subnet-da327bf6",  
    "SubnetAvailabilityZone": {  
      "Name": "us-east-1a"  
    },  
    "SubnetStatus": "Active"  
  },  
  {  
    "SubnetIdentifier": "subnet-42599426",  
    "SubnetAvailabilityZone": {  
      "Name": "us-east-1d"  
    },  
    "SubnetStatus": "Active"  
  },  
  {  
    "SubnetIdentifier": "subnet-bac383e0",  
    "SubnetAvailabilityZone": {  
      "Name": "us-east-1c"  
    },  
    "SubnetStatus": "Active"  
  },  
  {  
    "SubnetIdentifier": "subnet-6746046b",  
    "SubnetAvailabilityZone": {  
      "Name": "us-east-1f"  
    },  
    "SubnetStatus": "Active"  
  },  
  {  
    "SubnetIdentifier": "subnet-d7c825e8",  
    "SubnetAvailabilityZone": {  
      "Name": "us-east-1e"  
    },  
    "SubnetStatus": "Active"  
  },  
  {  
    "SubnetIdentifier": "subnet-cbfff283",  
    "SubnetAvailabilityZone": {  
      "Name": "us-east-1b"  
    },  
    "SubnetStatus": "Active"  
  }  
]
```

```

    },
    "PreferredMaintenanceWindow": "sat:12:35-sat:13:05",
    "PendingModifiedValues": {},
    "MultiAZ": false,
    "EngineVersion": "3.3.2",
    "AutoMinorVersionUpgrade": true,
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/f7bc0f8e-1a3a-4ace-9faa-
e8494fa3921a",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:ZK2VQBUWFDBAWHIXHAYG5G2PKY",
    "PubliclyAccessible": true
  }
}

```

Para obtener más información, consulte [Trabajar con una instancia de AWS DMS replicación](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [CreateReplicationInstance](#) la Referencia de AWS CLI comandos.

## create-replication-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `create-replication-subnet-group`.

### AWS CLI

Para crear un grupo de subredes

En el siguiente `create-replication-subnet-group` ejemplo, se crea un grupo que consta de 3 subredes.

```

aws dms create-replication-subnet-group \
  --replication-subnet-group-identifier my-subnet-group \
  --replication-subnet-group-description "my subnet group" \
  --subnet-ids subnet-da327bf6 subnet-bac383e0 subnet-d7c825e8

```

Salida:

```

{
  "ReplicationSubnetGroup": {
    "ReplicationSubnetGroupIdentifier": "my-subnet-group",
    "ReplicationSubnetGroupDescription": "my subnet group",

```

```

    "VpcId": "vpc-136a4c6a",
    "SubnetGroupStatus": "Complete",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-da327bf6",
        "SubnetAvailabilityZone": {
          "Name": "us-east-1a"
        },
        "SubnetStatus": "Active"
      },
      {
        "SubnetIdentifier": "subnet-bac383e0",
        "SubnetAvailabilityZone": {
          "Name": "us-east-1c"
        },
        "SubnetStatus": "Active"
      },
      {
        "SubnetIdentifier": "subnet-d7c825e8",
        "SubnetAvailabilityZone": {
          "Name": "us-east-1e"
        },
        "SubnetStatus": "Active"
      }
    ]
  }
}

```

Para obtener más información, consulte [Configuración de una red para una instancia de replicación](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [CreateReplicationSubnetGroup](#) la Referencia de AWS CLI comandos.

## create-replication-task

En el siguiente ejemplo de código, se muestra cómo usar `create-replication-task`.

### AWS CLI

Para crear una tarea de replicación

El siguiente `create-replication-task` ejemplo crea una tarea de replicación.

```
aws dms create-replication-task \
  --replication-task-identifier movedata \
  --source-endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA \
  --target-endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U \
  --replication-instance-arn $RI_ARN \
  --migration-type full-load \
  --table-mappings file://table-mappings.json
```

Contenidos de `table-mappings.json`:

```
{
  "rules": [
    {
      "rule-type": "selection",
      "rule-id": "1",
      "rule-name": "1",
      "object-locator": {
        "schema-name": "prodrep",
        "table-name": "%"
      },
      "rule-action": "include",
      "filters": []
    }
  ]
}
```

Salida:

```
{
  "ReplicationTask": {
    "ReplicationTaskIdentifier": "moveit2",
    "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
    "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
    "MigrationType": "full-load",
    "TableMappings": "...output omitted... ",
    "ReplicationTaskSettings": "...output omitted... "
```

```

    "Status": "creating",
    "ReplicationTaskCreationDate": 1590524772.505,
    "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
  }
}

```

Para obtener más información, consulte [Trabajar con AWS DMS tareas](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [CreateReplicationTask](#) la Referencia de AWS CLI comandos.

## delete-connection

En el siguiente ejemplo de código, se muestra cómo usar `delete-connection`.

### AWS CLI

Para eliminar una conexión

El siguiente `delete-connection` ejemplo desasocia un punto final de una instancia de replicación.

```

aws dms delete-connection \
  --endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA \
  --replication-instance-arn arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE

```

Salida:

```

{
  "Connection": {
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
    "Status": "deleting",
    "EndpointIdentifier": "src-database-1",
    "ReplicationInstanceIdentifier": "my-repl-instance"
  }
}

```

```
}

```

Para obtener más información, consulte [https://docs.aws.amazon.com/dms/latest/userguide/CHAP\\_EndPoints.Creating.html](https://docs.aws.amazon.com/dms/latest/userguide/CHAP_EndPoints.Creating.html) en la Guía del usuario de Database Migration Service.AWS

- Para obtener API AWS CLI más información, [DeleteConnection](#) consulte la Referencia de comandos.

## delete-endpoint

En el siguiente ejemplo de código, se muestra cómo usar delete-endpoint.

### AWS CLI

Para eliminar un punto final

En el siguiente delete-endpoint ejemplo, se elimina un punto final.

```
aws dms delete-endpoint \
  --endpoint-arn arn:aws:dms:us-
  east-1:123456789012:endpoint:0UJJVX04XZ4CYTSEG5XGMN2R3Y
```

Salida:

```
{
  "Endpoint": {
    "EndpointIdentifier": "src-endpoint",
    "EndpointType": "SOURCE",
    "EngineName": "s3",
    "EngineDisplayName": "Amazon S3",
    "ExtraConnectionAttributes": "bucketFolder=sourcedata;bucketName=my-corp-
data;compressionType=NONE;csvDelimiter=,;csvRowDelimiter=\n;",
    "Status": "deleting",
    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:0UJJVX04XZ4CYTSEG5XGMN2R3Y",
    "SslMode": "none",
    "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-role",
    "S3Settings": {
      "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-
role",
      "CsvRowDelimiter": "\n",
      "CsvDelimiter": ",",
      "BucketFolder": "sourcedata",
```

```
        "BucketName": "my-corp-data",
        "CompressionType": "NONE",
        "EnableStatistics": true
    }
}
```

Para obtener más información, consulte [Trabajar con AWS DMS puntos finales](#) en la Guía del usuario de AWS Database Migration Service.

- Para API obtener más información, consulte [DeleteEndpoint](#) la Referencia de AWS CLI comandos.

## delete-event-subscription

En el siguiente ejemplo de código, se muestra cómo usar `delete-event-subscription`.

### AWS CLI

Para eliminar una suscripción a un evento

En el siguiente `delete-event-subscription` ejemplo, se elimina una suscripción a un SNS tema de Amazon.

```
aws dms delete-event-subscription \
  --subscription-name "my-dms-events"
```

Salida:

```
{
  "EventSubscription": {
    "CustomerAwsId": "123456789012",
    "CustSubscriptionId": "my-dms-events",
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:my-sns-topic",
    "Status": "deleting",
    "SubscriptionCreationTime": "2020-05-21 21:58:38.598",
    "Enabled": true
  }
}
```

Para obtener más información, consulte [Trabajo con eventos y notificaciones](#) en la Guía del usuario AWS de Database Migration Service.



- Para API obtener más información, consulte [DeleteEventSubscription](#) la Referencia de AWS CLI comandos.

## delete-replication-instance

En el siguiente ejemplo de código, se muestra cómo usar delete-replication-instance.

### AWS CLI

Para eliminar una instancia de replicación

En el siguiente ejemplo de la delete-replication-instance, se elimina una instancia de replicación.

```
aws dms delete-replication-instance \  
  --replication-instance-arn arn:aws:dms:us-  
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE
```

Salida:

```
{  
  "ReplicationInstance": {  
    "ReplicationInstanceIdentifier": "my-repl-instance",  
    "ReplicationInstanceClass": "dms.t2.micro",  
    "ReplicationInstanceStatus": "deleting",  
    "AllocatedStorage": 5,  
    "InstanceCreateTime": 1590011235.952,  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-f839b688",  
        "Status": "active"  
      }  
    ],  
    "AvailabilityZone": "us-east-1e",  
    "ReplicationSubnetGroup": {  
      "ReplicationSubnetGroupIdentifier": "default",  
      "ReplicationSubnetGroupDescription": "default",  
      "VpcId": "vpc-136a4c6a",  
      "SubnetGroupStatus": "Complete",  
      "Subnets": [  
        {  
          "SubnetIdentifier": "subnet-da327bf6",
```

```
        "SubnetAvailabilityZone": {
            "Name": "us-east-1a"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-42599426",
        "SubnetAvailabilityZone": {
            "Name": "us-east-1d"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-bac383e0",
        "SubnetAvailabilityZone": {
            "Name": "us-east-1c"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-6746046b",
        "SubnetAvailabilityZone": {
            "Name": "us-east-1f"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-d7c825e8",
        "SubnetAvailabilityZone": {
            "Name": "us-east-1e"
        },
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-cbfff283",
        "SubnetAvailabilityZone": {
            "Name": "us-east-1b"
        },
        "SubnetStatus": "Active"
    }
]
},
"PreferredMaintenanceWindow": "wed:11:42-wed:12:12",
"PendingModifiedValues": {},
```

```

    "MultiAZ": true,
    "EngineVersion": "3.3.2",
    "AutoMinorVersionUpgrade": true,
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/f7bc0f8e-1a3a-4ace-9faa-
e8494fa3921a",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
    "ReplicationInstancePublicIpAddress": "54.225.120.92",
    "ReplicationInstancePrivateIpAddress": "172.31.30.121",
    "ReplicationInstancePublicIpAddresses": [
        "54.225.120.92",
        "3.230.18.248"
    ],
    "ReplicationInstancePrivateIpAddresses": [
        "172.31.30.121",
        "172.31.75.90"
    ],
    "PubliclyAccessible": true,
    "SecondaryAvailabilityZone": "us-east-1b"
}
}

```

Para obtener más información, consulte [Trabajar con una instancia de AWS DMS replicación](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DeleteReplicationInstancela](#) Referencia de AWS CLI comandos.

## delete-replication-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar delete-replication-subnet-group.

### AWS CLI

Para eliminar un grupo de subredes

En el siguiente delete-replication-subnet-group ejemplo, se elimina un grupo de subredes.

```

aws dms delete-replication-subnet-group \
--replication-subnet-group-identifier my-subnet-group

```

Salida:

(none)

Para obtener más información, consulte [Configuración de una red para una instancia de replicación](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DeleteReplicationSubnetGroup](#) la Referencia de AWS CLI comandos.

## delete-replication-task

En el siguiente ejemplo de código, se muestra cómo usar delete-replication-task.

### AWS CLI

Para eliminar una tarea de replicación

En el siguiente delete-replication-task ejemplo, se elimina una tarea de replicación.

```
aws dms delete-replication-task \
  --replication-task-arn arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII
```

Salida:

```
{
  "ReplicationTask": {
    "ReplicationTaskIdentifier": "moveit2",
    "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWYGAYUVLKIB732KEVWA",
    "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
    "MigrationType": "full-load",
    "TableMappings": "...output omitted...",
    "ReplicationTaskSettings": "...output omitted...",
    "Status": "deleting",
    "StopReason": "Stop Reason FULL_LOAD_ONLY_FINISHED",
    "ReplicationTaskCreationDate": 1590524772.505,
    "ReplicationTaskStartDate": 1590789988.677,
    "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
```

```
}  
}
```

Para obtener más información, consulte [Trabajar con AWS DMS tareas](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DeleteReplicationTask](#) la Referencia de AWS CLI comandos.

## **describe-account-attributes**

En el siguiente ejemplo de código, se muestra cómo usar `describe-account-attributes`.

### AWS CLI

Descripción de atributos de cuenta

En el siguiente `describe-account-attributes` ejemplo, se enumeran los atributos de su AWS cuenta.

```
aws dms describe-account-attributes
```

Salida:

```
{  
  "AccountQuotas": [  
    {  
      "AccountQuotaName": "ReplicationInstances",  
      "Used": 1,  
      "Max": 20  
    },  
    {  
      "AccountQuotaName": "AllocatedStorage",  
      "Used": 5,  
      "Max": 10000  
    },  
    ...remaining output omitted...  
  ],  
  "UniqueAccountIdentifier": "cqahfbfy5xee"  
}
```

- Para API obtener más información, consulte [DescribeAccountAttributes](#) la Referencia de AWS CLI comandos.

## describe-certificates

En el siguiente ejemplo de código, se muestra cómo usar `describe-certificates`.

### AWS CLI

Para ver una lista de los certificados disponibles

En el siguiente `describe-certificates` ejemplo, se enumeran los certificados disponibles en su AWS cuenta.

```
aws dms describe-certificates
```

Salida:

```
{
  "Certificates": [
    {
      "CertificateIdentifier": "my-cert",
      "CertificateCreationDate": 1543259542.506,
      "CertificatePem": "-----BEGIN CERTIFICATE-----
\nMIID9DCCAtygAwIBAgIBQjANBgkqhkiG9w0BAQ ...U"
      ... remaining output omitted ...
    }
  ]
}
```

Para obtener más información, consulte [Uso SSL](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DescribeCertificates](#) la Referencia de AWS CLI comandos.

## describe-connections

En el siguiente ejemplo de código, se muestra cómo usar `describe-connections`.

## AWS CLI

Para describir las conexiones

En el siguiente `describe-connections` ejemplo, se enumeran las conexiones que ha probado entre una instancia de replicación y un punto final.

```
aws dms describe-connections
```

Salida:

```
{
  "Connections": [
    {
      "Status": "successful",
      "ReplicationInstanceIdentifier": "test",
      "EndpointArn": "arn:aws:dms:us-east-1:123456789012:endpoint:ZW5UAN6P4E77EC7YWHK4RZZ3BE",
      "EndpointIdentifier": "testsrc1",
      "ReplicationInstanceArn": "arn:aws:dms:us-east-1:123456789012:rep:6UTDJGB0US3VI3SUWA66XFJCJQ"
    }
  ]
}
```

Para obtener más información, consulte [Creación de puntos finales de origen y destino](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DescribeConnections](#) la Referencia de AWS CLI comandos.

## `describe-endpoint-types`

En el siguiente ejemplo de código, se muestra cómo usar `describe-endpoint-types`.

## AWS CLI

Para ver una lista de los tipos de puntos finales disponibles

En el siguiente `describe-endpoint-types` ejemplo, se enumeran los tipos SQL de puntos finales de My que están disponibles.

```
aws dms describe-endpoint-types \  
  --filters "Name=engine-name,Values=mysql"
```

Salida:

```
{  
  "SupportedEndpointTypes": [  
    {  
      "EngineName": "mysql",  
      "SupportsCDC": true,  
      "EndpointType": "source",  
      "EngineDisplayName": "MySQL"  
    },  
    {  
      "EngineName": "mysql",  
      "SupportsCDC": true,  
      "EndpointType": "target",  
      "EngineDisplayName": "MySQL"  
    }  
  ]  
}
```

Para obtener más información, consulte Trabajar con AWS DMS puntos finales < [https://docs.aws.amazon.com/dms/latest/userguide/CHAP\\_Endpoints.html](https://docs.aws.amazon.com/dms/latest/userguide/CHAP_Endpoints.html) > en la Guía del usuario de Database Migration Service.AWS

- Para obtener API más información, [DescribeEndpointTypes](#) consulte AWS CLI la Referencia de comandos.

## describe-endpoints

En el siguiente ejemplo de código, se muestra cómo usar describe-endpoints.

### AWS CLI

Para describir los puntos finales

En el siguiente describe-endpoints ejemplo, se enumeran los puntos de conexión de su cuenta. AWS

```
aws dms describe-endpoints
```



**Salida:**

```
{
  "Endpoints": [
    {
      "Username": "dms",
      "Status": "active",
      "EndpointArn": "arn:aws:dms:us-east-1:123456789012:endpoint:SF2W0FLWYWKVE0HID2EKLP3SJI",
      "ServerName": "ec2-52-32-48-61.us-west-2.compute.amazonaws.com",
      "EndpointType": "SOURCE",
      "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/94d5c4e7-4e4c-44be-b58a-c8da7adf57cd",
      "DatabaseName": "test",
      "EngineName": "mysql",
      "EndpointIdentifier": "pri100",
      "Port": 8193
    },
    {
      "Username": "admin",
      "Status": "active",
      "EndpointArn": "arn:aws:dms:us-east-1:123456789012:endpoint:TJJZCIH3CJ24TJRU4VC32WEWFR",
      "ServerName": "test.example.com",
      "EndpointType": "SOURCE",
      "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/2431021b-1cf2-a2d4-77b2-59a9e4bce323",
      "DatabaseName": "EMPL",
      "EngineName": "oracle",
      "EndpointIdentifier": "test",
      "Port": 1521
    }
  ]
}
```

Para obtener más información, consulte [Trabajar con AWS DMS puntos finales](#) en la Guía del usuario de AWS Database Migration Service.

- Para API obtener más información, consulte [DescribeEndpoints](#) la Referencia de AWS CLI comandos.

## describe-event-categories

En el siguiente ejemplo de código, se muestra cómo usar `describe-event-categories`.

### AWS CLI

Para describir las categorías de eventos

En el siguiente `describe-event-categories` ejemplo, se enumeran las categorías de eventos disponibles.

```
aws dms describe-event-categories
```

Salida:

```
{
  "EventCategoryGroupList": [
    {
      "SourceType": "replication-instance",
      "EventCategories": [
        "low storage",
        "configuration change",
        "maintenance",
        "deletion",
        "creation",
        "failover",
        "failure"
      ]
    },
    {
      "SourceType": "replication-task",
      "EventCategories": [
        "configuration change",
        "state change",
        "deletion",
        "creation",
        "failure"
      ]
    }
  ]
}
```

Para obtener más información, consulte [Trabajo con eventos y notificaciones](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DescribeEventCategories](#) la Referencia de AWS CLI comandos.

## describe-event-subscriptions

En el siguiente ejemplo de código, se muestra cómo usar `describe-event-subscriptions`.

### AWS CLI

Para describir las suscripciones a eventos

En el siguiente `describe-event-subscriptions` ejemplo, se enumeran las suscripciones de eventos a un SNS tema de Amazon.

```
aws dms describe-event-subscriptions
```

Salida:

```
{
  "EventSubscriptionsList": [
    {
      "CustomerAwsId": "123456789012",
      "CustSubscriptionId": "my-dms-events",
      "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:my-sns-topic",
      "Status": "deleting",
      "SubscriptionCreationTime": "2020-05-21 22:28:51.924",
      "Enabled": true
    }
  ]
}
```

Para obtener más información, consulte [Trabajo con eventos y notificaciones](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DescribeEventSubscriptions](#) la Referencia de AWS CLI comandos.

## describe-events

En el siguiente ejemplo de código, se muestra cómo usar `describe-events`.

### AWS CLI

Para enumerar DMS eventos

En el siguiente `describe-events` ejemplo, se enumeran los eventos que se originaron en una instancia de replicación.

```
aws dms describe-events \  
  --source-type "replication-instance"
```

Salida:

```
{  
  "Events": [  
    {  
      "SourceIdentifier": "my-repl-instance",  
      "SourceType": "replication-instance",  
      "Message": "Replication application shutdown",  
      "EventCategories": [],  
      "Date": 1590771645.776  
    }  
  ]  
}
```

Para obtener más información, consulte [Trabajo con eventos y notificaciones](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DescribeEvents](#) la Referencia de AWS CLI comandos.

## describe-orderable-replication-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-orderable-replication-instances`.

### AWS CLI

Para describir las instancias de replicación que se pueden solicitar

En el siguiente describe-orderable-replication-instances ejemplo, se enumeran los tipos de instancias de replicación que puede solicitar.

```
aws dms describe-orderable-replication-instances
```

Salida:

```
{
  "OrderableReplicationInstances": [
    {
      "EngineVersion": "3.3.2",
      "ReplicationInstanceClass": "dms.c4.2xlarge",
      "StorageType": "gp2",
      "MinAllocatedStorage": 5,
      "MaxAllocatedStorage": 6144,
      "DefaultAllocatedStorage": 100,
      "IncludedAllocatedStorage": 100,
      "AvailabilityZones": [
        "us-east-1a",
        "us-east-1b",
        "us-east-1c",
        "us-east-1d",
        "us-east-1e",
        "us-east-1f"
      ]
    },
    {
      "EngineVersion": "3.3.2",
      "ReplicationInstanceClass": "dms.c4.4xlarge",
      "StorageType": "gp2",
      "MinAllocatedStorage": 5,
      "MaxAllocatedStorage": 6144,
      "DefaultAllocatedStorage": 100,
      "IncludedAllocatedStorage": 100,
      "AvailabilityZones": [
        "us-east-1a",
        "us-east-1b",
        "us-east-1c",
        "us-east-1d",
        "us-east-1e",
        "us-east-1f"
      ]
    }
  ],
}
```

```

    ...remaining output omitted...
}

```

Para obtener más información, consulte [Trabajar con una instancia de AWS DMS replicación](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DescribeOrderableReplicationInstances](#) la Referencia de AWS CLI comandos.

## describe-refresh-schemas-status

En el siguiente ejemplo de código, se muestra cómo usar `describe-refresh-schemas-status`.

### AWS CLI

Para ver el estado de actualización de un punto final

El siguiente `describe-refresh-schemas-status` ejemplo devuelve el estado de una solicitud de actualización anterior.

```

aws dms describe-refresh-schemas-status \
  --endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA

```

Salida:

```

{
  "RefreshSchemasStatus": {
    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
    "Status": "successful",
    "LastRefreshDate": 1590786544.605
  }
}

```

- Para API obtener más información, consulte [DescribeRefreshSchemasStatus](#) la Referencia de AWS CLI comandos.

## describe-replication-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-replication-instances`.

### AWS CLI

Para describir las instancias de replicación

En el siguiente `describe-replication-instances` ejemplo, se enumeran las instancias de replicación de su AWS cuenta.

```
aws dms describe-replication-instances
```

Salida:

```
{
  "ReplicationInstances": [
    {
      "ReplicationInstanceIdentifier": "my-repl-instance",
      "ReplicationInstanceClass": "dms.t2.micro",
      "ReplicationInstanceStatus": "available",
      "AllocatedStorage": 5,
      "InstanceCreateTime": 1590011235.952,
      "VpcSecurityGroups": [
        {
          "VpcSecurityGroupId": "sg-f839b688",
          "Status": "active"
        }
      ],
      "AvailabilityZone": "us-east-1e",
      "ReplicationSubnetGroup": {
        "ReplicationSubnetGroupIdentifier": "default",
        "ReplicationSubnetGroupDescription": "default",
        "VpcId": "vpc-136a4c6a",
        "SubnetGroupStatus": "Complete",
        "Subnets": [
          {
            "SubnetIdentifier": "subnet-da327bf6",
            "SubnetAvailabilityZone": {
              "Name": "us-east-1a"
            },
            "SubnetStatus": "Active"
          }
        ]
      }
    }
  ],
}
```

```
    {
      "SubnetIdentifier": "subnet-42599426",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1d"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-bac383e0",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1c"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-6746046b",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1f"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-d7c825e8",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1e"
      },
      "SubnetStatus": "Active"
    },
    {
      "SubnetIdentifier": "subnet-cbfff283",
      "SubnetAvailabilityZone": {
        "Name": "us-east-1b"
      },
      "SubnetStatus": "Active"
    }
  ]
},
"PreferredMaintenanceWindow": "wed:11:42-wed:12:12",
"PendingModifiedValues": {
  "MultiAZ": true
},
"MultiAZ": false,
"EngineVersion": "3.3.2",
"AutoMinorVersionUpgrade": true,
```



```

        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/
f7bc0f8e-1a3a-4ace-9faa-e8494fa3921a",
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
        "ReplicationInstancePublicIpAddress": "3.230.18.248",
        "ReplicationInstancePrivateIpAddress": "172.31.75.90",
        "ReplicationInstancePublicIpAddresses": [
            "3.230.18.248"
        ],
        "ReplicationInstancePrivateIpAddresses": [
            "172.31.75.90"
        ],
        "PubliclyAccessible": true,
        "FreeUntil": 1590194829.267
    }
]
}

```

Para obtener más información, consulte [Trabajar con una instancia de AWS DMS replicación](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DescribeReplicationInstances](#) la Referencia de AWS CLI comandos.

## describe-replication-subnet-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-replication-subnet-groups`.

### AWS CLI

Para mostrar los grupos de subredes disponibles

En el siguiente `describe-replication-subnet-groups` ejemplo, se enumeran los grupos de subredes disponibles.

```

aws dms describe-replication-subnet-groups \
  --filter "Name=replication-subnet-group-id,Values=my-subnet-group"

```

Salida:

```
{
```

```

"ReplicationSubnetGroups": [
  {
    "ReplicationSubnetGroupIdentifier": "my-subnet-group",
    "ReplicationSubnetGroupDescription": "my subnet group",
    "VpcId": "vpc-136a4c6a",
    "SubnetGroupStatus": "Complete",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-da327bf6",
        "SubnetAvailabilityZone": {
          "Name": "us-east-1a"
        },
        "SubnetStatus": "Active"
      },
      {
        "SubnetIdentifier": "subnet-bac383e0",
        "SubnetAvailabilityZone": {
          "Name": "us-east-1c"
        },
        "SubnetStatus": "Active"
      },
      {
        "SubnetIdentifier": "subnet-d7c825e8",
        "SubnetAvailabilityZone": {
          "Name": "us-east-1e"
        },
        "SubnetStatus": "Active"
      }
    ]
  }
]
}

```

Para obtener más información, consulte [Configuración de una red para una instancia de replicación](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DescribeReplicationSubnetGroups](#) la Referencia de AWS CLI comandos.

## describe-replication-task-assessment-results

En el siguiente ejemplo de código, se muestra cómo usar `describe-replication-task-assessment-results`.

## AWS CLI

Para enumerar los resultados de las evaluaciones de las tareas de replicación

En el siguiente `describe-replication-task-assessment-results` ejemplo se enumeran los resultados de una evaluación de tareas anterior.

```
aws dms describe-replication-task-assessment-results
```

Salida:

```
{
  "ReplicationTaskAssessmentResults": [
    {
      "ReplicationTaskIdentifier": "moveit2",
      "ReplicationTaskArn": "arn:aws:dms:us-east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII",
      "ReplicationTaskLastAssessmentDate": 1590790230.0,
      "AssessmentStatus": "No issues found",
      "AssessmentResultsFile": "moveit2/2020-05-29-22-10"
    }
  ]
}
```

Para obtener más información, consulte [Creación de un informe de evaluación de tareas](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DescribeReplicationTaskAssessmentResults](#) la Referencia de AWS CLI comandos.

## `describe-replication-tasks`

En el siguiente ejemplo de código, se muestra cómo usar `describe-replication-tasks`.

## AWS CLI

Para describir una tarea de replicación

El siguiente `describe-replication-tasks` ejemplo describe las tareas de replicación actuales.

```
aws dms describe-replication-tasks
```

**Salida:**

```
{
  "ReplicationTasks": [
    {
      "ReplicationTaskIdentifier": "moveit2",
      "SourceEndpointArn": "arn:aws:dms:us-east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
      "TargetEndpointArn": "arn:aws:dms:us-east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",
      "ReplicationInstanceArn": "arn:aws:dms:us-east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
      "MigrationType": "full-load",
      "TableMappings": "...output omitted... ",
      "ReplicationTaskSettings": "...output omitted... ",
      "Status": "stopped",
      "StopReason": "Stop Reason FULL_LOAD_ONLY_FINISHED",
      "ReplicationTaskCreationDate": 1590524772.505,
      "ReplicationTaskStartDate": 1590619805.212,
      "ReplicationTaskArn": "arn:aws:dms:us-east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII",
      "ReplicationTaskStats": {
        "FullLoadProgressPercent": 100,
        "ElapsedTimeMillis": 0,
        "TablesLoaded": 0,
        "TablesLoading": 0,
        "TablesQueued": 0,
        "TablesErrored": 0,
        "FreshStartDate": 1590619811.528,
        "StartDate": 1590619811.528,
        "StopDate": 1590619842.068
      }
    }
  ]
}
```

Para obtener más información, consulte [Trabajar con AWS DMS tareas](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DescribeReplicationTasks](#) la Referencia de AWS CLI comandos.

## describe-schemas

En el siguiente ejemplo de código, se muestra cómo usar `describe-schemas`.

### AWS CLI

Para describir los esquemas de bases de datos

En el siguiente `describe-schemas` ejemplo, se enumeran las tablas disponibles en un punto final.

```
aws dms describe-schemas \  
  --endpoint-arn "arn:aws:dms:us-  
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA"
```

Salida:

```
{  
  "Schemas": [  
    "prodrep"  
  ]  
}
```

Para obtener más información, consulte [Este es el título del tema](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [DescribeSchemas](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas de una instancia de replicación.

```
aws dms list-tags-for-resource \  
  --resource-arn arn:aws:dms:us-east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE
```

Salida:

```
{
  "TagList": [
    {
      "Key": "Project",
      "Value": "dbMigration"
    },
    {
      "Key": "Environment",
      "Value": "PROD"
    }
  ]
}
```

Para obtener más información, consulte [Etiquetado de recursos](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## modify-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `modify-endpoint`.

### AWS CLI

Para modificar un punto final

El siguiente `modify-endpoint` ejemplo agrega un atributo de conexión adicional a un punto final.

```
aws dms modify-endpoint \
  --endpoint-arn "arn:aws:dms:us-
east-1:123456789012:endpoint:GUVAFG34EECU0J6QVZ56DAHT3U" \
  --extra-connection-attributes "compressionType=GZIP"
```

Salida:

```
{
  "Endpoint": {
    "EndpointIdentifier": "src-endpoint",
```

```

    "EndpointType": "SOURCE",
    "EngineName": "s3",
    "EngineDisplayName": "Amazon S3",
    "ExtraConnectionAttributes":
"compressionType=GZIP;csvDelimiter=,;csvRowDelimiter=\n;",
    "Status": "active",
    "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:GUVAFG34EECU0J6QVZ56DAHT3U",
    "SslMode": "none",
    "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-role",
    "S3Settings": {
        "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-
role",
        "CsvRowDelimiter": "\\n",
        "CsvDelimiter": ",",
        "BucketFolder": "",
        "BucketName": "",
        "CompressionType": "GZIP",
        "EnableStatistics": true
    }
}
}
}

```

Para obtener más información, consulte Trabajar con AWS DMS puntos finales < [https://docs.aws.amazon.com/dms/latest/userguide/CHAP\\_Endpoints.html](https://docs.aws.amazon.com/dms/latest/userguide/CHAP_Endpoints.html)> en la Guía del usuario de Database Migration Service.AWS

- Para obtener API más información, [ModifyEndpoint](#) consulte AWS CLI la Referencia de comandos.

## modify-event-subscription

En el siguiente ejemplo de código, se muestra cómo usar `modify-event-subscription`.

### AWS CLI

Para modificar una suscripción a un evento

En el siguiente `modify-event-subscription` ejemplo, se cambia el tipo de origen de una suscripción a un evento.

```

aws dms modify-event-subscription \
  --subscription-name "my-dms-events" \

```

```
--source-type replication-task
```

Salida:

```
{
  "EventSubscription": {
    "CustomerAwsId": "123456789012",
    "CustSubscriptionId": "my-dms-events",
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:my-sns-topic",
    "Status": "modifying",
    "SubscriptionCreationTime": "2020-05-29 17:04:40.262",
    "SourceType": "replication-task",
    "Enabled": true
  }
}
```

Para obtener más información, consulte [Trabajo con eventos y notificaciones](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [ModifyEventSubscription](#) la Referencia de AWS CLI comandos.

## modify-replication-instance

En el siguiente ejemplo de código, se muestra cómo usar `modify-replication-instance`.

AWS CLI

Para modificar una instancia de replicación

En el siguiente `modify-replication-instance` ejemplo, se modifica una instancia de replicación para que utilice una implementación Multi-AZ.

```
aws dms modify-replication-instance \
  --replication-instance-arn arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE \
  --multi-az
```

Salida:

```
{
```



```

"ReplicationInstance": {
  "ReplicationInstanceIdentifier": "my-repl-instance",
  "ReplicationInstanceClass": "dms.t2.micro",
  "ReplicationInstanceStatus": "available",
  "AllocatedStorage": 5,
  "InstanceCreateTime": 1590011235.952,

  ...output omitted...

  "PendingModifiedValues": {
    "MultiAZ": true
  },
  "MultiAZ": false,
  "EngineVersion": "3.3.2",
  "AutoMinorVersionUpgrade": true,
  "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/f7bc0f8e-1a3a-4ace-9faa-
e8494fa3921a",

  ...output omitted...
}
}

```

Para obtener más información, consulte [Trabajar con una instancia de AWS DMS replicación](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [ModifyReplicationInstance](#) la Referencia de AWS CLI comandos.

## modify-replication-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-replication-subnet-group`.

### AWS CLI

Para modificar un grupo de subredes

El siguiente `modify-replication-subnet-group` ejemplo cambia las listas de subredes asociadas a un grupo de subredes.

```

aws dms modify-replication-subnet-group \
  --replication-subnet-group-identifier my-subnet-group \

```

```
--subnet-id subnet-da327bf6 subnet-bac383e0
```

Salida:

```
{
  "ReplicationSubnetGroup": {
    "ReplicationSubnetGroupIdentifier": "my-subnet-group",
    "ReplicationSubnetGroupDescription": "my subnet group",
    "VpcId": "vpc-136a4c6a",
    "SubnetGroupStatus": "Complete",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-da327bf6",
        "SubnetAvailabilityZone": {
          "Name": "us-east-1a"
        },
        "SubnetStatus": "Active"
      },
      {
        "SubnetIdentifier": "subnet-bac383e0",
        "SubnetAvailabilityZone": {
          "Name": "us-east-1c"
        },
        "SubnetStatus": "Active"
      }
    ]
  }
}
```

Para obtener más información, consulte [Configuración de una red para una instancia de replicación](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [ModifyReplicationSubnetGroup](#) la Referencia de AWS CLI comandos.

## modify-replication-task

En el siguiente ejemplo de código, se muestra cómo usar `modify-replication-task`.

### AWS CLI

Para modificar una tarea de replicación

El siguiente `modify-replication-task` ejemplo cambia las asignaciones de tablas de una tarea.

```
aws dms modify-replication-task \
  --replication-task-arn "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII" \
  --table-mappings file://table-mappings.json
```

Contenidos de `table-mappings.json`:

```
{
  "rules": [
    {
      "rule-type": "selection",
      "rule-id": "1",
      "rule-name": "1",
      "object-locator": {
        "schema-name": "prodrep",
        "table-name": "ACCT_%"
      },
      "rule-action": "include",
      "filters": []
    }
  ]
}
```

Salida:

```
{
  "ReplicationTask": {
    "ReplicationTaskIdentifier": "moveit2",
    "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWGWAYUVLKIB732KEVWA",
    "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",
    "MigrationType": "full-load",
    "TableMappings": "...output omitted...",
    "ReplicationTaskSettings": "...output omitted...",
    "Status": "modifying",
    "StopReason": "Stop Reason FULL_LOAD_ONLY_FINISHED",
```

```
    "ReplicationTaskCreationDate": 1590524772.505,  
    "ReplicationTaskStartDate": 1590789424.653,  
    "ReplicationTaskArn": "arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"  
  }  
}
```

Para obtener más información, consulte [Trabajar con AWS DMS tareas](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [ModifyReplicationTask](#) la Referencia de AWS CLI comandos.

## reboot-replication-instance

En el siguiente ejemplo de código, se muestra cómo usar `reboot-replication-instance`.

### AWS CLI

Para reiniciar una instancia de replicación

En el siguiente ejemplo de la `reboot-replication-instance`, se reinicia una instancia de replicación.

```
aws dms reboot-replication-instance \  
  --replication-instance-arn arn:aws:dms:us-  
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE
```

Salida:

```
{  
  "ReplicationInstance": {  
    "ReplicationInstanceIdentifier": "my-repl-instance",  
    "ReplicationInstanceClass": "dms.t2.micro",  
    "ReplicationInstanceStatus": "rebooting",  
    "AllocatedStorage": 5,  
    "InstanceCreateTime": 1590011235.952,  
    ... output omitted ...  
  }  
}
```

Para obtener más información, consulte [Trabajar con una instancia de AWS DMS replicación](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [RebootReplicationInstance](#) la Referencia de AWS CLI comandos.

## refresh-schemas

En el siguiente ejemplo de código, se muestra cómo usar `refresh-schemas`.

### AWS CLI

Para actualizar los esquemas de bases de datos

En el siguiente `refresh-schemas` ejemplo, se solicita que se AWS DMS actualice la lista de esquemas en un punto final.

```
aws dms refresh-schemas \  
  --replication-instance-arn arn:aws:dms:us-  
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE \  
  --endpoint-arn "arn:aws:dms:us-  
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA"
```

Salida:

```
{  
  "RefreshSchemasStatus": {  
    "EndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",  
    "ReplicationInstanceArn": "arn:aws:dms:us-  
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",  
    "Status": "refreshing",  
    "LastRefreshDate": 1590019949.103  
  }  
}
```

- Para API obtener más información, consulte [RefreshSchemas](#) la Referencia de AWS CLI comandos.

## reload-tables

En el siguiente ejemplo de código, se muestra cómo usar `reload-tables`.

## AWS CLI

Para actualizar la lista de tablas disponibles en un punto final

En el siguiente `reload-tables` ejemplo, se vuelve a cargar la lista de tablas disponibles en un punto final.

```
aws dms reload-tables \  
  --replication-task-arn "arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII" \  
  --tables-to-reload "SchemaName=prodrep,TableName=ACCT_BAL"
```

Salida:

```
{  
  "ReplicationTaskArn": "arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"  
}
```

- Para API obtener más información, consulte [ReloadTables](#) la Referencia de AWS CLI comandos.

## remove-tags-from-resource

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags-from-resource`.

## AWS CLI

Para eliminar etiquetas de una instancia de replicación

En el siguiente `remove-tags-from-resource` ejemplo, se eliminan las etiquetas de una instancia de replicación.

```
aws dms remove-tags-from-resource \  
  --resource-arn arn:aws:dms:us-east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE  
 \  
  --tag-keys Environment Project
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de recursos](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [RemoveTagsFromResource](#) la Referencia de AWS CLI comandos.

## start-replication-task-assessment

En el siguiente ejemplo de código, se muestra cómo usar `start-replication-task-assessment`.

### AWS CLI

Para iniciar una evaluación de tareas

El siguiente `start-replication-task-assessment` ejemplo inicia una evaluación de la tarea de replicación.

```
aws dms start-replication-task-assessment \  
  --replication-task-arn arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII
```

Salida:

```
{  
  "ReplicationTask": {  
    "ReplicationTaskIdentifier": "moveit2",  
    "SourceEndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:6GGI6YPWGWAYUVLKIB732KEVWA",  
    "TargetEndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",  
    "ReplicationInstanceArn": "arn:aws:dms:us-  
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",  
    "MigrationType": "full-load",  
    "TableMappings": ...output omitted...,  
    "ReplicationTaskSettings": ...output omitted...,  
    "Status": "testing",  
    "StopReason": "Stop Reason FULL_LOAD_ONLY_FINISHED",  
    "ReplicationTaskCreationDate": 1590524772.505,  
    "ReplicationTaskStartDate": 1590789988.677,  
    "ReplicationTaskArn": "arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"  
  }  
}
```

Para obtener más información, consulte [Creación de un informe de evaluación de tareas](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [StartReplicationTaskAssessment](#) la Referencia de AWS CLI comandos.

## start-replication-task

En el siguiente ejemplo de código, se muestra cómo usar `start-replication-task`.

### AWS CLI

Para iniciar una tarea de replicación

El siguiente `command-name` ejemplo muestra los widgets disponibles en su AWS cuenta.

```
aws dms start-replication-task \  
  --replication-task-arn arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII \  
  --start-replication-task-type reload-target
```

Salida:

```
{  
  "ReplicationTask": {  
    "ReplicationTaskIdentifier": "moveit2",  
    "SourceEndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",  
    "TargetEndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",  
    "ReplicationInstanceArn": "arn:aws:dms:us-  
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",  
    "MigrationType": "full-load",  
    "TableMappings": ...output omitted... ,  
    "ReplicationTaskSettings": ...output omitted... ,  
    "Status": "starting",  
    "ReplicationTaskCreationDate": 1590524772.505,  
    "ReplicationTaskStartDate": 1590619805.212,  
    "ReplicationTaskArn": "arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"  
  }  
}
```



Para obtener más información, consulte [Trabajar con AWS DMS tareas](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [StartReplicationTask](#) la Referencia de AWS CLI comandos.

## stop-replication-task

En el siguiente ejemplo de código, se muestra cómo usar `stop-replication-task`.

### AWS CLI

Para detener una tarea

El siguiente `stop-replication-task` ejemplo detiene una tarea.

```
aws dms stop-replication-task \  
  --replication-task-arn arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII
```

Salida:

```
{  
  "ReplicationTask": {  
    "ReplicationTaskIdentifier": "moveit2",  
    "SourceEndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",  
    "TargetEndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:E0M4SFKCZEYHZBFGAGZT3QEC5U",  
    "ReplicationInstanceArn": "arn:aws:dms:us-  
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",  
    "MigrationType": "full-load",  
    "TableMappings": ...output omitted...,  
    "ReplicationTaskSettings": ...output omitted...,  
    "Status": "stopping",  
    "ReplicationTaskCreationDate": 1590524772.505,  
    "ReplicationTaskStartDate": 1590789424.653,  
    "ReplicationTaskArn": "arn:aws:dms:us-  
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"  
  }  
}
```

Para obtener más información, consulte [Trabajar con AWS DMS tareas](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [StopReplicationTask](#) la Referencia de AWS CLI comandos.

## test-connection

En el siguiente ejemplo de código, se muestra cómo usar `test-connection`.

### AWS CLI

Para probar una conexión a un punto final

El siguiente `test-connection` ejemplo comprueba si se puede acceder a un punto final desde una instancia de replicación.

```
aws dms test-connection \  
  --replication-instance-arn arn:aws:dms:us-  
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE \  
  --endpoint-arn arn:aws:dms:us-  
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA
```

Salida:

```
{  
  "Connection": {  
    "ReplicationInstanceArn": "arn:aws:dms:us-  
east-1:123456789012:rep:T30M70UB5NM2LCVZF7JPGJRNUE",  
    "EndpointArn": "arn:aws:dms:us-  
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",  
    "Status": "testing",  
    "EndpointIdentifier": "src-database-1",  
    "ReplicationInstanceIdentifier": "my-repl-instance"  
  }  
}
```

Para obtener más información, consulte [Creación de puntos finales de origen y destino](#) en la Guía del usuario AWS de Database Migration Service.

- Para API obtener más información, consulte [TestConnection](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon DocumentDB que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante Amazon DocumentDB. AWS Command Line Interface

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **add-tags-to-resource**

En el siguiente ejemplo de código, se muestra cómo usar `add-tags-to-resource`.

AWS CLI

Para añadir una o más etiquetas a un recurso específico

En el siguiente `add-tags-to-resource` ejemplo, se agregan tres etiquetas a `sample-cluster`. Una etiqueta (`CropB`) tiene un nombre clave pero no un valor.

```
aws docdb add-tags-to-resource \  
  --resource-name arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster \  
  --tags Key="CropA",Value="Apple" Key="CropB" Key="CropC",Value="Corn"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de los recursos de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia de [AddTagsToResource](#) comandos AWS CLI .

## apply-pending-maintenance-action

En el siguiente ejemplo de código, se muestra cómo usar `apply-pending-maintenance-action`.

### AWS CLI

Para que las acciones de mantenimiento pendientes se lleven a cabo durante el siguiente período de mantenimiento

El siguiente `apply-pending-maintenance-action` ejemplo hace que todas las acciones de actualización del sistema se realicen durante el siguiente período de mantenimiento programado.

```
aws docdb apply-pending-maintenance-action \  
--resource-identifier arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster \  
--apply-action system-update \  
--opt-in-type next-maintenance
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Aplicación de las actualizaciones de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [ApplyPendingMaintenanceAction](#) de AWS CLI comandos.

## copy-db-cluster-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `copy-db-cluster-parameter-group`.

### AWS CLI

Para duplicar un grupo de parámetros de un clúster de base de datos existente

El siguiente `copy-db-cluster-parameter-group` ejemplo hace una copia del grupo de parámetros `custom-docdb3-6` denominado `custom-docdb3-6-copy`. Al realizar la copia, añada etiquetas al nuevo grupo de parámetros.

```
aws docdb copy-db-cluster-parameter-group \  
--source-db-cluster-parameter-group-identifier custom-docdb3-6 \  
--target-db-cluster-parameter-group-identifier custom-docdb3-6-copy
```

```
--target-db-cluster-parameter-group-identifier custom-docdb3-6-copy \  
--target-db-cluster-parameter-group-description "Copy of custom-docdb3-6" \  
--tags Key="CopyNumber",Value="1" Key="Modifiable",Value="Yes"
```

Salida:

```
{  
  "DBClusterParameterGroup": {  
    "DBParameterGroupFamily": "docdb3.6",  
    "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:12345678901:cluster-  
pg:custom-docdb3-6-copy",  
    "DBClusterParameterGroupName": "custom-docdb3-6-copy",  
    "Description": "Copy of custom-docdb3-6"  
  }  
}
```

Para obtener más información, consulte [Copiar un grupo de parámetros de un clúster de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [CopyDbClusterParameterGroup](#) de AWS CLI comandos.

## copy-db-cluster-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `copy-db-cluster-snapshot`.

AWS CLI

Para crear una copia de una instantánea

En el siguiente ejemplo de la `copy-db-cluster-snapshot` se crea una copia de `sample-cluster-snapshot` llamada `sample-cluster-snapshot-copy`. La copia tiene todas las etiquetas del original más una etiqueta nueva con el nombre de la clave `CopyNumber`.

```
aws docdb copy-db-cluster-snapshot \  
--source-db-cluster-snapshot-identifier sample-cluster-snapshot \  
--target-db-cluster-snapshot-identifier sample-cluster-snapshot-copy \  
--copy-tags \  
--tags Key="CopyNumber",Value="1"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Copiar una instantánea de clúster](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte [CopyDbClusterSnapshot](#) la Referencia de AWS CLI comandos.

## **create-db-cluster-parameter-group**

En el siguiente ejemplo de código, se muestra cómo usar `create-db-cluster-parameter-group`.

### AWS CLI

Para crear un grupo de parámetros de clúster de Amazon DocumentDB

En el siguiente `create-db-cluster-parameter-group` ejemplo, se crea el grupo de parámetros del clúster de base de datos `sample-parameter-group` utilizando la `docdb3.6` familia.

```
aws docdb create-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name sample-parameter-group \  
  --db-parameter-group-family docdb3.6 \  
  --description "Sample parameter group based on docdb3.6"
```

Salida:

```
{  
  "DBClusterParameterGroup": {  
    "Description": "Sample parameter group based on docdb3.6",  
    "DBParameterGroupFamily": "docdb3.6",  
    "DBClusterParameterGroupArn": "arn:aws:rds:us-west-2:123456789012:cluster-pg:sample-parameter-group",  
    "DBClusterParameterGroupName": "sample-parameter-group"  
  }  
}
```

Para obtener más información, consulte [Creación de un grupo de parámetros de clúster de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [CreateDbClusterParameterGroup](#) de AWS CLI comandos.

## create-db-cluster-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `create-db-cluster-snapshot`.

### AWS CLI

Para crear una instantánea manual del clúster de Amazon DocumentDB

En el siguiente `create-db-cluster-snapshot` ejemplo, se crea una instantánea de un clúster de Amazon DB con el nombre `sample-cluster-snapshot`.

```
aws docdb create-db-cluster-snapshot \  
  --db-cluster-identificador sample-cluster \  
  --db-cluster-snapshot-identificador sample-cluster-snapshot
```

Salida:

```
{  
  "DBClusterSnapshot": {  
    "MasterUsername": "master-user",  
    "SnapshotCreateTime": "2019-03-18T18:27:14.794Z",  
    "AvailabilityZones": [  
      "us-west-2a",  
      "us-west-2b",  
      "us-west-2c",  
      "us-west-2d",  
      "us-west-2e",  
      "us-west-2f"  
    ],  
    "SnapshotType": "manual",  
    "DBClusterSnapshotArn": "arn:aws:rds:us-west-2:123456789012:cluster-snapshot:sample-cluster-snapshot",  
    "EngineVersion": "3.6.0",  
    "PercentProgress": 0,  
    "DBClusterSnapshotIdentifier": "sample-cluster-snapshot",  
    "Engine": "docdb",  
    "DBClusterIdentifier": "sample-cluster",  
    "Status": "creating",  
    "ClusterCreateTime": "2019-03-15T20:29:58.836Z",  
    "Port": 0,  
    "StorageEncrypted": false,  
    "VpcId": "vpc-91280df6"  
  }  
}
```

```
}
```

Para obtener más información, consulte [Creación de una instantánea de clúster manual](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte [CreateDbClusterSnapshot](#) la Referencia de AWS CLI comandos.

## **create-db-cluster**

En el siguiente ejemplo de código, se muestra cómo usar `create-db-cluster`.

### AWS CLI

Para crear un clúster de Amazon DocumentDB

En el siguiente `create-db-cluster` ejemplo, se crea un clúster de Amazon DocumentDB denominado `sample-cluster` con el intervalo de mantenimiento preferido los domingos entre las 20:30 y las 11:00.

```
aws docdb create-db-cluster \  
  --db-cluster-identifier sample-cluster \  
  --engine docdb \  
  --master-username master-user \  
  --master-user-password password \  
  --preferred-maintenance-window Sun:20:30-Sun:21:00
```

Salida:

```
{  
  "DBCluster": {  
    "DBClusterParameterGroup": "default.docdb3.6",  
    "AssociatedRoles": [],  
    "DBSubnetGroup": "default",  
    "ClusterCreateTime": "2019-03-18T18:06:34.616Z",  
    "Status": "creating",  
    "Port": 27017,  
    "PreferredMaintenanceWindow": "sun:20:30-sun:21:00",  
    "HostedZoneId": "ZNKXH85TT8WVW",  
    "DBClusterMembers": [],  
    "Engine": "docdb",
```



```

    "DBClusterIdentifier": "sample-cluster",
    "PreferredBackupWindow": "10:12-10:42",
    "AvailabilityZones": [
        "us-west-2d",
        "us-west-2f",
        "us-west-2e"
    ],
    "MasterUsername": "master-user",
    "BackupRetentionPeriod": 1,
    "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
    "VpcSecurityGroups": [
        {
            "VpcSecurityGroupId": "sg-77186e0d",
            "Status": "active"
        }
    ],
    "StorageEncrypted": false,
    "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster",
    "DbClusterResourceId": "cluster-L3R4YRSBUYDP4GLMTJ2WF5GH5Q",
    "MultiAZ": false,
    "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
    "EngineVersion": "3.6.0"
}
}

```

Para obtener más información, consulte [Creación de un clúster de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [CreateDbCluster](#) de AWS CLI comandos.

## create-db-instance

En el siguiente ejemplo de código, se muestra cómo usar `create-db-instance`.

### AWS CLI

Para crear una instancia de clúster de Amazon DocumentDB

El siguiente código de `create-db-instance` ejemplo crea la instancia `sample-cluster-instance-2` en el clúster de Amazon DocumentDB. `sample-cluster`

```
aws docdb create-db-instance \  
  --db-cluster-identifier sample-cluster \  
  --db-instance-class db.r4.xlarge \  
  --db-instance-identifier sample-cluster-instance-2 \  
  --engine docdb
```

Salida:

```
{  
  "DBInstance": {  
    "DBInstanceStatus": "creating",  
    "PendingModifiedValues": {  
      "PendingCloudwatchLogsExports": {  
        "LogTypesToEnable": [  
          "audit"  
        ]  
      }  
    },  
    "PubliclyAccessible": false,  
    "PreferredBackupWindow": "00:00-00:30",  
    "PromotionTier": 1,  
    "EngineVersion": "3.6.0",  
    "BackupRetentionPeriod": 3,  
    "DBInstanceIdentifier": "sample-cluster-instance-2",  
    "PreferredMaintenanceWindow": "tue:10:28-tue:10:58",  
    "StorageEncrypted": false,  
    "Engine": "docdb",  
    "DBClusterIdentifier": "sample-cluster",  
    "DBSubnetGroup": {  
      "Subnets": [  
        {  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2a"  
          },  
          "SubnetStatus": "Active",  
          "SubnetIdentifier": "subnet-4e26d263"  
        },  
        {  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2c"  
          },  
          "SubnetStatus": "Active",  
          "SubnetIdentifier": "subnet-afc329f4"  
        }  
      ]  
    }  
  }  
}
```

```

    },
    {
      "SubnetAvailabilityZone": {
        "Name": "us-west-2d"
      },
      "SubnetStatus": "Active",
      "SubnetIdentifier": "subnet-53ab3636"
    },
    {
      "SubnetAvailabilityZone": {
        "Name": "us-west-2b"
      },
      "SubnetStatus": "Active",
      "SubnetIdentifier": "subnet-991cb8d0"
    }
  ],
  "DBSubnetGroupDescription": "default",
  "SubnetGroupStatus": "Complete",
  "VpcId": "vpc-91280df6",
  "DBSubnetGroupName": "default"
},
"DBInstanceClass": "db.r4.xlarge",
"VpcSecurityGroups": [
  {
    "Status": "active",
    "VpcSecurityGroupId": "sg-77186e0d"
  }
],
"DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster-
instance-2",
"DbiResourceId": "db-XEKJLEMGRV5ZKCARUVA4H03ITE"
}
}

```

Para obtener más información, consulte [Añadir una instancia de Amazon DocumentDB a un clúster](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [CreateDbInstance](#) de AWS CLI comandos.

## create-db-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `create-db-subnet-group`.

## AWS CLI

Para crear un grupo de subredes de Amazon DocumentDB

En el siguiente `create-db-subnet-group` ejemplo, se crea un grupo de subredes de Amazon DocumentDB denominado `sample-subnet-group`

```
aws docdb create-db-subnet-group \  
  --db-subnet-group-description "a sample subnet group" \  
  --db-subnet-group-name sample-subnet-group \  
  --subnet-ids "subnet-29ab1025" "subnet-991cb8d0" "subnet-53ab3636"
```

Salida:

```
{  
  "DBSubnetGroup": {  
    "SubnetGroupStatus": "Complete",  
    "DBSubnetGroupName": "sample-subnet-group",  
    "DBSubnetGroupDescription": "a sample subnet group",  
    "VpcId": "vpc-91280df6",  
    "DBSubnetGroupArn": "arn:aws:rds:us-west-2:123456789012:subgrp:sample-  
subnet-group",  
    "Subnets": [  
      {  
        "SubnetStatus": "Active",  
        "SubnetIdentifier": "subnet-53ab3636",  
        "SubnetAvailabilityZone": {  
          "Name": "us-west-2d"  
        }  
      },  
      {  
        "SubnetStatus": "Active",  
        "SubnetIdentifier": "subnet-991cb8d0",  
        "SubnetAvailabilityZone": {  
          "Name": "us-west-2b"  
        }  
      },  
      {  
        "SubnetStatus": "Active",  
        "SubnetIdentifier": "subnet-29ab1025",  
        "SubnetAvailabilityZone": {  
          "Name": "us-west-2c"  
        }  
      }  
    ]  
  }  
}
```

```
    }  
  ]  
}
```

Para obtener más información, consulte [Creación de un grupo de subredes de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia de [CreateDbSubnetGroup](#) comandos AWS CLI .

## **delete-db-cluster-parameter-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-cluster-parameter-group`.

AWS CLI

Para eliminar un grupo de parámetros de un clúster de Amazon DocumentDB

En el siguiente `delete-db-cluster-parameter-group` ejemplo, se elimina el grupo de parámetros de Amazon DocumentDB. `sample-parameter-group`

```
aws docdb delete-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name sample-parameter-group
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un grupo de parámetros de clúster de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [DeleteDbClusterParameterGroup](#) de AWS CLI comandos.

## **delete-db-cluster-snapshot**

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-cluster-snapshot`.

AWS CLI

Para eliminar una instantánea de un clúster de Amazon DocumentDB

En el siguiente `delete-db-cluster-snapshot` ejemplo, se elimina la instantánea del clúster de Amazon DocumentDB. `sample-cluster-snapshot`

```
aws docdb delete-db-cluster-snapshot \  
  --db-cluster-snapshot-identifíer sample-cluster-snapshot
```

Salida:

```
{  
  "DBClusterSnapshot": {  
    "DBClusterIdentifier": "sample-cluster",  
    "AvailabilityZones": [  
      "us-west-2a",  
      "us-west-2b",  
      "us-west-2c",  
      "us-west-2d"  
    ],  
    "DBClusterSnapshotIdentifier": "sample-cluster-snapshot",  
    "VpcId": "vpc-91280df6",  
    "DBClusterSnapshotArn": "arn:aws:rds:us-west-2:123456789012:cluster-  
snapshot:sample-cluster-snapshot",  
    "EngineVersion": "3.6.0",  
    "Engine": "docdb",  
    "SnapshotCreateTime": "2019-03-18T18:27:14.794Z",  
    "Status": "available",  
    "MasterUsername": "master-user",  
    "ClusterCreateTime": "2019-03-15T20:29:58.836Z",  
    "PercentProgress": 100,  
    "StorageEncrypted": false,  
    "SnapshotType": "manual",  
    "Port": 0  
  }  
}
```

Para obtener más información, consulte [Eliminar una instantánea de clúster](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte [DeleteDbClusterSnapshot](#) la Referencia de AWS CLI comandos.

## delete-db-cluster

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-cluster`.

### AWS CLI

Para eliminar un clúster de Amazon DocumentDB

En el siguiente `delete-db-cluster` ejemplo, se elimina el clúster de Amazon `sample-cluster` DocumentDB. No se realiza ninguna copia de seguridad del clúster antes de eliminarlo. NOTE: Debe eliminar todas las instancias asociadas al clúster para poder eliminarlo.

```
aws docdb delete-db-cluster \  
  --db-cluster-identifier sample-cluster \  
  --skip-final-snapshot
```

Salida:

```
{  
  "DBCluster": {  
    "DBClusterIdentifier": "sample-cluster",  
    "DBSubnetGroup": "default",  
    "EngineVersion": "3.6.0",  
    "Engine": "docdb",  
    "LatestRestorableTime": "2019-03-18T18:07:24.610Z",  
    "PreferredMaintenanceWindow": "sun:20:30-sun:21:00",  
    "StorageEncrypted": false,  
    "EarliestRestorableTime": "2019-03-18T18:07:24.610Z",  
    "Port": 27017,  
    "VpcSecurityGroups": [  
      {  
        "Status": "active",  
        "VpcSecurityGroupId": "sg-77186e0d"  
      }  
    ],  
    "MultiAZ": false,  
    "MasterUsername": "master-user",  
    "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster",  
    "Status": "available",  
    "PreferredBackupWindow": "10:12-10:42",  
    "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-  
west-2.docdb.amazonaws.com",  
    "AvailabilityZones": [  

```

```

        "us-west-2c",
        "us-west-2b",
        "us-west-2a"
    ],
    "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
    "DbClusterResourceId": "cluster-L3R4YRSBUYDP4GLMTJ2WF5GH5Q",
    "ClusterCreateTime": "2019-03-18T18:06:34.616Z",
    "AssociatedRoles": [],
    "DBClusterParameterGroup": "default.docdb3.6",
    "HostedZoneId": "ZNKXH85TT8WVW",
    "BackupRetentionPeriod": 1,
    "DBClusterMembers": []
}
}

```

Para obtener más información, consulte [Eliminar un clúster de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [DeleteDbCluster](#) de AWS CLI comandos.

## delete-db-instance

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-instance`.

### AWS CLI

Para eliminar una instancia de Amazon DocumentDB

En el siguiente `delete-db-instance` ejemplo, se elimina la instancia de Amazon `sample-cluster-instance-2` DocumentDB.

```
aws docdb delete-db-instance \
  --db-instance-identifier sample-cluster-instance-2
```

Salida:

```
{
  "DBInstance": {
    "DBSubnetGroup": {
      "Subnets": [
        {
```



```
        "SubnetAvailabilityZone": {
            "Name": "us-west-2a"
        },
        "SubnetStatus": "Active",
        "SubnetIdentifier": "subnet-4e26d263"
    },
    {
        "SubnetAvailabilityZone": {
            "Name": "us-west-2c"
        },
        "SubnetStatus": "Active",
        "SubnetIdentifier": "subnet-afc329f4"
    },
    {
        "SubnetAvailabilityZone": {
            "Name": "us-west-2d"
        },
        "SubnetStatus": "Active",
        "SubnetIdentifier": "subnet-53ab3636"
    },
    {
        "SubnetAvailabilityZone": {
            "Name": "us-west-2b"
        },
        "SubnetStatus": "Active",
        "SubnetIdentifier": "subnet-991cb8d0"
    }
],
"DBSubnetGroupName": "default",
"DBSubnetGroupDescription": "default",
"VpcId": "vpc-91280df6",
"SubnetGroupStatus": "Complete"
},
"PreferredBackupWindow": "00:00-00:30",
"InstanceCreateTime": "2019-03-18T18:37:33.709Z",
"DBInstanceClass": "db.r4.xlarge",
"DbiResourceId": "db-XEKJLEMGRV5ZKCARUVA4H03ITE",
"BackupRetentionPeriod": 3,
"Engine": "docdb",
"VpcSecurityGroups": [
    {
        "Status": "active",
        "VpcSecurityGroupId": "sg-77186e0d"
    }
]
```

```
    ],
    "AutoMinorVersionUpgrade": true,
    "PromotionTier": 1,
    "EngineVersion": "3.6.0",
    "Endpoint": {
      "Address": "sample-cluster-instance-2.corcjzrlsfc.us-
west-2.docdb.amazonaws.com",
      "HostedZoneId": "ZNKXH85TT8WVW",
      "Port": 27017
    },
    "DBInstanceIdentifier": "sample-cluster-instance-2",
    "PreferredMaintenanceWindow": "tue:10:28-tue:10:58",
    "EnabledCloudwatchLogsExports": [
      "audit"
    ],
    "PendingModifiedValues": {},
    "DBInstanceStatus": "deleting",
    "PubliclyAccessible": false,
    "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster-
instance-2",
    "DBClusterIdentifier": "sample-cluster",
    "AvailabilityZone": "us-west-2c",
    "StorageEncrypted": false
  }
}
```

Para obtener más información, consulte [Eliminar una instancia de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [DeleteDbInstance](#) de AWS CLI comandos.

## delete-db-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-subnet-group`.

### AWS CLI

Para eliminar un grupo de subredes de Amazon DocumentDB

En el siguiente `delete-db-subnet-group` ejemplo, se elimina el grupo de subredes de Amazon DocumentDB. `sample-subnet-group`

```
aws docdb delete-db-subnet-group \  
  --db-subnet-group-name sample-subnet-group
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un grupo de subredes de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia de [DeleteDbSubnetGroup](#) comandos AWS CLI .

## describe-db-cluster-parameter-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-cluster-parameter-groups`.

AWS CLI

Para ver los detalles de uno o más grupos de parámetros del clúster de Amazon DocumentDB

En el siguiente `describe-db-cluster-parameter-groups` ejemplo, se muestran los detalles del grupo de parámetros del clúster de Amazon DocumentDB. `custom3-6-param-grp`

```
aws docdb describe-db-cluster-parameter-groups \  
  --db-cluster-parameter-group-name custom3-6-param-grp
```

Salida:

```
{  
  "DBClusterParameterGroups": [  
    {  
      "DBParameterGroupFamily": "docdb3.6",  
      "DBClusterParameterGroupArn": "arn:aws:rds:us-  
east-1:123456789012:cluster-pg:custom3-6-param-grp",  
      "Description": "Custom docdb3.6 parameter group",  
      "DBClusterParameterGroupName": "custom3-6-param-grp"  
    }  
  ]  
}
```

Para obtener más información, consulte [Visualización de los grupos de parámetros del clúster de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [DescribeDbClusterParameterGroups](#) de AWS CLI comandos.

## **describe-db-cluster-parameters**

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-cluster-parameters`.

### AWS CLI

Para ver la lista de parámetros detallada de un grupo de parámetros de clúster de Amazon DocumentDB.

En el siguiente `describe-db-cluster-parameters` ejemplo, se enumeran los parámetros del grupo de parámetros `custom3-6-param-grp` de Amazon DocumentDB.

```
aws docdb describe-db-cluster-parameters \  
  --db-cluster-parameter-group-name custom3-6-param-grp
```

Salida:

```
{  
  "Parameters": [  
    {  
      "DataType": "string",  
      "ParameterName": "audit_logs",  
      "IsModifiable": true,  
      "ApplyMethod": "pending-reboot",  
      "Source": "system",  
      "ApplyType": "dynamic",  
      "AllowedValues": "enabled,disabled",  
      "Description": "Enables auditing on cluster.",  
      "ParameterValue": "disabled"  
    },  
    {  
      "DataType": "string",  
      "ParameterName": "tls",  
      "IsModifiable": true,  
      "ApplyMethod": "pending-reboot",  
      "Source": "system",
```

```

    "ApplyType": "static",
    "AllowedValues": "disabled,enabled",
    "Description": "Config to enable/disable TLS",
    "ParameterValue": "enabled"
  },
  {
    "DataType": "string",
    "ParameterName": "ttl_monitor",
    "IsModifiable": true,
    "ApplyMethod": "pending-reboot",
    "Source": "user",
    "ApplyType": "dynamic",
    "AllowedValues": "disabled,enabled",
    "Description": "Enables TTL Monitoring",
    "ParameterValue": "enabled"
  }
]
}

```

Para obtener más información, consulte [Visualización de los parámetros del clúster de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [DescribeDbClusterParameters](#) de AWS CLI comandos.

## describe-db-cluster-snapshot-attributes

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-cluster-snapshot-attributes`.

### AWS CLI

Para enumerar los nombres y valores de los atributos de una instantánea de Amazon DocumentDB

En el siguiente `describe-db-cluster-snapshot-attributes` ejemplo, se enumeran los nombres y valores de los atributos de la instantánea de Amazon DocumentDB. `sample-cluster-snapshot`

```

aws docdb describe-db-cluster-snapshot-attributes \
  --db-cluster-snapshot-identifier sample-cluster-snapshot

```

Salida:

```
{
  "DBClusterSnapshotAttributesResult": {
    "DBClusterSnapshotAttributes": [
      {
        "AttributeName": "restore",
        "AttributeValues": []
      }
    ],
    "DBClusterSnapshotIdentifier": "sample-cluster-snapshot"
  }
}
```

Para obtener más información, consulte [DescribeDBCluster SnapshotAttributes](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte [DescribeDbClusterSnapshotAttributes](#) la Referencia de AWS CLI comandos.

## describe-db-cluster-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-cluster-snapshots`.

AWS CLI

Para describir las instantáneas de Amazon DocumentDB

En el siguiente `describe-db-cluster-snapshots` ejemplo, se muestran los detalles de la instantánea de Amazon DocumentDB. `sample-cluster-snapshot`

```
aws docdb describe-db-cluster-snapshots \
  --db-cluster-snapshot-identifier sample-cluster-snapshot
```

Salida:

```
{
  "DBClusterSnapshots": [
    {
      "AvailabilityZones": [
        "us-west-2a",
        "us-west-2b",
```

```

        "us-west-2c",
        "us-west-2d"
    ],
    "Status": "available",
    "DBClusterSnapshotArn": "arn:aws:rds:us-west-2:123456789012:cluster-
snapshot:sample-cluster-snapshot",
    "SnapshotCreateTime": "2019-03-15T20:41:26.515Z",
    "SnapshotType": "manual",
    "DBClusterSnapshotIdentifier": "sample-cluster-snapshot",
    "DBClusterIdentifier": "sample-cluster",
    "MasterUsername": "master-user",
    "StorageEncrypted": false,
    "VpcId": "vpc-91280df6",
    "EngineVersion": "3.6.0",
    "PercentProgress": 100,
    "Port": 0,
    "Engine": "docdb",
    "ClusterCreateTime": "2019-03-15T20:29:58.836Z"
    }
]
}

```

Para obtener más información, consulte [escribeDBClusterInstantáneas D](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [DescribeDbClusterSnapshots](#) de AWS CLI comandos.

## describe-db-clusters

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-clusters`.

### AWS CLI

Para obtener información detallada sobre uno o más clústeres de Amazon DocumentDB.

En el siguiente `describe-db-clusters` ejemplo, se muestran los detalles del clúster de Amazon DocumentDB. `sample-cluster` Al omitir el `--db-cluster-identifier` parámetro, puede obtener información de hasta 100 clústeres.

```

aws docdb describe-db-clusters
  --db-cluster-identifier sample-cluster

```

**Salida:**

```
{
  "DBClusters": [
    {
      "DBClusterParameterGroup": "default.docdb3.6",
      "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
      "PreferredBackupWindow": "00:00-00:30",
      "DBClusterIdentifier": "sample-cluster",
      "ClusterCreateTime": "2019-03-15T20:29:58.836Z",
      "LatestRestorableTime": "2019-03-18T20:28:03.239Z",
      "MasterUsername": "master-user",
      "DBClusterMembers": [
        {
          "PromotionTier": 1,
          "DBClusterParameterGroupStatus": "in-sync",
          "IsClusterWriter": false,
          "DBInstanceIdentifier": "sample-cluster"
        },
        {
          "PromotionTier": 1,
          "DBClusterParameterGroupStatus": "in-sync",
          "IsClusterWriter": true,
          "DBInstanceIdentifier": "sample-cluster2"
        }
      ],
      "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
      "VpcSecurityGroups": [
        {
          "VpcSecurityGroupId": "sg-77186e0d",
          "Status": "active"
        }
      ],
      "Engine": "docdb",
      "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
      "DBSubnetGroup": "default",
      "MultiAZ": true,
      "AvailabilityZones": [
        "us-west-2a",
        "us-west-2c",
        "us-west-2b"
      ],
    },
  ],
}
```



```

    "EarliestRestorableTime": "2019-03-15T20:30:47.020Z",
    "DbClusterResourceId": "cluster-UP4EF2PVDDFVHHDJQTYDAIGHLE",
    "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-
cluster",
    "BackupRetentionPeriod": 3,
    "HostedZoneId": "ZNKXH85TT8WVW",
    "StorageEncrypted": false,
    "EnabledCloudwatchLogsExports": [
        "audit"
    ],
    "AssociatedRoles": [],
    "EngineVersion": "3.6.0",
    "Port": 27017,
    "Status": "available"
  }
]
}

```

Para obtener más información, consulte [Descripción de los clústeres de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [DescribeDbClusters](#) de AWS CLI comandos.

## describe-db-engine-versions

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-engine-versions`.

### AWS CLI

Para enumerar las versiones disponibles del motor Amazon DocumentDB

En el siguiente `describe-db-engine-versions` ejemplo, se enumeran todas las versiones disponibles del motor Amazon DocumentDB.

```
aws docdb describe-db-engine-versions \
  --engine docdb
```

Salida:

```
{
  "DBEngineVersions": [
    {
```

```

    "DBEngineVersionDescription": "DocDB version 1.0.200837",
    "DBParameterGroupFamily": "docdb3.6",
    "EngineVersion": "3.6.0",
    "ValidUpgradeTarget": [],
    "DBEngineDescription": "Amazon DocumentDB (with MongoDB compatibility)",
    "SupportsLogExportsToCloudwatchLogs": true,
    "Engine": "docdb",
    "ExportableLogTypes": [
        "audit"
    ]
}
]
}

```

Para obtener más información, consulte [escribeDBEngineLas versiones D](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte [DescribeDbEngineVersions](#) la Referencia de AWS CLI comandos.

## describe-db-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-instances`.

### AWS CLI

Para obtener información sobre las instancias de Amazon DocumentDB aprovisionadas

En el siguiente `describe-db-instances` ejemplo, se muestran detalles sobre la instancia de Amazon DocumentDB. `sample-cluster-instance` Al omitir el `--db-instance-identifier` parámetro, obtiene información sobre un máximo de 100 instancias.

```

aws docdb describe-db-instances \
  --db-instance-identifier sample-cluster-instance

```

Salida:

```

{
  "DBInstances": [
    {
      "Endpoint": {
        "HostedZoneId": "ZNKXH85TT8WVW",

```

```
    "Address": "sample-cluster-instance.corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
    "Port": 27017
  },
  "PreferredBackupWindow": "00:00-00:30",
  "DBInstanceStatus": "available",
  "DBInstanceClass": "db.r4.large",
  "EnabledCloudwatchLogsExports": [
    "audit"
  ],
  "DBInstanceIdentifier": "sample-cluster-instance",
  "DBSubnetGroup": {
    "Subnets": [
      {
        "SubnetStatus": "Active",
        "SubnetIdentifier": "subnet-4e26d263",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2a"
        }
      },
      {
        "SubnetStatus": "Active",
        "SubnetIdentifier": "subnet-afc329f4",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2c"
        }
      },
      {
        "SubnetStatus": "Active",
        "SubnetIdentifier": "subnet-53ab3636",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2d"
        }
      },
      {
        "SubnetStatus": "Active",
        "SubnetIdentifier": "subnet-991cb8d0",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2b"
        }
      }
    ],
    "DBSubnetGroupName": "default",
    "SubnetGroupStatus": "Complete",
```

```
        "DBSubnetGroupDescription": "default",
        "VpcId": "vpc-91280df6"
    },
    "InstanceCreateTime": "2019-03-15T20:36:06.338Z",
    "Engine": "docdb",
    "StorageEncrypted": false,
    "AutoMinorVersionUpgrade": true,
    "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster-
instance",
    "PreferredMaintenanceWindow": "tue:08:39-tue:09:09",
    "VpcSecurityGroups": [
        {
            "Status": "active",
            "VpcSecurityGroupId": "sg-77186e0d"
        }
    ],
    "DBClusterIdentifier": "sample-cluster",
    "PendingModifiedValues": {},
    "BackupRetentionPeriod": 3,
    "PubliclyAccessible": false,
    "EngineVersion": "3.6.0",
    "PromotionTier": 1,
    "AvailabilityZone": "us-west-2c",
    "DbiResourceId": "db-A2GIKUV6KP0HITGGKI2NHVISZA"
    }
]
}
```

Para obtener más información, consulte [Descripción de las instancias de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [DescribeDbInstances](#) de AWS CLI comandos.

## **describe-db-subnet-groups**

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-subnet-groups`.

### AWS CLI

Para recuperar una lista de descripciones de subredes de Amazon DocumentDB

En el siguiente describe-db-subnet-groups ejemplo se describen los detalles de la subred de Amazon DocumentDB denominada. default

```
aws docdb describe-db-subnet-groups \  
--db-subnet-group-name default
```

Salida:

```
{  
  "DBSubnetGroups": [  
    {  
      "VpcId": "vpc-91280df6",  
      "DBSubnetGroupArn": "arn:aws:rds:us-west-2:123456789012:subgrp:default",  
      "Subnets": [  
        {  
          "SubnetIdentifier": "subnet-4e26d263",  
          "SubnetStatus": "Active",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2a"  
          }  
        },  
        {  
          "SubnetIdentifier": "subnet-afc329f4",  
          "SubnetStatus": "Active",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2c"  
          }  
        },  
        {  
          "SubnetIdentifier": "subnet-53ab3636",  
          "SubnetStatus": "Active",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2d"  
          }  
        },  
        {  
          "SubnetIdentifier": "subnet-991cb8d0",  
          "SubnetStatus": "Active",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2b"  
          }  
        }  
      ],  
    }  
  ],  
}
```

```
        "DBSubnetGroupName": "default",
        "SubnetGroupStatus": "Complete",
        "DBSubnetGroupDescription": "default"
    }
]
}
```

Para obtener más información, consulte [Descripción de los grupos de subredes](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [DescribeDbSubnetGroups](#) de AWS CLI comandos.

## **describe-engine-default-cluster-parameters**

En el siguiente ejemplo de código, se muestra cómo usar `describe-engine-default-cluster-parameters`.

### AWS CLI

Para describir la información de los parámetros predeterminados del motor y del sistema para Amazon DocumentDB

En el siguiente `describe-engine-default-cluster-parameters` ejemplo, se muestran los detalles de la información de los parámetros predeterminados del motor y del sistema para el grupo de parámetros de Amazon DocumentDB. `docdb3.6`

```
aws docdb describe-engine-default-cluster-parameters \
  --db-parameter-group-family docdb3.6
```

Salida:

```
{
  "EngineDefaults": {
    "DBParameterGroupFamily": "docdb3.6",
    "Parameters": [
      {
        "ApplyType": "dynamic",
        "ParameterValue": "disabled",
        "Description": "Enables auditing on cluster.",
        "Source": "system",
```

```

        "DataType": "string",
        "MinimumEngineVersion": "3.6.0",
        "AllowedValues": "enabled,disabled",
        "ParameterName": "audit_logs",
        "IsModifiable": true
    },
    {
        "ApplyType": "static",
        "ParameterValue": "enabled",
        "Description": "Config to enable/disable TLS",
        "Source": "system",
        "DataType": "string",
        "MinimumEngineVersion": "3.6.0",
        "AllowedValues": "disabled,enabled",
        "ParameterName": "tls",
        "IsModifiable": true
    },
    {
        "ApplyType": "dynamic",
        "ParameterValue": "enabled",
        "Description": "Enables TTL Monitoring",
        "Source": "system",
        "DataType": "string",
        "MinimumEngineVersion": "3.6.0",
        "AllowedValues": "disabled,enabled",
        "ParameterName": "ttl_monitor",
        "IsModifiable": true
    }
]
}
}

```

Para obtener más información, consulte [DescribeEngineDefaultClusterParameters](#) la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte [DescribeEngineDefaultClusterParameters](#) la Referencia de AWS CLI comandos.

## describe-event-categories

En el siguiente ejemplo de código, se muestra cómo usar `describe-event-categories`.

## AWS CLI

Para describir todas las categorías de eventos de Amazon DocumentDB

En el siguiente `describe-event-categories` ejemplo, se enumeran todas las categorías del tipo de fuente de eventos de Amazon DocumentDB. `db-instance`

```
aws docdb describe-event-categories \  
  --source-type db-cluster
```

Salida:

```
{  
  "EventCategoriesMapList": [  
    {  
      "SourceType": "db-cluster",  
      "EventCategories": [  
        "failover",  
        "maintenance",  
        "notification",  
        "failure"  
      ]  
    }  
  ]  
}
```

Para obtener más información, consulte [Visualización de categorías de eventos](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte [DescribeEventCategories](#) la Referencia de AWS CLI comandos.

## **describe-events**

En el siguiente ejemplo de código, se muestra cómo usar `describe-events`.

## AWS CLI

Para enumerar los eventos de Amazon DocumentDB

En el siguiente `describe-events` ejemplo, se enumeran todos los eventos de Amazon DocumentDB de las últimas 24 horas (1440 minutos).



```
aws docdb describe-events \  
--duration 1440
```

Este comando no genera ninguna salida. Salida:

```
{  
  "Events": [  
    {  
      "EventCategories": [  
        "failover"  
      ],  
      "Message": "Started cross AZ failover to DB instance: sample-cluster",  
      "Date": "2019-03-18T21:36:29.807Z",  
      "SourceArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-  
cluster",  
      "SourceIdentifier": "sample-cluster",  
      "SourceType": "db-cluster"  
    },  
    {  
      "EventCategories": [  
        "availability"  
      ],  
      "Message": "DB instance restarted",  
      "Date": "2019-03-18T21:36:40.793Z",  
      "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster",  
      "SourceIdentifier": "sample-cluster",  
      "SourceType": "db-instance"  
    },  
    {  
      "EventCategories": [],  
      "Message": "A new writer was promoted. Restarting database as a  
reader.",  
      "Date": "2019-03-18T21:36:43.873Z",  
      "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",  
      "SourceIdentifier": "sample-cluster2",  
      "SourceType": "db-instance"  
    },  
    {  
      "EventCategories": [  
        "availability"  
      ],  
      "Message": "DB instance restarted",  
      "Date": "2019-03-18T21:36:51.257Z",
```

```
    "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
    "SourceIdentifier": "sample-cluster2",
    "SourceType": "db-instance"
  },
  {
    "EventCategories": [
      "failover"
    ],
    "Message": "Completed failover to DB instance: sample-cluster",
    "Date": "2019-03-18T21:36:53.462Z",
    "SourceArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-
cluster",
    "SourceIdentifier": "sample-cluster",
    "SourceType": "db-cluster"
  },
  {
    "Date": "2019-03-19T16:51:48.847Z",
    "EventCategories": [
      "configuration change"
    ],
    "Message": "Updated parameter audit_logs to enabled with apply method
pending-reboot",
    "SourceIdentifier": "custom3-6-param-grp",
    "SourceType": "db-parameter-group"
  },
  {
    "EventCategories": [
      "configuration change"
    ],
    "Message": "Applying modification to database instance class",
    "Date": "2019-03-19T17:55:20.095Z",
    "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
    "SourceIdentifier": "sample-cluster2",
    "SourceType": "db-instance"
  },
  {
    "EventCategories": [
      "availability"
    ],
    "Message": "DB instance shutdown",
    "Date": "2019-03-19T17:56:31.127Z",
    "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
    "SourceIdentifier": "sample-cluster2",
    "SourceType": "db-instance"
  }
]
```

```
  },
  {
    "EventCategories": [
      "configuration change"
    ],
    "Message": "Finished applying modification to DB instance class",
    "Date": "2019-03-19T18:00:45.822Z",
    "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
    "SourceIdentifier": "sample-cluster2",
    "SourceType": "db-instance"
  },
  {
    "EventCategories": [
      "availability"
    ],
    "Message": "DB instance restarted",
    "Date": "2019-03-19T18:00:53.397Z",
    "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
    "SourceIdentifier": "sample-cluster2",
    "SourceType": "db-instance"
  },
  {
    "EventCategories": [
      "availability"
    ],
    "Message": "DB instance shutdown",
    "Date": "2019-03-19T18:23:36.045Z",
    "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
    "SourceIdentifier": "sample-cluster2",
    "SourceType": "db-instance"
  },
  {
    "EventCategories": [
      "availability"
    ],
    "Message": "DB instance restarted",
    "Date": "2019-03-19T18:23:46.209Z",
    "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
    "SourceIdentifier": "sample-cluster2",
    "SourceType": "db-instance"
  },
  {
    "Date": "2019-03-19T18:39:05.822Z",
    "EventCategories": [
```

```

        "configuration change"
    ],
    "Message": "Updated parameter ttl_monitor to enabled with apply method
immediate",
    "SourceIdentifier": "custom3-6-param-grp",
    "SourceType": "db-parameter-group"
  },
  {
    "Date": "2019-03-19T18:39:48.067Z",
    "EventCategories": [
      "configuration change"
    ],
    "Message": "Updated parameter audit_logs to disabled with apply method
immediate",
    "SourceIdentifier": "custom3-6-param-grp",
    "SourceType": "db-parameter-group"
  }
]
}

```

Para obtener más información, consulte [Visualización de eventos de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [DescribeEvents](#) de AWS CLI comandos.

## describe-orderable-db-instance-options

En el siguiente ejemplo de código, se muestra cómo usar `describe-orderable-db-instance-options`.

AWS CLI

Para encontrar las opciones de instancia de Amazon DocumentDB, puede solicitar

En el siguiente `describe-orderable-db-instance-options` ejemplo, se enumeran todas las opciones de instancia de Amazon DocumentDB para una región.

```

aws docdb describe-orderable-db-instance-options \
  --engine docdb \
  --region us-east-1

```

**Salida:**

```
{
  "OrderableDBInstanceOptions": [
    {
      "Vpc": true,
      "AvailabilityZones": [
        {
          "Name": "us-east-1a"
        },
        {
          "Name": "us-east-1b"
        },
        {
          "Name": "us-east-1c"
        },
        {
          "Name": "us-east-1d"
        }
      ],
      "EngineVersion": "3.6.0",
      "DBInstanceClass": "db.r4.16xlarge",
      "LicenseModel": "na",
      "Engine": "docdb"
    },
    {
      "Vpc": true,
      "AvailabilityZones": [
        {
          "Name": "us-east-1a"
        },
        {
          "Name": "us-east-1b"
        },
        {
          "Name": "us-east-1c"
        },
        {
          "Name": "us-east-1d"
        }
      ],
      "EngineVersion": "3.6.0",
      "DBInstanceClass": "db.r4.2xlarge",
    }
  ]
}
```

```
    "LicenseModel": "na",
    "Engine": "docdb"
  },
  {
    "Vpc": true,
    "AvailabilityZones": [
      {
        "Name": "us-east-1a"
      },
      {
        "Name": "us-east-1b"
      },
      {
        "Name": "us-east-1c"
      },
      {
        "Name": "us-east-1d"
      }
    ],
    "EngineVersion": "3.6.0",
    "DBInstanceClass": "db.r4.4xlarge",
    "LicenseModel": "na",
    "Engine": "docdb"
  },
  {
    "Vpc": true,
    "AvailabilityZones": [
      {
        "Name": "us-east-1a"
      },
      {
        "Name": "us-east-1b"
      },
      {
        "Name": "us-east-1c"
      },
      {
        "Name": "us-east-1d"
      }
    ],
    "EngineVersion": "3.6.0",
    "DBInstanceClass": "db.r4.8xlarge",
    "LicenseModel": "na",
    "Engine": "docdb"
  }
]
```

```
    },
    {
      "Vpc": true,
      "AvailabilityZones": [
        {
          "Name": "us-east-1a"
        },
        {
          "Name": "us-east-1b"
        },
        {
          "Name": "us-east-1c"
        },
        {
          "Name": "us-east-1d"
        }
      ],
      "EngineVersion": "3.6.0",
      "DBInstanceClass": "db.r4.large",
      "LicenseModel": "na",
      "Engine": "docdb"
    },
    {
      "Vpc": true,
      "AvailabilityZones": [
        {
          "Name": "us-east-1a"
        },
        {
          "Name": "us-east-1b"
        },
        {
          "Name": "us-east-1c"
        },
        {
          "Name": "us-east-1d"
        }
      ],
      "EngineVersion": "3.6.0",
      "DBInstanceClass": "db.r4.xlarge",
      "LicenseModel": "na",
      "Engine": "docdb"
    }
  ]
}
```

```
}
```

Para obtener más información, consulte [Añadir una instancia de Amazon DocumentDB a un clúster](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [DescribeOrderableDbInstanceOptions](#) de AWS CLI comandos.

## **describe-pending-maintenance-actions**

En el siguiente ejemplo de código, se muestra cómo usar `describe-pending-maintenance-actions`.

AWS CLI

Para enumerar las acciones de mantenimiento pendientes de Amazon DocumentDB

En el siguiente `describe-pending-maintenance-actions` ejemplo, se enumeran todas las acciones de mantenimiento pendientes de Amazon DocumentDB.

```
aws docdb describe-pending-maintenance-actions
```

Salida:

```
{
  "PendingMaintenanceActions": []
}
```

Para obtener más información, consulte [Mantenimiento de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [DescribePendingMaintenanceActions](#) de AWS CLI comandos.

## **failover-db-cluster**

En el siguiente ejemplo de código, se muestra cómo usar `failover-db-cluster`.



## AWS CLI

Para forzar a un clúster de Amazon DocumentDB a realizar una conmutación por error a una réplica

El siguiente `failover-db-cluster` ejemplo hace que la instancia principal del clúster `sample-cluster` de Amazon DocumentDB realice una conmutación por error a una réplica.

```
aws docdb failover-db-cluster \  
  --db-cluster-identifier sample-cluster
```

Salida:

```
{  
  "DBCluster": {  
    "AssociatedRoles": [],  
    "DBClusterIdentifier": "sample-cluster",  
    "EngineVersion": "3.6.0",  
    "DBSubnetGroup": "default",  
    "MasterUsername": "master-user",  
    "EarliestRestorableTime": "2019-03-15T20:30:47.020Z",  
    "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-  
west-2.docdb.amazonaws.com",  
    "AvailabilityZones": [  
      "us-west-2a",  
      "us-west-2c",  
      "us-west-2b"  
    ],  
    "LatestRestorableTime": "2019-03-18T21:35:23.548Z",  
    "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",  
    "PreferredBackupWindow": "00:00-00:30",  
    "Port": 27017,  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-77186e0d",  
        "Status": "active"  
      }  
    ],  
    "StorageEncrypted": false,  
    "ClusterCreateTime": "2019-03-15T20:29:58.836Z",  
    "MultiAZ": true,  
    "Status": "available",  
    "DBClusterMembers": [  

```

```

    {
      "DBClusterParameterGroupStatus": "in-sync",
      "IsClusterWriter": false,
      "DBInstanceIdentifier": "sample-cluster",
      "PromotionTier": 1
    },
    {
      "DBClusterParameterGroupStatus": "in-sync",
      "IsClusterWriter": true,
      "DBInstanceIdentifier": "sample-cluster2",
      "PromotionTier": 2
    }
  ],
  "EnabledCloudwatchLogsExports": [
    "audit"
  ],
  "DBClusterParameterGroup": "default.docdb3.6",
  "HostedZoneId": "ZNKXH85TT8WVW",
  "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster",
  "BackupRetentionPeriod": 3,
  "DbClusterResourceId": "cluster-UP4EF2PVDDFVHHDJQTYDAIGHLE",
  "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
  "Engine": "docdb"
}
}

```

Para obtener más información, consulte [Amazon DocumentDB Failover](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia de [FailoverDbCluster](#) comandos AWS CLI .

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar todas las etiquetas de un recurso de Amazon DocumentDB

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas del clúster de Amazon DocumentDB. `sample-cluster`

```
aws docdb list-tags-for-resource \  
  --resource-name arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster
```

Salida:

```
{  
  "TagList": [  
    {  
      "Key": "A",  
      "Value": "ALPHA"  
    },  
    {  
      "Key": "B",  
      "Value": ""  
    },  
    {  
      "Key": "C",  
      "Value": "CHARLIE"  
    }  
  ]  
}
```

Para obtener más información, consulte [Publicar etiquetas en un recurso de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [ListTagsForResource](#) de AWS CLI comandos.

## modify-db-cluster-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-cluster-parameter-group`.

### AWS CLI

Para modificar un grupo de parámetros de un clúster de base de datos Amazon DocumentDB

El siguiente `modify-db-cluster-parameter-group` ejemplo modifica el `custom3-6-param-grp` grupo de parámetros del clúster de Amazon DocumentDB estableciendo los dos `audit_logs` parámetros `ttl_monitor` y activándolos. Los cambios se aplican en el siguiente reinicio.

```
aws docdb modify-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name custom3-6-param-grp \  
  --  
parameters ParameterName=audit_logs,ParameterValue=enabled,ApplyMethod=pending-  
reboot \  
  
ParameterName=ttl_monitor,ParameterValue=enabled,ApplyMethod=pending-reboot
```

Salida:

```
{  
  "DBClusterParameterGroupName": "custom3-6-param-grp"  
}
```

Para obtener más información, consulte [Modificación de un grupo de parámetros de clúster de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [ModifyDbClusterParameterGroup](#) de AWS CLI comandos.

## modify-db-cluster-snapshot-attribute

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-cluster-snapshot-attribute`.

AWS CLI

Ejemplo 1: Para añadir un atributo a una instantánea de Amazon DocumentDB

En el siguiente `modify-db-cluster-snapshot-attribute` ejemplo, se añaden cuatro valores de atributo a una instantánea de clúster de Amazon DocumentDB.

```
aws docdb modify-db-cluster-snapshot-attribute \  
  --db-cluster-snapshot-identifier sample-cluster-snapshot \  
  --attribute-name restore \  
  --values-to-add 123456789011 123456789012 123456789013
```

Salida:

```
{
```

```

"DBClusterSnapshotAttributesResult": {
  "DBClusterSnapshotAttributes": [
    {
      "AttributeName": "restore",
      "AttributeValues": [
        "123456789011",
        "123456789012",
        "123456789013"
      ]
    }
  ],
  "DBClusterSnapshotIdentifier": "sample-cluster-snapshot"
}

```

Ejemplo 2: Para eliminar atributos de una instantánea de Amazon DocumentDB

El siguiente `modify-db-cluster-snapshot-attribute` ejemplo elimina dos valores de atributo de una instantánea de clúster de Amazon DocumentDB.

```

aws docdb modify-db-cluster-snapshot-attribute \
  --db-cluster-snapshot-identifier sample-cluster-snapshot \
  --attribute-name restore \
  --values-to-remove 123456789012

```

Salida:

```

{
  "DBClusterSnapshotAttributesResult": {
    "DBClusterSnapshotAttributes": [
      {
        "AttributeName": "restore",
        "AttributeValues": [
          "123456789011",
          "123456789013"
        ]
      }
    ],
    "DBClusterSnapshotIdentifier": "sample-cluster-snapshot"
  }
}

```

Para obtener más información, consulte [ModifyDBCluster SnapshotAttribute](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte [ModifyDbClusterSnapshotAttribute](#) la Referencia de AWS CLI comandos.

## modify-db-cluster

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-cluster`.

### AWS CLI

Para modificar un clúster de Amazon DocumentDB

El siguiente `modify-db-cluster` ejemplo modifica el `sample-cluster` clúster de Amazon DocumentDB haciendo que el período de retención de las copias de seguridad automáticas sea de 7 días y cambiando las ventanas preferidas tanto para las copias de seguridad como para el mantenimiento. Todos los cambios se aplican en el siguiente período de mantenimiento.

```
aws docdb modify-db-cluster \  
  --db-cluster-identifier sample-cluster \  
  --no-apply-immediately \  
  --backup-retention-period 7 \  
  --preferred-backup-window 18:00-18:30 \  
  --preferred-maintenance-window sun:20:00-sun:20:30
```

Salida:

```
{  
  "DBCluster": {  
    "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-  
west-2.docdb.amazonaws.com",  
    "DBClusterMembers": [  
      {  
        "DBClusterParameterGroupStatus": "in-sync",  
        "DBInstanceIdentifier": "sample-cluster",  
        "IsClusterWriter": true,  
        "PromotionTier": 1  
      },  
      {  
        "DBClusterParameterGroupStatus": "in-sync",  
        "DBInstanceIdentifier": "sample-cluster2",
```

```
        "IsClusterWriter": false,
        "PromotionTier": 2
    }
],
"HostedZoneId": "ZNKXH85TT8WWW",
"StorageEncrypted": false,
"PreferredBackupWindow": "18:00-18:30",
"MultiAZ": true,
"EngineVersion": "3.6.0",
"MasterUsername": "master-user",
"ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
"DBSubnetGroup": "default",
"LatestRestorableTime": "2019-03-18T22:08:13.408Z",
"EarliestRestorableTime": "2019-03-15T20:30:47.020Z",
"PreferredMaintenanceWindow": "sun:20:00-sun:20:30",
"AssociatedRoles": [],
"EnabledCloudwatchLogsExports": [
    "audit"
],
"Engine": "docdb",
"DBClusterParameterGroup": "default.docdb3.6",
"DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster",
"BackupRetentionPeriod": 7,
"DBClusterIdentifier": "sample-cluster",
"AvailabilityZones": [
    "us-west-2a",
    "us-west-2c",
    "us-west-2b"
],
"Status": "available",
"DbClusterResourceId": "cluster-UP4EF2PVDDFVHHDJQTYDAIGHLE",
"ClusterCreateTime": "2019-03-15T20:29:58.836Z",
"VpcSecurityGroups": [
    {
        "VpcSecurityGroupId": "sg-77186e0d",
        "Status": "active"
    }
],
"Port": 27017
}
}
```

Para obtener más información, consulte [Modificación de un clúster de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [ModifyDbCluster](#) de AWS CLI comandos.

## modify-db-instance

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-instance`.

### AWS CLI

Para modificar una instancia de Amazon DocumentDB

El siguiente `modify-db-instance` ejemplo modifica la `sample-cluster2` instancia de Amazon DocumentDB cambiando su clase de instancia `db.r4.4xlarge` a `y` su nivel de promoción a `5`. Los cambios se aplican inmediatamente, pero solo se pueden ver una vez que el estado de la instancia esté disponible.

```
aws docdb modify-db-instance \  
  --db-instance-identifier sample-cluster2 \  
  --apply-immediately \  
  --db-instance-class db.r4.4xlarge \  
  --promotion-tier 5
```

Salida:

```
{  
  "DBInstance": {  
    "EngineVersion": "3.6.0",  
    "StorageEncrypted": false,  
    "DBInstanceClass": "db.r4.large",  
    "PreferredMaintenanceWindow": "mon:08:39-mon:09:09",  
    "AutoMinorVersionUpgrade": true,  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-77186e0d",  
        "Status": "active"  
      }  
    ],  
    "PreferredBackupWindow": "18:00-18:30",  
    "EnabledCloudwatchLogsExports": [  

```



```
    "audit"
  ],
  "AvailabilityZone": "us-west-2f",
  "DBInstanceIdentifier": "sample-cluster2",
  "InstanceCreateTime": "2019-03-15T20:36:06.338Z",
  "Engine": "docdb",
  "BackupRetentionPeriod": 7,
  "DBSubnetGroup": {
    "DBSubnetGroupName": "default",
    "DBSubnetGroupDescription": "default",
    "SubnetGroupStatus": "Complete",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-4e26d263",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2a"
        },
        "SubnetStatus": "Active"
      },
      {
        "SubnetIdentifier": "subnet-afc329f4",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2c"
        },
        "SubnetStatus": "Active"
      },
      {
        "SubnetIdentifier": "subnet-53ab3636",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2d"
        },
        "SubnetStatus": "Active"
      },
      {
        "SubnetIdentifier": "subnet-991cb8d0",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2b"
        },
        "SubnetStatus": "Active"
      }
    ]
  },
  "VpcId": "vpc-91280df6"
},
"PromotionTier": 2,
```

```

    "Endpoint": {
      "Address": "sample-cluster2.corcjozrlsfc.us-west-2.docdb.amazonaws.com",
      "HostedZoneId": "ZNKXH85TT8WW",
      "Port": 27017
    },
    "DbiResourceId": "db-A2GIKUV6KPOHITGGKI2NHVISZA",
    "DBClusterIdentifier": "sample-cluster",
    "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
    "PendingModifiedValues": {
      "DBInstanceClass": "db.r4.4xlarge"
    },
    "PubliclyAccessible": false,
    "DBInstanceStatus": "available"
  }
}

```

Para obtener más información, consulte [Modificación de una instancia de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [ModifyDbInstance](#) de AWS CLI comandos.

## modify-db-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-subnet-group`.

### AWS CLI

Para modificar un grupo de subredes de Amazon DocumentDB

El siguiente `modify-db-subnet-group` ejemplo modifica el grupo de subredes `sample-subnet-group` añadiendo las subredes especificadas y una nueva descripción.

```

aws docdb modify-db-subnet-group \
  --db-subnet-group-name sample-subnet-group \
  --subnet-ids subnet-b3806e8f subnet-53ab3636 subnet-991cb8d0 \
  --db-subnet-group-description "New subnet description"

```

Salida:

```
{
```

```
"DBSubnetGroup": {
  "DBSubnetGroupName": "sample-subnet-group",
  "SubnetGroupStatus": "Complete",
  "DBSubnetGroupArn": "arn:aws:rds:us-west-2:123456789012:subgrp:sample-
subnet-group",
  "VpcId": "vpc-91280df6",
  "DBSubnetGroupDescription": "New subnet description",
  "Subnets": [
    {
      "SubnetIdentifier": "subnet-b3806e8f",
      "SubnetStatus": "Active",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2a"
      }
    },
    {
      "SubnetIdentifier": "subnet-53ab3636",
      "SubnetStatus": "Active",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2c"
      }
    },
    {
      "SubnetIdentifier": "subnet-991cb8d0",
      "SubnetStatus": "Active",
      "SubnetAvailabilityZone": {
        "Name": "us-west-2b"
      }
    }
  ]
}
```

Para obtener más información, consulte [Modificación de un grupo de subredes de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia de [ModifyDbSubnetGroup](#) comandos AWS CLI .

## **reboot-db-instance**

En el siguiente ejemplo de código, se muestra cómo usar `reboot-db-instance`.

## AWS CLI

Para reiniciar una instancia de Amazon DocumentDB

En el siguiente `reboot-db-instance` ejemplo, se reinicia la instancia de Amazon `sample-cluster2` DocumentDB.

```
aws docdb reboot-db-instance \  
  --db-instance-identifier sample-cluster2
```

Este comando no genera ninguna salida. Salida:

```
{  
  "DBInstance": {  
    "PreferredBackupWindow": "18:00-18:30",  
    "DBInstanceIdentifier": "sample-cluster2",  
    "VpcSecurityGroups": [  
      {  
        "Status": "active",  
        "VpcSecurityGroupId": "sg-77186e0d"  
      }  
    ],  
    "DBSubnetGroup": {  
      "VpcId": "vpc-91280df6",  
      "Subnets": [  
        {  
          "SubnetStatus": "Active",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2a"  
          },  
          "SubnetIdentifier": "subnet-4e26d263"  
        },  
        {  
          "SubnetStatus": "Active",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2c"  
          },  
          "SubnetIdentifier": "subnet-afc329f4"  
        },  
        {  
          "SubnetStatus": "Active",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2d"  
          }  
        }  
      ]  
    }  
  }  
}
```

```
    },
    "SubnetIdentifier": "subnet-53ab3636"
  },
  {
    "SubnetStatus": "Active",
    "SubnetAvailabilityZone": {
      "Name": "us-west-2b"
    },
    "SubnetIdentifier": "subnet-991cb8d0"
  }
],
"SubnetGroupStatus": "Complete",
"DBSubnetGroupName": "default",
"DBSubnetGroupDescription": "default"
},
"PendingModifiedValues": {},
"Endpoint": {
  "Address": "sample-cluster2.corcjozrlsfc.us-west-2.docdb.amazonaws.com",
  "HostedZoneId": "ZNKXH85TT8WW",
  "Port": 27017
},
"EnabledCloudwatchLogsExports": [
  "audit"
],
"StorageEncrypted": false,
"DbiResourceId": "db-A2GIKUV6KPOHITGGKI2NHVISZA",
"AutoMinorVersionUpgrade": true,
"Engine": "docdb",
"InstanceCreateTime": "2019-03-15T20:36:06.338Z",
"EngineVersion": "3.6.0",
"PromotionTier": 5,
"BackupRetentionPeriod": 7,
"DBClusterIdentifier": "sample-cluster",
"PreferredMaintenanceWindow": "mon:08:39-mon:09:09",
"PubliclyAccessible": false,
"DBInstanceClass": "db.r4.4xlarge",
"AvailabilityZone": "us-west-2d",
"DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
"DBInstanceStatus": "rebooting"
}
}
```

Para obtener más información, consulte [Reiniciar una Amazon Instance DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para obtener API más información, consulte la Referencia de comandos [RebootDbInstance](#).AWS CLI

## **remove-tags-from-resource**

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags-from-resource`.

### AWS CLI

Para eliminar etiquetas de un recurso de Amazon DocumentDB

En el siguiente `remove-tags-from-resource` ejemplo, se elimina la etiqueta con el nombre B de clave del clúster de Amazon DocumentDB. `sample-cluster`

```
aws docdb remove-tags-from-resource \  
  --resource-name arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster \  
  --tag-keys B
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar etiquetas de un Amazon DocumentDBResource](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte [RemoveTagsFromResource](#) la Referencia de AWS CLI comandos.

## **reset-db-cluster-parameter-group**

En el siguiente ejemplo de código, se muestra cómo usar `reset-db-cluster-parameter-group`.

### AWS CLI

Para restablecer el valor del parámetro especificado a sus valores predeterminados en un grupo de parámetros de Amazon DocumentDB

En el siguiente `reset-db-cluster-parameter-group` ejemplo, se restablece el valor `ttl_monitor` predeterminado del parámetro del `custom3-6-param-grp` grupo de parámetros de Amazon DocumentDB.

```
aws docdb reset-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name custom3-6-param-grp \  
  --parameters ParameterName=ttl_monitor,ApplyMethod=immediate
```

Salida:

```
{  
  "DBClusterParameterGroupName": "custom3-6-param-grp"  
}
```

Para obtener más información, consulte el título en la Guía para desarrolladores de Amazon DocumentDB.

Para restablecer los valores de parámetros especificados o de todos los parámetros a sus valores predeterminados en un grupo de parámetros de Amazon DocumentDB

El siguiente `reset-db-cluster-parameter-group` ejemplo restablece todos los parámetros del `custom3-6-param-grp` grupo de parámetros de Amazon DocumentDB a su valor predeterminado.

```
aws docdb reset-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name custom3-6-param-grp \  
  --reset-all-parameters
```

Salida:

```
{  
  "DBClusterParameterGroupName": "custom3-6-param-grp"  
}
```

Para obtener más información, consulte [Restablecimiento de un grupo de parámetros de clúster de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia de [ResetDbClusterParameterGroup](#) comandos AWS CLI .

## **restore-db-cluster-from-snapshot**

En el siguiente ejemplo de código, se muestra cómo usar `restore-db-cluster-from-snapshot`.

## AWS CLI

Para restaurar un clúster de Amazon DocumentDB a partir de una instantánea automática o manual

En el siguiente `restore-db-cluster-from-snapshot` ejemplo, se crea un nuevo clúster de Amazon DocumentDB cuyo nombre `sample-cluster-2019-03-16-00-01-restored` proviene de la instantánea. `rds:sample-cluster-2019-03-16-00-01`

```
aws docdb restore-db-cluster-from-snapshot \  
  --db-cluster-identifier sample-cluster-2019-03-16-00-01-restored \  
  --engine docdb \  
  --snapshot-identifier rds:sample-cluster-2019-03-16-00-01
```

Salida:

```
{  
  "DBCluster": {  
    "ClusterCreateTime": "2019-03-19T18:45:01.857Z",  
    "HostedZoneId": "ZNKXH85TT8WVW",  
    "Engine": "docdb",  
    "DBClusterMembers": [],  
    "MultiAZ": false,  
    "AvailabilityZones": [  
      "us-west-2a",  
      "us-west-2c",  
      "us-west-2b"  
    ],  
    "StorageEncrypted": false,  
    "ReaderEndpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-ro-  
corcjorzrlsfc.us-west-2.docdb.amazonaws.com",  
    "Endpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-  
corcjorzrlsfc.us-west-2.docdb.amazonaws.com",  
    "Port": 27017,  
    "PreferredBackupWindow": "00:00-00:30",  
    "DBSubnetGroup": "default",  
    "DBClusterIdentifier": "sample-cluster-2019-03-16-00-01-restored",  
    "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",  
    "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-  
cluster-2019-03-16-00-01-restored",  
    "DBClusterParameterGroup": "default.docdb3.6",  
    "DbClusterResourceId": "cluster-X0046Q3RH4LWSYNH3NMZKXPISU",  
    "MasterUsername": "master-user",
```



```

    "EngineVersion": "3.6.0",
    "BackupRetentionPeriod": 3,
    "AssociatedRoles": [],
    "Status": "creating",
    "VpcSecurityGroups": [
      {
        "Status": "active",
        "VpcSecurityGroupId": "sg-77186e0d"
      }
    ]
  }
}

```

Para obtener más información, consulte [Restauración desde una instantánea de clúster](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte [RestoreDbClusterFromSnapshot](#) la Referencia de AWS CLI comandos.

## restore-db-cluster-to-point-in-time

En el siguiente ejemplo de código, se muestra cómo usar `restore-db-cluster-to-point-in-time`.

### AWS CLI

Para restaurar un clúster de Amazon DocumentDB a point-in-time partir de una instantánea manual

En el siguiente `restore-db-cluster-to-point-in-time` ejemplo, se utiliza `sample-cluster-snapshot` para crear un nuevo clúster de Amazon DocumentDB `sample-cluster-pit`, utilizando la última hora de restauración.

```

aws docdb restore-db-cluster-to-point-in-time \
  --db-cluster-identififer sample-cluster-pit \
  --source-db-cluster-identififer arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster \
  --use-latest-restorable-time

```

Salida:

```
{
```

```

"DBCluster": {
  "StorageEncrypted": false,
  "BackupRetentionPeriod": 3,
  "MasterUsername": "master-user",
  "HostedZoneId": "ZNKXH85TT8WWW",
  "PreferredBackupWindow": "00:00-00:30",
  "MultiAZ": false,
  "DBClusterIdentifier": "sample-cluster-pit",
  "DBSubnetGroup": "default",
  "ClusterCreateTime": "2019-04-03T15:55:21.320Z",
  "AssociatedRoles": [],
  "DBClusterParameterGroup": "default.docdb3.6",
  "DBClusterMembers": [],
  "Status": "creating",
  "AvailabilityZones": [
    "us-west-2a",
    "us-west-2d",
    "us-west-2b"
  ],
  "ReaderEndpoint": "sample-cluster-pit.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
  "Port": 27017,
  "Engine": "docdb",
  "EngineVersion": "3.6.0",
  "VpcSecurityGroups": [
    {
      "VpcSecurityGroupId": "sg-77186e0d",
      "Status": "active"
    }
  ],
  "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
  "Endpoint": "sample-cluster-pit.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
  "DbClusterResourceId": "cluster-NLCABBXOSE2QPQ4GOLZIFWEPLM",
  "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster-
pit"
}
}

```

Para obtener más información, consulte [Restauración de una instantánea a un punto en el tiempo en](#) la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte [RestoreDbClusterToPointInTime](#) la Referencia de AWS CLI comandos.

## start-db-cluster

En el siguiente ejemplo de código, se muestra cómo usar `start-db-cluster`.

### AWS CLI

Para iniciar un clúster de Amazon DocumentDB detenido

En el siguiente `start-db-cluster` ejemplo, se inicia el clúster de Amazon DocumentDB especificado.

```
aws docdb start-db-cluster \  
  --db-cluster-identifier sample-cluster
```

Salida:

```
{  
  "DBCluster": {  
    "ClusterCreateTime": "2019-03-19T18:45:01.857Z",  
    "HostedZoneId": "ZNKXH85TT8WVW",  
    "Engine": "docdb",  
    "DBClusterMembers": [],  
    "MultiAZ": false,  
    "AvailabilityZones": [  
      "us-east-1a",  
      "us-east-1c",  
      "us-east-1f"  
    ],  
    "StorageEncrypted": false,  
    "ReaderEndpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-ro-  
corcjorzrlsfc.us-east-1.docdb.amazonaws.com",  
    "Endpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-  
corcjorzrlsfc.us-east-1.docdb.amazonaws.com",  
    "Port": 27017,  
    "PreferredBackupWindow": "00:00-00:30",  
    "DBSubnetGroup": "default",  
    "DBClusterIdentifier": "sample-cluster-2019-03-16-00-01-restored",  
    "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",  
    "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:sample-  
cluster-2019-03-16-00-01-restored",  
    "DBClusterParameterGroup": "default.docdb3.6",  
    "DbClusterResourceId": "cluster-X0046Q3RH4LWSYNH3NMZKXPISU",  
    "MasterUsername": "master-user",
```

```

    "EngineVersion": "3.6.0",
    "BackupRetentionPeriod": 3,
    "AssociatedRoles": [],
    "Status": "creating",
    "VpcSecurityGroups": [
      {
        "Status": "active",
        "VpcSecurityGroupId": "sg-77186e0d"
      }
    ]
  }
}

```

Para obtener más información, consulte [Detener e iniciar un clúster de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [StartDbCluster](#) de AWS CLI comandos.

## stop-db-cluster

En el siguiente ejemplo de código, se muestra cómo usar `stop-db-cluster`.

### AWS CLI

Para detener un clúster de Amazon DocumentDB en ejecución

El siguiente `stop-db-cluster` ejemplo detiene el clúster de Amazon DocumentDB especificado.

```

aws docdb stop-db-cluster \
  --db-cluster-identifier sample-cluster

```

Salida:

```

{
  "DBCluster": {
    "ClusterCreateTime": "2019-03-19T18:45:01.857Z",
    "HostedZoneId": "ZNKXH85TT8WVW",
    "Engine": "docdb",
    "DBClusterMembers": [],
    "MultiAZ": false,
    "AvailabilityZones": [

```

```

        "us-east-1a",
        "us-east-1c",
        "us-east-1f"
    ],
    "StorageEncrypted": false,
    "ReaderEndpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-ro-
corcjzrlsfc.us-east-1.docdb.amazonaws.com",
    "Endpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-
corcjzrlsfc.us-east-1.docdb.amazonaws.com",
    "Port": 27017,
    "PreferredBackupWindow": "00:00-00:30",
    "DBSubnetGroup": "default",
    "DBClusterIdentifier": "sample-cluster-2019-03-16-00-01-restored",
    "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
    "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:sample-
cluster-2019-03-16-00-01-restored",
    "DBClusterParameterGroup": "default.docdb3.6",
    "DbClusterResourceId": "cluster-X0046Q3RH4LWSYNH3NMZKXPISU",
    "MasterUsername": "master-user",
    "EngineVersion": "3.6.0",
    "BackupRetentionPeriod": 3,
    "AssociatedRoles": [],
    "Status": "creating",
    "VpcSecurityGroups": [
        {
            "Status": "active",
            "VpcSecurityGroupId": "sg-77186e0d"
        }
    ]
}
}

```

Para obtener más información, consulte [Detener e iniciar un clúster de Amazon DocumentDB](#) en la Guía para desarrolladores de Amazon DocumentDB.

- Para API obtener más información, consulte la Referencia [StopDbCluster](#) de AWS CLI comandos.

## Ejemplos de DynamoDB que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante DynamoDB. AWS Command Line Interface

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **batch-get-item**

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-item`.

#### AWS CLI

Recuperación de varios elementos de una tabla

En el siguiente ejemplo `batch-get-items`, se leen varios elementos de la tabla `MusicCollection` mediante un lote de tres solicitudes `GetItem` y se solicita el número de unidades de capacidad de lectura consumidas por la operación. El comando devuelve solo el atributo `AlbumTitle`.

```
aws dynamodb batch-get-item \  
  --request-items file://request-items.json \  
  --return-consumed-capacity TOTAL
```

Contenidos de `request-items.json`:

```
{  
  "MusicCollection": {  
    "Keys": [  
      {  
        "Artist": {"S": "No One You Know"},  
        "SongTitle": {"S": "Call Me Today"}  
      },  
      {  
        "Artist": {"S": "Acme Band"},  
        "SongTitle": {"S": "Happy Day"}  
      }  
    ]  
  }  
}
```

```

    },
    {
      "Artist": {"S": "No One You Know"},
      "SongTitle": {"S": "Scared of My Shadow"}
    }
  ],
  "ProjectionExpression": "AlbumTitle"
}
}

```

Salida:

```

{
  "Responses": {
    "MusicCollection": [
      {
        "AlbumTitle": {
          "S": "Somewhat Famous"
        }
      },
      {
        "AlbumTitle": {
          "S": "Blue Sky Blues"
        }
      },
      {
        "AlbumTitle": {
          "S": "Louder Than Ever"
        }
      }
    ]
  },
  "UnprocessedKeys": {},
  "ConsumedCapacity": [
    {
      "TableName": "MusicCollection",
      "CapacityUnits": 1.5
    }
  ]
}

```

Para obtener más información, consulte [Operaciones por lotes](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [BatchGetItem](#) la Referencia de AWS CLI comandos.

## batch-write-item

En el siguiente ejemplo de código, se muestra cómo usar `batch-write-item`.

### AWS CLI

Adición de varios elementos a una tabla

En el siguiente ejemplo de `batch-write-item`, se añaden tres elementos nuevos a la tabla `MusicCollection` mediante un lote de tres solicitudes `PutItem`. También solicita información sobre el número de unidades de capacidad de escritura consumidas por la operación y cualquier colección de elementos modificada por la operación.

```
aws dynamodb batch-write-item \  
  --request-items file://request-items.json \  
  --return-consumed-capacity INDEXES \  
  --return-item-collection-metrics SIZE
```

Contenidos de `request-items.json`:

```
{  
  "MusicCollection": [  
    {  
      "PutRequest": {  
        "Item": {  
          "Artist": {"S": "No One You Know"},  
          "SongTitle": {"S": "Call Me Today"},  
          "AlbumTitle": {"S": "Somewhat Famous"}  
        }  
      }  
    },  
    {  
      "PutRequest": {  
        "Item": {  
          "Artist": {"S": "Acme Band"},  
          "SongTitle": {"S": "Happy Day"},  
          "AlbumTitle": {"S": "Songs About Life"}  
        }  
      }  
    }  
  ]  
}
```



```

    },
    {
      "PutRequest": {
        "Item": {
          "Artist": {"S": "No One You Know"},
          "SongTitle": {"S": "Scared of My Shadow"},
          "AlbumTitle": {"S": "Blue Sky Blues"}
        }
      }
    }
  ]
}

```

Salida:

```

{
  "UnprocessedItems": {},
  "ItemCollectionMetrics": {
    "MusicCollection": [
      {
        "ItemCollectionKey": {
          "Artist": {
            "S": "No One You Know"
          }
        },
        "SizeEstimateRangeGB": [
          0.0,
          1.0
        ]
      },
      {
        "ItemCollectionKey": {
          "Artist": {
            "S": "Acme Band"
          }
        },
        "SizeEstimateRangeGB": [
          0.0,
          1.0
        ]
      }
    ]
  }
},

```

```

    "ConsumedCapacity": [
      {
        "TableName": "MusicCollection",
        "CapacityUnits": 6.0,
        "Table": {
          "CapacityUnits": 3.0
        },
        "LocalSecondaryIndexes": {
          "AlbumTitleIndex": {
            "CapacityUnits": 3.0
          }
        }
      }
    ]
  }
}

```

Para obtener más información, consulte [Operaciones por lotes](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [BatchWriteItem](#) la Referencia de AWS CLI comandos.

## create-backup

En el siguiente ejemplo de código, se muestra cómo usar create-backup.

### AWS CLI

Para crear una copia de seguridad de una tabla de DynamoDB existente

En el siguiente create-backup ejemplo, se crea una copia de seguridad de la MusicCollection tabla.

```

aws dynamodb create-backup \
  --table-name MusicCollection \
  --backup-name MusicCollectionBackup

```

Salida:

```

{
  "BackupDetails": {
    "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection/backup/01576616366715-b4e58d3a",

```

```
    "BackupName": "MusicCollectionBackup",
    "BackupSizeBytes": 0,
    "BackupStatus": "CREATING",
    "BackupType": "USER",
    "BackupCreationDateTime": 1576616366.715
  }
}
```

Para obtener más información, consulte [Backup and Restore on Demand para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [CreateBackup](#).AWS CLI

## create-global-table

En el siguiente ejemplo de código, se muestra cómo usar `create-global-table`.

### AWS CLI

Para crear una tabla global

El siguiente `create-global-table` ejemplo crea una tabla global a partir de dos tablas idénticas en las AWS regiones separadas especificadas.

```
aws dynamodb create-global-table \
  --global-table-name MusicCollection \
  --replication-group RegionName=us-east-2 RegionName=us-east-1 \
  --region us-east-2
```

Salida:

```
{
  "GlobalTableDescription": {
    "ReplicationGroup": [
      {
        "RegionName": "us-east-2"
      },
      {
        "RegionName": "us-east-1"
      }
    ]
  },
}
```

```

    "GlobalTableArn": "arn:aws:dynamodb::123456789012:global-table/
MusicCollection",
    "CreationDateTime": 1576625818.532,
    "GlobalTableStatus": "CREATING",
    "GlobalTableName": "MusicCollection"
  }
}

```

Para obtener más información, consulte las tablas [globales de DynamoDB en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [CreateGlobalTable](#).AWS CLI

## create-table

En el siguiente ejemplo de código, se muestra cómo usar `create-table`.

### AWS CLI

Ejemplo 1: Creación de una tabla con etiquetas

En el siguiente ejemplo `create-table`, se utilizan los atributos y el esquema de claves especificados para crear una tabla denominada `MusicCollection`. En esta tabla se utiliza el rendimiento aprovisionado y se cifra en reposo con el valor predeterminado `AWS . CMK`. El comando también aplica una etiqueta a la tabla, con una clave de `Owner` y un valor de `blueTeam`.

```

aws dynamodb create-table \
  --table-name MusicCollection \
  --attribute-
definitions AttributeName=Artist,AttributeType=S AttributeName=SongTitle,AttributeType=S
\
  --key-
schema AttributeName=Artist,KeyType=HASH AttributeName=SongTitle,KeyType=RANGE \
  --provisioned-throughput ReadCapacityUnits=5,WriteCapacityUnits=5 \
  --tags Key=Owner,Value=blueTeam

```

Salida:

```

{
  "TableDescription": {

```

```

    "AttributeDefinitions": [
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "WriteCapacityUnits": 5,
      "ReadCapacityUnits": 5
    },
    "TableSizeBytes": 0,
    "TableName": "MusicCollection",
    "TableStatus": "CREATING",
    "KeySchema": [
      {
        "KeyType": "HASH",
        "AttributeName": "Artist"
      },
      {
        "KeyType": "RANGE",
        "AttributeName": "SongTitle"
      }
    ],
    "ItemCount": 0,
    "CreationDateTime": "2020-05-26T16:04:41.627000-07:00",
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
    "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}

```

Para obtener más información, consulte [Operaciones básicas con tablas](#) en la Guía para desarrolladores de Amazon DynamoDB.

### Ejemplo 2: Creación de una tabla en modo bajo demanda

En el siguiente ejemplo, se crea una tabla denominada `MusicCollection` mediante el modo bajo demanda, en lugar del modo de rendimiento aprovisionado. Esto resulta útil para tablas con cargas de trabajo impredecibles.

```
aws dynamodb create-table \
  --table-name MusicCollection \
  --attribute-
definitions AttributeName=Artist,AttributeType=S AttributeName=SongTitle,AttributeType=S
\
  --key-
schema AttributeName=Artist,KeyType=HASH AttributeName=SongTitle,KeyType=RANGE \
  --billing-mode PAY_PER_REQUEST
```

Salida:

```
{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],
    "TableName": "MusicCollection",
    "KeySchema": [
      {
        "AttributeName": "Artist",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "SongTitle",
        "KeyType": "RANGE"
      }
    ],
    "TableStatus": "CREATING",
    "CreationDateTime": "2020-05-27T11:44:10.807000-07:00",
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 0,
      "WriteCapacityUnits": 0
    },
    "TableSizeBytes": 0,
    "ItemCount": 0,
```

```

    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
    "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "BillingModeSummary": {
      "BillingMode": "PAY_PER_REQUEST"
    }
  }
}

```

Para obtener más información, consulte [Operaciones básicas con tablas](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 3: Crear una tabla y cifrarla con una tabla gestionada por el cliente CMK

En el siguiente ejemplo, se crea una tabla con un nombre `MusicCollection` y se cifra mediante una tabla gestionada por el cliente. CMK

```

aws dynamodb create-table \
  --table-name MusicCollection \
  --attribute-
definitions AttributeName=Artist,AttributeType=S AttributeName=SongTitle,AttributeType=S
\
  --key-
schema AttributeName=Artist,KeyType=HASH AttributeName=SongTitle,KeyType=RANGE \
  --provisioned-throughput ReadCapacityUnits=5,WriteCapacityUnits=5 \
  --sse-specification Enabled=true,SSEType=KMS,KMSMasterKeyId=abcd1234-abcd-1234-
a123-ab1234a1b234

```

Salida:

```

{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],
    "TableName": "MusicCollection",
    "KeySchema": [

```

```

    {
      "AttributeName": "Artist",
      "KeyType": "HASH"
    },
    {
      "AttributeName": "SongTitle",
      "KeyType": "RANGE"
    }
  ],
  "TableStatus": "CREATING",
  "CreationDateTime": "2020-05-27T11:12:16.431000-07:00",
  "ProvisionedThroughput": {
    "NumberOfDecreasesToday": 0,
    "ReadCapacityUnits": 5,
    "WriteCapacityUnits": 5
  },
  "TableSizeBytes": 0,
  "ItemCount": 0,
  "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
  "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "SSEDescription": {
    "Status": "ENABLED",
    "SSEType": "KMS",
    "KMSMasterKeyArn": "arn:aws:kms:us-west-2:123456789012:key/abcd1234-
abcd-1234-a123-ab1234a1b234"
  }
}
}

```

Para obtener más información, consulte [Operaciones básicas con tablas](#) en la Guía para desarrolladores de Amazon DynamoDB.

#### Ejemplo 4: Creación de una tabla con un índice secundario local

En el siguiente ejemplo, se utilizan los atributos y el esquema de claves especificados para crear una tabla denominada `MusicCollection` con un índice secundario local denominado `AlbumTitleIndex`.

```

aws dynamodb create-table \
  --table-name MusicCollection \
  --attribute-
definitions AttributeName=Artist,AttributeType=S AttributeName=SongTitle,AttributeType=S Att
\

```



```

--key-
schema AttributeName=Artist,KeyType=HASH AttributeName=SongTitle,KeyType=RANGE \
--provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
--local-secondary-indexes \
  "[
    {
      \"IndexName\": \"AlbumTitleIndex\",
      \"KeySchema\": [
        {\"AttributeName\": \"Artist\", \"KeyType\": \"HASH\"},
        {\"AttributeName\": \"AlbumTitle\", \"KeyType\": \"RANGE\"}
      ],
      \"Projection\": {
        \"ProjectionType\": \"INCLUDE\",
        \"NonKeyAttributes\": [\"Genre\", \"Year\"]
      }
    }
  ]"

```

Salida:

```

{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "AlbumTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],
    "TableName": "MusicCollection",
    "KeySchema": [
      {
        "AttributeName": "Artist",
        "KeyType": "HASH"
      },
      {

```

```

        "AttributeName": "SongTitle",
        "KeyType": "RANGE"
    }
],
"TableStatus": "CREATING",
"CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
"ProvisionedThroughput": {
    "NumberOfDecreasesToday": 0,
    "ReadCapacityUnits": 10,
    "WriteCapacityUnits": 5
},
"TableSizeBytes": 0,
"ItemCount": 0,
"TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
"TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"LocalSecondaryIndexes": [
    {
        "IndexName": "AlbumTitleIndex",
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "AlbumTitle",
                "KeyType": "RANGE"
            }
        ],
        "Projection": {
            "ProjectionType": "INCLUDE",
            "NonKeyAttributes": [
                "Genre",
                "Year"
            ]
        },
        "IndexSizeBytes": 0,
        "ItemCount": 0,
        "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitleIndex"
    }
]
}
}

```

Para obtener más información, consulte [Operaciones básicas con tablas](#) en la Guía para desarrolladores de Amazon DynamoDB.

#### Ejemplo 5: Creación de una tabla con un índice secundario global

En el siguiente ejemplo, se crea una tabla llamada `GameScores` con un índice secundario global denominado `GameTitleIndex`. La tabla base tiene una clave de partición `UserId` y una clave de clasificación `GameTitle`, lo que permite encontrar la mejor puntuación de un usuario individual en un juego específico de manera eficiente, mientras que GSI tiene una clave de partición `GameTitle` y otra de clasificación de `TopScore`, lo que permite encontrar rápidamente la puntuación más alta general de un juego en particular.

```
aws dynamodb create-table \
  --table-name GameScores \
  --attribute-
definitions AttributeName=UserId,AttributeType=S AttributeName=GameTitle,AttributeType=S Att
  \
  --key-schema AttributeName=UserId,KeyType=HASH \
                AttributeName=GameTitle,KeyType=RANGE \
  --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
  --global-secondary-indexes \
    "[
      {
        \"IndexName\": \"GameTitleIndex\",
        \"KeySchema\": [
          {\"AttributeName\": \"GameTitle\", \"KeyType\": \"HASH\"},
          {\"AttributeName\": \"TopScore\", \"KeyType\": \"RANGE\"}
        ],
        \"Projection\": {
          \"ProjectionType\": \"INCLUDE\",
          \"NonKeyAttributes\": [\"UserId\"]
        },
        \"ProvisionedThroughput\": {
          \"ReadCapacityUnits\": 10,
          \"WriteCapacityUnits\": 5
        }
      }
    ]"
```

Salida:

```
{
```

```
"TableDescription": {
  "AttributeDefinitions": [
    {
      "AttributeName": "GameTitle",
      "AttributeType": "S"
    },
    {
      "AttributeName": "TopScore",
      "AttributeType": "N"
    },
    {
      "AttributeName": "UserId",
      "AttributeType": "S"
    }
  ],
  "TableName": "GameScores",
  "KeySchema": [
    {
      "AttributeName": "UserId",
      "KeyType": "HASH"
    },
    {
      "AttributeName": "GameTitle",
      "KeyType": "RANGE"
    }
  ],
  "TableStatus": "CREATING",
  "CreationDateTime": "2020-05-26T17:28:15.602000-07:00",
  "ProvisionedThroughput": {
    "NumberOfDecreasesToday": 0,
    "ReadCapacityUnits": 10,
    "WriteCapacityUnits": 5
  },
  "TableSizeBytes": 0,
  "ItemCount": 0,
  "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
  "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "GlobalSecondaryIndexes": [
    {
      "IndexName": "GameTitleIndex",
      "KeySchema": [
        {
          "AttributeName": "GameTitle",
          "KeyType": "HASH"
        }
      ]
    }
  ]
}
```

```

        },
        {
            "AttributeName": "TopScore",
            "KeyType": "RANGE"
        }
    ],
    "Projection": {
        "ProjectionType": "INCLUDE",
        "NonKeyAttributes": [
            "UserId"
        ]
    },
    "IndexStatus": "CREATING",
    "ProvisionedThroughput": {
        "NumberOfDecreasesToday": 0,
        "ReadCapacityUnits": 10,
        "WriteCapacityUnits": 5
    },
    "IndexSizeBytes": 0,
    "ItemCount": 0,
    "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/index/GameTitleIndex"
    }
]
}
}

```

Para obtener más información, consulte [Operaciones básicas con tablas](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 6: Creación de una tabla con varios índices secundarios globales a la vez

En el siguiente ejemplo, se crea una tabla denominada GameScores con dos índices secundarios globales. Los GSI esquemas se transmiten a través de un archivo y no a través de la línea de comandos.

```

aws dynamodb create-table \
  --table-name GameScores \
  --attribute-
definitions AttributeName=UserId,AttributeType=S AttributeName=GameTitle,AttributeType=S Att
  \
  --key-
schema AttributeName=UserId,KeyType=HASH AttributeName=GameTitle,KeyType=RANGE \

```

```
--provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \  
--global-secondary-indexes file://gsi.json
```

Contenidos de `gsi.json`:

```
[  
  {  
    "IndexName": "GameTitleIndex",  
    "KeySchema": [  
      {  
        "AttributeName": "GameTitle",  
        "KeyType": "HASH"  
      },  
      {  
        "AttributeName": "TopScore",  
        "KeyType": "RANGE"  
      }  
    ],  
    "Projection": {  
      "ProjectionType": "ALL"  
    },  
    "ProvisionedThroughput": {  
      "ReadCapacityUnits": 10,  
      "WriteCapacityUnits": 5  
    }  
  },  
  {  
    "IndexName": "GameDataIndex",  
    "KeySchema": [  
      {  
        "AttributeName": "GameTitle",  
        "KeyType": "HASH"  
      },  
      {  
        "AttributeName": "Date",  
        "KeyType": "RANGE"  
      }  
    ],  
    "Projection": {  
      "ProjectionType": "ALL"  
    },  
    "ProvisionedThroughput": {  
      "ReadCapacityUnits": 5,  

```

```

    "WriteCapacityUnits": 5
  }
}
]

```

**Salida:**

```

{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "Date",
        "AttributeType": "S"
      },
      {
        "AttributeName": "GameTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "TopScore",
        "AttributeType": "N"
      },
      {
        "AttributeName": "UserId",
        "AttributeType": "S"
      }
    ],
    "TableName": "GameScores",
    "KeySchema": [
      {
        "AttributeName": "UserId",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "GameTitle",
        "KeyType": "RANGE"
      }
    ],
    "TableStatus": "CREATING",
    "CreationDateTime": "2020-08-04T16:40:55.524000-07:00",
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 10,

```

```
    "WriteCapacityUnits": 5
  },
  "TableSizeBytes": 0,
  "ItemCount": 0,
  "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
  "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "GlobalSecondaryIndexes": [
    {
      "IndexName": "GameTitleIndex",
      "KeySchema": [
        {
          "AttributeName": "GameTitle",
          "KeyType": "HASH"
        },
        {
          "AttributeName": "TopScore",
          "KeyType": "RANGE"
        }
      ],
      "Projection": {
        "ProjectionType": "ALL"
      },
      "IndexStatus": "CREATING",
      "ProvisionedThroughput": {
        "NumberOfDecreasesToday": 0,
        "ReadCapacityUnits": 10,
        "WriteCapacityUnits": 5
      },
      "IndexSizeBytes": 0,
      "ItemCount": 0,
      "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/index/GameTitleIndex"
    },
    {
      "IndexName": "GameDateIndex",
      "KeySchema": [
        {
          "AttributeName": "GameTitle",
          "KeyType": "HASH"
        },
        {
          "AttributeName": "Date",
          "KeyType": "RANGE"
        }
      ]
    }
  ]
}
```



```

    ],
    "Projection": {
      "ProjectionType": "ALL"
    },
    "IndexStatus": "CREATING",
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 5,
      "WriteCapacityUnits": 5
    },
    "IndexSizeBytes": 0,
    "ItemCount": 0,
    "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/index/GameDateIndex"
  }
]
}
}

```

Para obtener más información, consulte [Operaciones básicas con tablas](#) en la Guía para desarrolladores de Amazon DynamoDB.

#### Ejemplo 7: Creación de una tabla que tiene habilitado Streams

En el siguiente ejemplo, se crea una tabla denominada GameScores con DynamoDB Streams habilitado. En el flujo se escribirán tanto las imágenes nuevas como las antiguas de cada elemento.

```

aws dynamodb create-table \
  --table-name GameScores \
  --attribute-
definitions AttributeName=UserId,AttributeType=S AttributeName=GameTitle,AttributeType=S
  \
  --key-
schema AttributeName=UserId,KeyType=HASH AttributeName=GameTitle,KeyType=RANGE \
  --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
  --stream-specification StreamEnabled=TRUE,StreamViewType=NEW_AND_OLD_IMAGES

```

Salida:

```

{
  "TableDescription": {
    "AttributeDefinitions": [

```

```

    {
      "AttributeName": "GameTitle",
      "AttributeType": "S"
    },
    {
      "AttributeName": "UserId",
      "AttributeType": "S"
    }
  ],
  "TableName": "GameScores",
  "KeySchema": [
    {
      "AttributeName": "UserId",
      "KeyType": "HASH"
    },
    {
      "AttributeName": "GameTitle",
      "KeyType": "RANGE"
    }
  ],
  "TableStatus": "CREATING",
  "CreationDateTime": "2020-05-27T10:49:34.056000-07:00",
  "ProvisionedThroughput": {
    "NumberOfDecreasesToday": 0,
    "ReadCapacityUnits": 10,
    "WriteCapacityUnits": 5
  },
  "TableSizeBytes": 0,
  "ItemCount": 0,
  "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
  "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "StreamSpecification": {
    "StreamEnabled": true,
    "StreamViewType": "NEW_AND_OLD_IMAGES"
  },
  "LatestStreamLabel": "2020-05-27T17:49:34.056",
  "LatestStreamArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/stream/2020-05-27T17:49:34.056"
}
}

```

Para obtener más información, consulte [Operaciones básicas con tablas](#) en la Guía para desarrolladores de Amazon DynamoDB.

## Ejemplo 8: Creación de una tabla con un flujo habilitado solo de claves

En el siguiente ejemplo, se crea una tabla denominada GameScores con DynamoDB Streams habilitado. Solo se escriben en el flujo los atributos de clave del elementos modificados.

```
aws dynamodb create-table \
  --table-name GameScores \
  --attribute-
definitions AttributeName=UserId,AttributeType=S AttributeName=GameTitle,AttributeType=S
\
  --key-
schema AttributeName=UserId,KeyType=HASH AttributeName=GameTitle,KeyType=RANGE \
  --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
  --stream-specification StreamEnabled=TRUE,StreamViewType=KEYS_ONLY
```

Salida:

```
{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "GameTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "UserId",
        "AttributeType": "S"
      }
    ],
    "TableName": "GameScores",
    "KeySchema": [
      {
        "AttributeName": "UserId",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "GameTitle",
        "KeyType": "RANGE"
      }
    ],
    "TableStatus": "CREATING",
    "CreationDateTime": "2023-05-25T18:45:34.140000+00:00",
    "ProvisionedThroughput": {
```

```

        "NumberOfDecreasesToday": 0,
        "ReadCapacityUnits": 10,
        "WriteCapacityUnits": 5
    },
    "TableSizeBytes": 0,
    "ItemCount": 0,
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
    "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "StreamSpecification": {
        "StreamEnabled": true,
        "StreamViewType": "KEYS_ONLY"
    },
    "LatestStreamLabel": "2023-05-25T18:45:34.140",
    "LatestStreamArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/stream/2023-05-25T18:45:34.140",
    "DeletionProtectionEnabled": false
}
}

```

Para obtener más información, consulte [Captura de datos de cambios para DynamoDB Streams](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 9: Creación de una tabla mediante la clase de tabla de acceso poco frecuente estándar de DynamoDB

En el siguiente ejemplo se crea una tabla denominada GameScores y asigna la clase de tabla Estándar - Acceso poco frecuente (DynamoDB Standard-IA). Esta clase de tabla está optimizada para que el almacenamiento sea el costo dominante.

```

aws dynamodb create-table \
  --table-name GameScores \
  --attribute-
definitions AttributeName=UserId,AttributeType=S AttributeName=GameTitle,AttributeType=S
\
  --key-
schema AttributeName=UserId,KeyType=HASH AttributeName=GameTitle,KeyType=RANGE \
  --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
  --table-class STANDARD_INFREQUENT_ACCESS

```

Salida:

```
{
```

```
"TableDescription": {
  "AttributeDefinitions": [
    {
      "AttributeName": "GameTitle",
      "AttributeType": "S"
    },
    {
      "AttributeName": "UserId",
      "AttributeType": "S"
    }
  ],
  "TableName": "GameScores",
  "KeySchema": [
    {
      "AttributeName": "UserId",
      "KeyType": "HASH"
    },
    {
      "AttributeName": "GameTitle",
      "KeyType": "RANGE"
    }
  ],
  "TableStatus": "CREATING",
  "CreationDateTime": "2023-05-25T18:33:07.581000+00:00",
  "ProvisionedThroughput": {
    "NumberOfDecreasesToday": 0,
    "ReadCapacityUnits": 10,
    "WriteCapacityUnits": 5
  },
  "TableSizeBytes": 0,
  "ItemCount": 0,
  "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
  "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "TableClassSummary": {
    "TableClass": "STANDARD_INFREQUENT_ACCESS"
  },
  "DeletionProtectionEnabled": false
}
```

Para obtener más información, consulte [Clases de tabla](#) en la Guía para desarrolladores de Amazon DynamoDB.

## Ejemplo 10: Creación de una tabla con la protección contra eliminación habilitada

En el siguiente ejemplo, se crea una tabla denominada GameScores y habilita la protección contra eliminación.

```
aws dynamodb create-table \  
  --table-name GameScores \  
  --attribute-  
definitions AttributeName=UserId,AttributeType=S AttributeName=GameTitle,AttributeType=S \  
  \  
  --key-  
schema AttributeName=UserId,KeyType=HASH AttributeName=GameTitle,KeyType=RANGE \  
  --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \  
  --deletion-protection-enabled
```

Salida:

```
{  
  "TableDescription": {  
    "AttributeDefinitions": [  
      {  
        "AttributeName": "GameTitle",  
        "AttributeType": "S"  
      },  
      {  
        "AttributeName": "UserId",  
        "AttributeType": "S"  
      }  
    ],  
    "TableName": "GameScores",  
    "KeySchema": [  
      {  
        "AttributeName": "UserId",  
        "KeyType": "HASH"  
      },  
      {  
        "AttributeName": "GameTitle",  
        "KeyType": "RANGE"  
      }  
    ],  
    "TableStatus": "CREATING",  
    "CreationDateTime": "2023-05-25T23:02:17.093000+00:00",  
    "ProvisionedThroughput": {
```

```

        "NumberOfDecreasesToday": 0,
        "ReadCapacityUnits": 10,
        "WriteCapacityUnits": 5
    },
    "TableSizeBytes": 0,
    "ItemCount": 0,
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
    "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "DeletionProtectionEnabled": true
}
}

```

Para obtener más información, consulte [Uso de la protección contra eliminación](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [CreateTable](#) la Referencia de AWS CLI comandos.

## delete-backup

En el siguiente ejemplo de código, se muestra cómo usar delete-backup.

### AWS CLI

Para eliminar una copia de seguridad de DynamoDB existente

En el siguiente delete-backup ejemplo, se elimina la copia de seguridad existente especificada.

```

aws dynamodb delete-backup \
  --backup-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection/
backup/01576616366715-b4e58d3a

```

Salida:

```

{
  "BackupDescription": {
    "BackupDetails": {
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01576616366715-b4e58d3a",
      "BackupName": "MusicCollectionBackup",
      "BackupSizeBytes": 0,
      "BackupStatus": "DELETED",
      "BackupType": "USER",

```

```

        "BackupCreationDateTime": 1576616366.715
    },
    "SourceTableDetails": {
        "TableName": "MusicCollection",
        "TableId": "b0c04bcc-309b-4352-b2ae-9088af169fe2",
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
        "TableSizeBytes": 0,
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "SongTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableCreationDateTime": 1576615228.571,
        "ProvisionedThroughput": {
            "ReadCapacityUnits": 5,
            "WriteCapacityUnits": 5
        },
        "ItemCount": 0,
        "BillingMode": "PROVISIONED"
    },
    "SourceTableFeatureDetails": {}
}
}

```

Para obtener más información, consulte [Backup and Restore on Demand para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [DeleteBackup](#).AWS CLI

## delete-item

En el siguiente ejemplo de código, se muestra cómo usar `delete-item`.

### AWS CLI

#### Ejemplo 1: Eliminación de un elemento



En el siguiente ejemplo `delete-item`, se elimina un elemento de la tabla `MusicCollection` y se solicitan detalles sobre el elemento que se ha eliminado y la capacidad utilizada por la solicitud.

```
aws dynamodb delete-item \  
  --table-name MusicCollection \  
  --key file://key.json \  
  --return-values ALL_OLD \  
  --return-consumed-capacity TOTAL \  
  --return-item-collection-metrics SIZE
```

Contenidos de `key.json`:

```
{  
  "Artist": {"S": "No One You Know"},  
  "SongTitle": {"S": "Scared of My Shadow"}  
}
```

Salida:

```
{  
  "Attributes": {  
    "AlbumTitle": {  
      "S": "Blue Sky Blues"  
    },  
    "Artist": {  
      "S": "No One You Know"  
    },  
    "SongTitle": {  
      "S": "Scared of My Shadow"  
    }  
  },  
  "ConsumedCapacity": {  
    "TableName": "MusicCollection",  
    "CapacityUnits": 2.0  
  },  
  "ItemCollectionMetrics": {  
    "ItemCollectionKey": {  
      "Artist": {  
        "S": "No One You Know"  
      }  
    }  
  },  
}
```

```

        "SizeEstimateRangeGB": [
            0.0,
            1.0
        ]
    }
}

```

Para obtener más información, consulte [Escritura de un elemento](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 2: Eliminación de un elemento de forma condicional

En el siguiente ejemplo, se elimina un elemento de la tabla ProductCatalog solo si ProductCategory es Sporting Goods o Gardening Supplies y su precio está comprendido entre 500 y 600. Devuelve detalles sobre el elemento que se ha eliminado.

```

aws dynamodb delete-item \
  --table-name ProductCatalog \
  --key '{"Id":{"N":"456"}}' \
  --condition-expression "(ProductCategory IN (:cat1, :cat2)) and (#P between :lo and :hi)" \
  --expression-attribute-names file://names.json \
  --expression-attribute-values file://values.json \
  --return-values ALL_OLD

```

Contenidos de names.json:

```

{
  "#P": "Price"
}

```

Contenidos de values.json:

```

{
  ":cat1": {"S": "Sporting Goods"},
  ":cat2": {"S": "Gardening Supplies"},
  ":lo": {"N": "500"},
  ":hi": {"N": "600"}
}

```

Salida:

```
{
  "Attributes": {
    "Id": {
      "N": "456"
    },
    "Price": {
      "N": "550"
    },
    "ProductCategory": {
      "S": "Sporting Goods"
    }
  }
}
```

Para obtener más información, consulte [Escritura de un elemento](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [DeleteItem](#) la Referencia de AWS CLI comandos.

## delete-table

En el siguiente ejemplo de código, se muestra cómo usar delete-table.

### AWS CLI

Eliminación de una tabla

En el siguiente ejemplo de delete-table se elimina la tabla MusicCollection.

```
aws dynamodb delete-table \
  --table-name MusicCollection
```

Salida:

```
{
  "TableDescription": {
    "TableStatus": "DELETING",
    "TableSizeBytes": 0,
    "ItemCount": 0,
    "TableName": "MusicCollection",
    "ProvisionedThroughput": {
```

```

        "NumberOfDecreasesToday": 0,
        "WriteCapacityUnits": 5,
        "ReadCapacityUnits": 5
    }
}

```

Para obtener más información, consulte [Eliminación de una tabla](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [DeleteTable](#) la Referencia de AWS CLI comandos.

## describe-backup

En el siguiente ejemplo de código, se muestra cómo usar `describe-backup`.

### AWS CLI

Para obtener información sobre una copia de seguridad existente de una tabla

En el siguiente `describe-backup` ejemplo, se muestra información sobre la copia de seguridad existente especificada.

```

aws dynamodb describe-backup \
  --backup-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection/
  backup/01576616366715-b4e58d3a

```

Salida:

```

{
  "BackupDescription": {
    "BackupDetails": {
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01576616366715-b4e58d3a",
      "BackupName": "MusicCollectionBackup",
      "BackupSizeBytes": 0,
      "BackupStatus": "AVAILABLE",
      "BackupType": "USER",
      "BackupCreationDateTime": 1576616366.715
    },
    "SourceTableDetails": {

```

```
    "TableName": "MusicCollection",
    "TableId": "b0c04bcc-309b-4352-b2ae-9088af169fe2",
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
    "TableSizeBytes": 0,
    "KeySchema": [
      {
        "AttributeName": "Artist",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "SongTitle",
        "KeyType": "RANGE"
      }
    ],
    "TableCreationDateTime": 1576615228.571,
    "ProvisionedThroughput": {
      "ReadCapacityUnits": 5,
      "WriteCapacityUnits": 5
    },
    "ItemCount": 0,
    "BillingMode": "PROVISIONED"
  },
  "SourceTableFeatureDetails": {}
}
}
```

Para obtener más información, consulte [Backup and Restore on Demand para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [DescribeBackup](#).AWS CLI

## **describe-continuous-backups**

En el siguiente ejemplo de código, se muestra cómo usar `describe-continuous-backups`.

### AWS CLI

Para obtener información sobre las copias de seguridad continuas de una tabla de DynamoDB

En el siguiente `describe-continuous-backups` ejemplo, se muestran detalles sobre la configuración de copia de seguridad continua de la `MusicCollection` tabla.

```
aws dynamodb describe-continuous-backups \  
  --table-name MusicCollection
```

Salida:

```
{  
  "ContinuousBackupsDescription": {  
    "ContinuousBackupsStatus": "ENABLED",  
    "PointInTimeRecoveryDescription": {  
      "PointInTimeRecoveryStatus": "DISABLED"  
    }  
  }  
}
```

Para obtener más información, consulte [Point-in-Time Recovery for DynamoDB en la Guía para desarrolladores de Amazon DynamoDB](#).

- Para obtener API más información, consulte la Referencia de comandos. [DescribeContinuousBackups](#) AWS CLI

## describe-contributor-insights

En el siguiente ejemplo de código, se muestra cómo usar `describe-contributor-insights`.

AWS CLI

Para ver la configuración de Contributor Insights de una tabla de DynamoDB

En el siguiente `describe-contributor-insights` ejemplo, se muestra la configuración de Contributor Insights para la `MusicCollection` tabla y el índice secundario `AlbumTitle-index` global.

```
aws dynamodb describe-contributor-insights \  
  --table-name MusicCollection \  
  --index-name AlbumTitle-index
```

Salida:

```
{
```

```
"TableName": "MusicCollection",
"IndexName": "AlbumTitle-index",
"ContributorInsightsRuleList": [
  "DynamoDBContributorInsights-PKC-MusicCollection-1576629651520",
  "DynamoDBContributorInsights-SKC-MusicCollection-1576629651520",
  "DynamoDBContributorInsights-PKT-MusicCollection-1576629651520",
  "DynamoDBContributorInsights-SKT-MusicCollection-1576629651520"
],
"ContributorInsightsStatus": "ENABLED",
"LastUpdateDateTime": 1576629654.78
}
```

Para obtener más información, consulte [Análisis del acceso a los datos mediante CloudWatch Contributor Insights for DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [DescribeContributorInsights](#).AWS CLI

## describe-endpoints

En el siguiente ejemplo de código, se muestra cómo usar describe-endpoints.

AWS CLI

Para ver la información de los puntos finales regionales

En el siguiente describe-endpoints ejemplo, se muestran detalles sobre los puntos finales de la AWS región actual.

```
aws dynamodb describe-endpoints
```

Salida:

```
{
  "Endpoints": [
    {
      "Address": "dynamodb.us-west-2.amazonaws.com",
      "CachePeriodInMinutes": 1440
    }
  ]
}
```

Para obtener más información, consulte los [puntos de conexión y las cuotas de Amazon DynamoDB](#) en la referencia general.AWS

- Para API obtener más información, consulte la Referencia de [DescribeEndpoints](#) comandos AWS CLI .

## describe-global-table-settings

En el siguiente ejemplo de código, se muestra cómo usar describe-global-table-settings.

### AWS CLI

Para obtener información sobre la configuración de una tabla global de DynamoDB

En el siguiente describe-global-table-settings ejemplo, se muestra la configuración de la tabla MusicCollection global.

```
aws dynamodb describe-global-table-settings \  
  --global-table-name MusicCollection
```

Salida:

```
{  
  "GlobalTableName": "MusicCollection",  
  "ReplicaSettings": [  
    {  
      "RegionName": "us-east-1",  
      "ReplicaStatus": "ACTIVE",  
      "ReplicaProvisionedReadCapacityUnits": 10,  
      "ReplicaProvisionedReadCapacityAutoScalingSettings": {  
        "AutoScalingDisabled": true  
      },  
      "ReplicaProvisionedWriteCapacityUnits": 5,  
      "ReplicaProvisionedWriteCapacityAutoScalingSettings": {  
        "AutoScalingDisabled": true  
      }  
    },  
    {  
      "RegionName": "us-east-2",  
      "ReplicaStatus": "ACTIVE",  
      "ReplicaProvisionedReadCapacityUnits": 10,  
      "ReplicaProvisionedReadCapacityAutoScalingSettings": {
```



```

        "AutoScalingDisabled": true
    },
    "ReplicaProvisionedWriteCapacityUnits": 5,
    "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
        "AutoScalingDisabled": true
    }
}
]
}

```

Para obtener más información, consulte las tablas [globales de DynamoDB en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [DescribeGlobalTableSettings](#).AWS CLI

## describe-global-table

En el siguiente ejemplo de código, se muestra cómo usar `describe-global-table`.

### AWS CLI

Para mostrar información sobre una tabla global de DynamoDB

En el siguiente `describe-global-table` ejemplo, se muestran detalles sobre la tabla `MusicCollection` global.

```
aws dynamodb describe-global-table \
  --global-table-name MusicCollection
```

Salida:

```

{
  "GlobalTableDescription": {
    "ReplicationGroup": [
      {
        "RegionName": "us-east-2"
      },
      {
        "RegionName": "us-east-1"
      }
    ],

```

```
    "GlobalTableArn": "arn:aws:dynamodb::123456789012:global-table/  
MusicCollection",  
    "CreationDateTime": 1576625818.532,  
    "GlobalTableStatus": "ACTIVE",  
    "GlobalTableName": "MusicCollection"  
  }  
}
```

Para obtener más información, consulte las tablas [globales de DynamoDB en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [DescribeGlobalTable](#).AWS CLI

## describe-limits

En el siguiente ejemplo de código, se muestra cómo usar `describe-limits`.

AWS CLI

Para ver los límites de la capacidad aprovisionada

En el siguiente `describe-limits` ejemplo, se muestran los límites de capacidad aprovisionada de su cuenta en la región actual. AWS

```
aws dynamodb describe-limits
```

Salida:

```
{  
  "AccountMaxReadCapacityUnits": 80000,  
  "AccountMaxWriteCapacityUnits": 80000,  
  "TableMaxReadCapacityUnits": 40000,  
  "TableMaxWriteCapacityUnits": 40000  
}
```

Para obtener más información, consulte [Límites de DynamoDB en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [DescribeLimits](#).AWS CLI

## describe-table-replica-auto-scaling

En el siguiente ejemplo de código, se muestra cómo usar `describe-table-replica-auto-scaling`.

### AWS CLI

Para ver la configuración de escalado automático en las réplicas de una tabla global

El siguiente `describe-table-replica-auto-scaling` ejemplo muestra la configuración de escalado automático en todas las réplicas de la tabla `MusicCollection` global.

```
aws dynamodb describe-table-replica-auto-scaling \  
--table-name MusicCollection
```

Salida:

```
{  
  "TableAutoScalingDescription": {  
    "TableName": "MusicCollection",  
    "TableStatus": "ACTIVE",  
    "Replicas": [  
      {  
        "RegionName": "us-east-1",  
        "GlobalSecondaryIndexes": [],  
        "ReplicaProvisionedReadCapacityAutoScalingSettings": {  
          "MinimumUnits": 5,  
          "MaximumUnits": 40000,  
          "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/  
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/  
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",  
          "ScalingPolicies": [  
            {  
              "PolicyName": "DynamoDBReadCapacityUtilization:table/  
MusicCollection",  
              "TargetTrackingScalingPolicyConfiguration": {  
                "TargetValue": 70.0  
              }  
            }  
          ]  
        },  
        "ReplicaProvisionedWriteCapacityAutoScalingSettings": {  
          "MinimumUnits": 5,  
          "MaximumUnits": 40000,  
          "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/  
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/  
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",  
          "ScalingPolicies": [  
            {  
              "PolicyName": "DynamoDBWriteCapacityUtilization:table/  
MusicCollection",  
              "TargetTrackingScalingPolicyConfiguration": {  
                "TargetValue": 70.0  
              }  
            }  
          ]  
        }  
      }  
    ]  
  }  
}
```

```
        "MaximumUnits": 40000,
        "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
        "ScalingPolicies": [
            {
                "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
                "TargetTrackingScalingPolicyConfiguration": {
                    "TargetValue": 70.0
                }
            }
        ]
    },
    "ReplicaStatus": "ACTIVE"
},
{
    "RegionName": "us-east-2",
    "GlobalSecondaryIndexes": [],
    "ReplicaProvisionedReadCapacityAutoScalingSettings": {
        "MinimumUnits": 5,
        "MaximumUnits": 40000,
        "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
        "ScalingPolicies": [
            {
                "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
                "TargetTrackingScalingPolicyConfiguration": {
                    "TargetValue": 70.0
                }
            }
        ]
    },
    "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
        "MinimumUnits": 5,
        "MaximumUnits": 40000,
        "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
        "ScalingPolicies": [
            {
```

```

    "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
    "TargetTrackingScalingPolicyConfiguration": {
      "TargetValue": 70.0
    }
  ],
  "ReplicaStatus": "ACTIVE"
}
]
}
}

```

Para obtener más información, consulte las tablas [globales de DynamoDB en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [DescribeTableReplicaAutoScaling](#).AWS CLI

## describe-table

En el siguiente ejemplo de código, se muestra cómo usar `describe-table`.

### AWS CLI

Descripción de una tabla

En el siguiente ejemplo `describe-table`, se describe la tabla `MusicCollection`.

```
aws dynamodb describe-table \
  --table-name MusicCollection
```

Salida:

```

{
  "Table": {
    "AttributeDefinitions": [
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {

```

```

        "AttributeName": "SongTitle",
        "AttributeType": "S"
    }
],
"ProvisionedThroughput": {
    "NumberOfDecreasesToday": 0,
    "WriteCapacityUnits": 5,
    "ReadCapacityUnits": 5
},
"TableSizeBytes": 0,
"TableName": "MusicCollection",
"TableStatus": "ACTIVE",
"KeySchema": [
    {
        "KeyType": "HASH",
        "AttributeName": "Artist"
    },
    {
        "KeyType": "RANGE",
        "AttributeName": "SongTitle"
    }
],
"ItemCount": 0,
"CreationDateTime": 1421866952.062
}
}

```

Para obtener más información, consulte [Descripción de una tabla](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [DescribeTable](#) la Referencia de AWS CLI comandos.

## **describe-time-to-live**

En el siguiente ejemplo de código, se muestra cómo usar `describe-time-to-live`.

### AWS CLI

Visualización de la configuración de tiempo de vida de una tabla

En el siguiente ejemplo de `describe-time-to-live`, se muestra la configuración de tiempo de vida de la tabla `MusicCollection`.

```
aws dynamodb describe-time-to-live \  
  --table-name MusicCollection
```

Salida:

```
{  
  "TimeToLiveDescription": {  
    "TimeToLiveStatus": "ENABLED",  
    "AttributeName": "ttl"  
  }  
}
```

Para obtener más información, consulte [Tiempo de vida](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [DescribeTimeToLive](#) la Referencia de AWS CLI comandos.

## get-item

En el siguiente ejemplo de código, se muestra cómo usar `get-item`.

### AWS CLI

Ejemplo 1: Lectura de un elemento de una tabla

En el siguiente ejemplo `get-item`, se recupera un elemento de la tabla `MusicCollection`. La tabla tiene una clave hash-and-range principal (`ArtistySongTitle`), por lo que debe especificar estos dos atributos. El comando también solicita información sobre la capacidad de lectura consumida por la operación.

```
aws dynamodb get-item \  
  --table-name MusicCollection \  
  --key file://key.json \  
  --return-consumed-capacity TOTAL
```

Contenidos de `key.json`:

```
{  
  "Artist": {"S": "Acme Band"},
```

```
"SongTitle": {"S": "Happy Day"}
}
```

Salida:

```
{
  "Item": {
    "AlbumTitle": {
      "S": "Songs About Life"
    },
    "SongTitle": {
      "S": "Happy Day"
    },
    "Artist": {
      "S": "Acme Band"
    }
  },
  "ConsumedCapacity": {
    "TableName": "MusicCollection",
    "CapacityUnits": 0.5
  }
}
```

Para obtener más información, consulte [Lectura de un elemento](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 2: Lectura de un elemento mediante una lectura coherente

En el siguiente ejemplo, se recupera un elemento de la tabla `MusicCollection` con lecturas altamente coherentes.

```
aws dynamodb get-item \
  --table-name MusicCollection \
  --key file://key.json \
  --consistent-read \
  --return-consumed-capacity TOTAL
```

Contenidos de `key.json`:

```
{
  "Artist": {"S": "Acme Band"},
  "SongTitle": {"S": "Happy Day"}
```



```
}

```

Salida:

```
{
  "Item": {
    "AlbumTitle": {
      "S": "Songs About Life"
    },
    "SongTitle": {
      "S": "Happy Day"
    },
    "Artist": {
      "S": "Acme Band"
    }
  },
  "ConsumedCapacity": {
    "TableName": "MusicCollection",
    "CapacityUnits": 1.0
  }
}
```

Para obtener más información, consulte [Lectura de un elemento](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 3: Recuperación de atributos específicos de un elemento

En el siguiente ejemplo, se utiliza una expresión de proyección para recuperar solo tres atributos del elemento deseado.

```
aws dynamodb get-item \
  --table-name ProductCatalog \
  --key '{"Id": {"N": "102"}}' \
  --projection-expression "#T, #C, #P" \
  --expression-attribute-names file://names.json
```

Contenidos de `names.json`:

```
{
  "#T": "Title",
  "#C": "ProductCategory",
  "#P": "Price"
}
```

```
}
```

Salida:

```
{
  "Item": {
    "Price": {
      "N": "20"
    },
    "Title": {
      "S": "Book 102 Title"
    },
    "ProductCategory": {
      "S": "Book"
    }
  }
}
```

Para obtener más información, consulte [Lectura de un elemento](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [GetItem](#) la Referencia de AWS CLI comandos.

## list-backups

En el siguiente ejemplo de código, se muestra cómo usar `list-backups`.

AWS CLI

Ejemplo 1: Para enumerar todas las copias de seguridad de DynamoDB existentes

En el siguiente `list-backups` ejemplo, se enumeran todas las copias de seguridad existentes.

```
aws dynamodb list-backups
```

Salida:

```
{
  "BackupSummaries": [
    {
      "TableName": "MusicCollection",
```

```

        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
        "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-a1bcd234",
        "BackupName": "MusicCollectionBackup1",
        "BackupCreationDateTime": "2020-02-12T14:41:51.617000-08:00",
        "BackupStatus": "AVAILABLE",
        "BackupType": "USER",
        "BackupSizeBytes": 170
    },
    {
        "TableName": "MusicCollection",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
        "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-b2abc345",
        "BackupName": "MusicCollectionBackup2",
        "BackupCreationDateTime": "2020-06-26T11:08:35.431000-07:00",
        "BackupStatus": "AVAILABLE",
        "BackupType": "USER",
        "BackupSizeBytes": 400
    }
]
}

```

Para obtener más información, consulte [Backup and Restore on Demand para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 2: Para enumerar las copias de seguridad creadas por los usuarios en un intervalo de tiempo específico

El siguiente ejemplo muestra solo las copias de seguridad de la `MusicCollection` tabla que creó el usuario (no las que DynamoDB creó automáticamente) con una fecha de creación entre el 1 de enero de 2020 y el 1 de marzo de 2020.

```

aws dynamodb list-backups \
  --table-name MusicCollection \
  --time-range-lower-bound 1577836800 \
  --time-range-upper-bound 1583020800 \
  --backup-type USER

```

Salida:

```
{
  "BackupSummaries": [
    {
      "TableName": "MusicCollection",
      "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-a1bcd234",
      "BackupName": "MusicCollectionBackup1",
      "BackupCreationDateTime": "2020-02-12T14:41:51.617000-08:00",
      "BackupStatus": "AVAILABLE",
      "BackupType": "USER",
      "BackupSizeBytes": 170
    }
  ]
}
```

Para obtener más información, consulte [Backup and Restore on Demand para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 3: Para limitar el tamaño de la página

El siguiente ejemplo devuelve una lista de todas las copias de seguridad existentes, pero recupera solo un elemento en cada llamada y, si es necesario, realiza varias llamadas para obtener la lista completa. Limitar el tamaño de página resulta útil cuando se ejecutan comandos de la lista en un gran número de recursos, lo que puede provocar un error de “tiempo de espera” cuando se utiliza el tamaño de página predeterminado de 1000.

```
aws dynamodb list-backups \
  --page-size 1
```

Salida:

```
{
  "BackupSummaries": [
    {
      "TableName": "MusicCollection",
      "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
```

```

        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
        "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-a1bcd234",
        "BackupName": "MusicCollectionBackup1",
        "BackupCreationDateTime": "2020-02-12T14:41:51.617000-08:00",
        "BackupStatus": "AVAILABLE",
        "BackupType": "USER",
        "BackupSizeBytes": 170
    },
    {
        "TableName": "MusicCollection",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
        "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-b2abc345",
        "BackupName": "MusicCollectionBackup2",
        "BackupCreationDateTime": "2020-06-26T11:08:35.431000-07:00",
        "BackupStatus": "AVAILABLE",
        "BackupType": "USER",
        "BackupSizeBytes": 400
    }
]
}

```

Para obtener más información, consulte [Backup and Restore on Demand para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 4: Para limitar el número de artículos devueltos

El siguiente ejemplo limita el número de artículos devueltos a 1. La respuesta incluye un valor `NextToken` con el que recuperar la siguiente página de resultados.

```

aws dynamodb list-backups \
  --max-items 1

```

Salida:

```

{
  "BackupSummaries": [
    {

```

```

        "TableName": "MusicCollection",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
        "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-a1bcd234",
        "BackupName": "MusicCollectionBackup1",
        "BackupCreationDateTime": "2020-02-12T14:41:51.617000-08:00",
        "BackupStatus": "AVAILABLE",
        "BackupType": "USER",
        "BackupSizeBytes": 170
    }
  ],
  "NextToken":
  "abCDeFGhiJKlMnOPqrSTuvwxYZ1aBCdEFghijK7LM51nOpqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9"
}

```

Para obtener más información, consulte [Backup and Restore on Demand para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 5: Para recuperar la siguiente página de resultados

El comando siguiente utiliza el valor `NextToken` de una llamada anterior al comando `list-backups` para recuperar otra página de resultados. Puesto que la respuesta en este caso no incluye ningún valor `NextToken`, sabemos que hemos llegado al final de los resultados.

```

aws dynamodb list-backups \
  --starting-
  token abCDeFGhiJKlMnOPqrSTuvwxYZ1aBCdEFghijK7LM51nOpqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9

```

Salida

```

{
  "BackupSummaries": [
    {
      "TableName": "MusicCollection",
      "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
      "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-b2abc345",
      "BackupName": "MusicCollectionBackup2",

```

```
    "BackupCreationDateTime": "2020-06-26T11:08:35.431000-07:00",
    "BackupStatus": "AVAILABLE",
    "BackupType": "USER",
    "BackupSizeBytes": 400
  }
]
```

Para obtener más información, consulte [Backup and Restore on Demand para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [ListBackups](#).AWS CLI

## list-contributor-insights

En el siguiente ejemplo de código, se muestra cómo usar `list-contributor-insights`.

AWS CLI

Ejemplo 1: Para ver una lista de resúmenes de Contributor Insights

En el siguiente `list-contributor-insights` ejemplo, se muestra una lista de resúmenes de Contributor Insights.

```
aws dynamodb list-contributor-insights
```

Salida:

```
{
  "ContributorInsightsSummaries": [
    {
      "TableName": "MusicCollection",
      "IndexName": "AlbumTitle-index",
      "ContributorInsightsStatus": "ENABLED"
    },
    {
      "TableName": "ProductCatalog",
      "ContributorInsightsStatus": "ENABLED"
    },
    {
      "TableName": "Forum",
      "ContributorInsightsStatus": "ENABLED"
    }
  ]
}
```

```

    },
    {
      "TableName": "Reply",
      "ContributorInsightsStatus": "ENABLED"
    },
    {
      "TableName": "Thread",
      "ContributorInsightsStatus": "ENABLED"
    }
  ]
}

```

Para obtener más información, consulte [Análisis del acceso a los datos mediante CloudWatch Contributor Insights for DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 2: Para limitar el número de artículos devueltos

El siguiente ejemplo limita el número de artículos devueltos a 4. La respuesta incluye un valor `NextToken` con el que recuperar la siguiente página de resultados.

```

aws dynamodb list-contributor-insights \
  --max-results 4

```

Salida:

```

{
  "ContributorInsightsSummaries": [
    {
      "TableName": "MusicCollection",
      "IndexName": "AlbumTitle-index",
      "ContributorInsightsStatus": "ENABLED"
    },
    {
      "TableName": "ProductCatalog",
      "ContributorInsightsStatus": "ENABLED"
    },
    {
      "TableName": "Forum",
      "ContributorInsightsStatus": "ENABLED"
    }
  ],
  "NextToken":
  "abCDeFGhiJKlmnOPqrSTuvwXYZ1aBCdEFghijK7LM51n0pqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9"
}

```



```
}

```

Para obtener más información, consulte [Análisis del acceso a los datos mediante CloudWatch Contributor Insights for DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 3: Para recuperar la siguiente página de resultados

El comando siguiente utiliza el valor `NextToken` de una llamada anterior al comando `list-contributor-insights` para recuperar otra página de resultados. Puesto que la respuesta en este caso no incluye ningún valor `NextToken`, sabemos que hemos llegado al final de los resultados.

```
aws dynamodb list-contributor-insights \
  --max-results 4 \
  --next-
token abCDeFGhiJKLmnOPqrSTuvwxYZ1aBCdEFghijK7LM51n0ppqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9
```

Salida:

```
{
  "ContributorInsightsSummaries": [
    {
      "TableName": "Reply",
      "ContributorInsightsStatus": "ENABLED"
    },
    {
      "TableName": "Thread",
      "ContributorInsightsStatus": "ENABLED"
    }
  ]
}
```

Para obtener más información, consulte [Análisis del acceso a los datos mediante CloudWatch Contributor Insights for DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [ListContributorInsights](#).AWS CLI

## list-global-tables

En el siguiente ejemplo de código, se muestra cómo usar `list-global-tables`.

## AWS CLI

Para enumerar las tablas globales de DynamoDB existentes

En el siguiente `list-global-tables` ejemplo, se muestran todas las tablas globales existentes.

```
aws dynamodb list-global-tables
```

Salida:

```
{
  "GlobalTables": [
    {
      "GlobalTableName": "MusicCollection",
      "ReplicationGroup": [
        {
          "RegionName": "us-east-2"
        },
        {
          "RegionName": "us-east-1"
        }
      ]
    }
  ]
}
```

Para obtener más información, consulte las tablas [globales de DynamoDB en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [ListGlobalTables](#).AWS CLI

## list-tables

En el siguiente ejemplo de código, se muestra cómo usar `list-tables`.

## AWS CLI

Ejemplo 1: Creación de una lista de tablas

En el siguiente `list-tables` ejemplo, se enumeran todas las tablas asociadas a la AWS cuenta corriente y a la región.

```
aws dynamodb list-tables
```

Salida:

```
{
  "TableNames": [
    "Forum",
    "ProductCatalog",
    "Reply",
    "Thread"
  ]
}
```

Para obtener más información, consulte [Enumeración de nombres de tablas](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 2: Limitación del tamaño de la página

En el siguiente ejemplo, se devuelve una lista de todas las tablas existentes, pero recupera solo un elemento en cada llamada y, si es necesario, realiza varias llamadas para obtener la lista completa. Limitar el tamaño de página resulta útil cuando se ejecutan comandos de la lista en un gran número de recursos, lo que puede provocar un error de “tiempo de espera” cuando se utiliza el tamaño de página predeterminado de 1000.

```
aws dynamodb list-tables \
  --page-size 1
```

Salida:

```
{
  "TableNames": [
    "Forum",
    "ProductCatalog",
    "Reply",
    "Thread"
  ]
}
```

Para obtener más información, consulte [Enumeración de nombres de tablas](#) en la Guía para desarrolladores de Amazon DynamoDB.

### Ejemplo 3: Limitación del número de elementos devueltos

En el siguiente ejemplo, se limita el número de filas devueltas a 2. La respuesta incluye un valor `NextToken` con el que recuperar la siguiente página de resultados.

```
aws dynamodb list-tables \  
  --max-items 2
```

Salida:

```
{  
  "TableNames": [  
    "Forum",  
    "ProductCatalog"  
  ],  
  "NextToken":  
  "abCDeFGhiJKlMnOPqrSTuvwxYZ1aBCdEFghijK7LM51n0ppqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9"  
}
```

Para obtener más información, consulte [Enumeración de nombres de tablas](#) en la Guía para desarrolladores de Amazon DynamoDB.

### Ejemplo 4: Recuperación de la siguiente página de resultados

El comando siguiente utiliza el valor `NextToken` de una llamada anterior al comando `list-tables` para recuperar otra página de resultados. Puesto que la respuesta en este caso no incluye ningún valor `NextToken`, sabemos que hemos llegado al final de los resultados.

```
aws dynamodb list-tables \  
  --starting-  
  token abCDeFGhiJKlMnOPqrSTuvwxYZ1aBCdEFghijK7LM51n0ppqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9
```

Salida:

```
{  
  "TableNames": [  
    "Reply",  
    "Thread"  
  ]  
}
```

```
]
}
```

Para obtener más información, consulte [Enumeración de nombres de tablas](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [ListTables](#) la Referencia de AWS CLI comandos.

## list-tags-of-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-of-resource`.

### AWS CLI

Ejemplo 1: Para enumerar las etiquetas de un recurso de DynamoDB

En el siguiente `list-tags-of-resource` ejemplo, se muestran las etiquetas de la `MusicCollection` tabla.

```
aws dynamodb list-tags-of-resource \
  --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection
```

Salida:

```
{
  "Tags": [
    {
      "Key": "Owner",
      "Value": "blueTeam"
    },
    {
      "Key": "Environment",
      "Value": "Production"
    }
  ]
}
```

Para obtener más información, consulte [Etiquetado para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 2: limitar el número de etiquetas devueltas

El siguiente ejemplo limita el número de etiquetas devueltas a 1. La respuesta incluye un valor `NextToken` con el que recuperar la siguiente página de resultados.

```
aws dynamodb list-tags-of-resource \  
  --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \  
  --max-items 1
```

Salida:

```
{  
  "Tags": [  
    {  
      "Key": "Owner",  
      "Value": "blueTeam"  
    }  
  ],  
  "NextToken":  
  "abCDeFGhiJKlMnOPqrSTuvwxYZ1aBCdEFghijK7LM51n0pqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9"  
}
```

Para obtener más información, consulte [Etiquetado para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 3: Para recuperar la siguiente página de resultados

El comando siguiente utiliza el valor `NextToken` de una llamada anterior al comando `list-tags-of-resource` para recuperar otra página de resultados. Puesto que la respuesta en este caso no incluye ningún valor `NextToken`, sabemos que hemos llegado al final de los resultados.

```
aws dynamodb list-tags-of-resource \  
  --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \  
  --starting-  
  token abCDeFGhiJKlMnOPqrSTuvwxYZ1aBCdEFghijK7LM51n0pqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9
```

Salida:

```
{  
  "Tags": [  
    {  
      "Key": "Environment",  
      "Value": "Production"  
    }  
  ]  
}
```

```

    }
  ]
}

```

Para obtener más información, consulte [Etiquetado para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos. [ListTagsOfResource](#) AWS CLI

## put-item

En el siguiente ejemplo de código, se muestra cómo usar `put-item`.

### AWS CLI

Ejemplo 1: Adición de un elemento a una tabla

En el siguiente `put-item` ejemplo, se agrega un elemento nuevo a la `MusicCollection` tabla.

```

aws dynamodb put-item \
  --table-name MusicCollection \
  --item file://item.json \
  --return-consumed-capacity TOTAL \
  --return-item-collection-metrics SIZE

```

Contenidos de `item.json`:

```

{
  "Artist": {"S": "No One You Know"},
  "SongTitle": {"S": "Call Me Today"},
  "AlbumTitle": {"S": "Greatest Hits"}
}

```

Salida:

```

{
  "ConsumedCapacity": {
    "TableName": "MusicCollection",
    "CapacityUnits": 1.0
  },
  "ItemCollectionMetrics": {

```

```

    "ItemCollectionKey": {
      "Artist": {
        "S": "No One You Know"
      }
    },
    "SizeEstimateRangeGB": [
      0.0,
      1.0
    ]
  }
}

```

Para obtener más información, consulte [Escritura de un elemento](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 2: Sobrescritura condicional de un elemento de una tabla

En el siguiente ejemplo de `put-item` se sobrescribe un elemento existente de la tabla `MusicCollection` solo si ese elemento existente tiene un atributo `AlbumTitle` con un valor de `Greatest Hits`. El comando devuelve el valor anterior del elemento.

```

aws dynamodb put-item \
  --table-name MusicCollection \
  --item file://item.json \
  --condition-expression "#A = :A" \
  --expression-attribute-names file://names.json \
  --expression-attribute-values file://values.json \
  --return-values ALL_OLD

```

Contenidos de `item.json`:

```

{
  "Artist": {"S": "No One You Know"},
  "SongTitle": {"S": "Call Me Today"},
  "AlbumTitle": {"S": "Somewhat Famous"}
}

```

Contenidos de `names.json`:

```

{
  "#A": "AlbumTitle"
}

```



```
}
```

Contenidos de `values.json`:

```
{
  "A": {"S": "Greatest Hits"}
}
```

Salida:

```
{
  "Attributes": {
    "AlbumTitle": {
      "S": "Greatest Hits"
    },
    "Artist": {
      "S": "No One You Know"
    },
    "SongTitle": {
      "S": "Call Me Today"
    }
  }
}
```

Si la clave ya existe, debería ver el siguiente resultado:

```
A client error (ConditionalCheckFailedException) occurred when calling the PutItem
operation: The conditional request failed.
```

Para obtener más información, consulte [Escritura de un elemento](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [PutItem](#) la Referencia de AWS CLI comandos.

## query

En el siguiente ejemplo de código, se muestra cómo usar `query`.

### AWS CLI

#### Ejemplo 1: Consulta de una tabla

En el siguiente ejemplo de query se consultan elementos de la tabla `MusicCollection`. La tabla tiene una clave hash-and-range principal (`ArtistySongTitle`), pero esta consulta solo especifica el valor de la clave hash. Devuelve los títulos de las canciones del artista llamado “No One You Know”.

```
aws dynamodb query \  
  --table-name MusicCollection \  
  --projection-expression "SongTitle" \  
  --key-condition-expression "Artist = :v1" \  
  --expression-attribute-values file://expression-attributes.json \  
  --return-consumed-capacity TOTAL
```

Contenidos de `expression-attributes.json`:

```
{  
  ":v1": {"S": "No One You Know"}  
}
```

Salida:

```
{  
  "Items": [  
    {  
      "SongTitle": {  
        "S": "Call Me Today"  
      },  
      "SongTitle": {  
        "S": "Scared of My Shadow"  
      }  
    }  
  ],  
  "Count": 2,  
  "ScannedCount": 2,  
  "ConsumedCapacity": {  
    "TableName": "MusicCollection",  
    "CapacityUnits": 0.5  
  }  
}
```

Para obtener más información, consulte [Uso de consultas en DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 2: Consulta de una tabla con lecturas altamente coherentes y recorrer el índice en orden descendente

En el siguiente ejemplo se realiza la misma consulta que en el primer ejemplo, pero se devuelven los resultados en orden inverso y se utilizan lecturas altamente coherentes.

```
aws dynamodb query \  
  --table-name MusicCollection \  
  --projection-expression "SongTitle" \  
  --key-condition-expression "Artist = :v1" \  
  --expression-attribute-values file://expression-attributes.json \  
  --consistent-read \  
  --no-scan-index-forward \  
  --return-consumed-capacity TOTAL
```

Contenidos de `expression-attributes.json`:

```
{  
  ":v1": {"S": "No One You Know"}  
}
```

Salida:

```
{  
  "Items": [  
    {  
      "SongTitle": {  
        "S": "Scared of My Shadow"  
      }  
    },  
    {  
      "SongTitle": {  
        "S": "Call Me Today"  
      }  
    }  
  ],  
  "Count": 2,  
  "ScannedCount": 2,  
  "ConsumedCapacity": {  
    "TableName": "MusicCollection",  
    "CapacityUnits": 1.0  
  }  
}
```

```
}

```

Para obtener más información, consulte [Uso de consultas en DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

### Ejemplo 3: Filtrado de resultados específicos

En el siguiente ejemplo se consulta MusicCollection pero se excluyen los resultados con valores específicos en el atributo AlbumTitle. Tenga en cuenta que esto no afecta a ScannedCount o ConsumedCapacity, ya que el filtro se aplica después de leer los elementos.

```
aws dynamodb query \
  --table-name MusicCollection \
  --key-condition-expression "#n1 = :v1" \
  --filter-expression "NOT (#n2 IN (:v2, :v3))" \
  --expression-attribute-names file://names.json \
  --expression-attribute-values file://values.json \
  --return-consumed-capacity TOTAL
```

Contenidos de values.json:

```
{
  ":v1": {"S": "No One You Know"},
  ":v2": {"S": "Blue Sky Blues"},
  ":v3": {"S": "Greatest Hits"}
}
```

Contenidos de names.json:

```
{
  "#n1": "Artist",
  "#n2": "AlbumTitle"
}
```

Salida:

```
{
  "Items": [
    {
      "AlbumTitle": {
        "S": "Somewhat Famous"
      }
    }
  ]
}
```

```

    },
    "Artist": {
      "S": "No One You Know"
    },
    "SongTitle": {
      "S": "Call Me Today"
    }
  }
],
"Count": 1,
"ScannedCount": 2,
"ConsumedCapacity": {
  "TableName": "MusicCollection",
  "CapacityUnits": 0.5
}
}

```

Para obtener más información, consulte [Uso de consultas en DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

#### Ejemplo 4: Recuperación de un solo recuento de elementos

En el siguiente ejemplo, se recupera un recuento de los elementos que coinciden con la consulta, pero no recupera ninguno de los elementos en sí.

```

aws dynamodb query \
  --table-name MusicCollection \
  --select COUNT \
  --key-condition-expression "Artist = :v1" \
  --expression-attribute-values file://expression-attributes.json

```

Contenidos de `expression-attributes.json`:

```

{
  ":v1": {"S": "No One You Know"}
}

```

Salida:

```

{
  "Count": 2,
  "ScannedCount": 2,

```

```
"ConsumedCapacity": null
}
```

Para obtener más información, consulte [Uso de consultas en DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

#### Ejemplo 5: Consulta de un índice

El siguiente ejemplo consulta el índice secundario global AlbumTitleIndex. La consulta devuelve todos los atributos de la tabla base que se han proyectado en el índice secundario local. Tenga en cuenta que, al consultar un índice secundario local o un índice secundario global, también debe proporcionar el nombre de la tabla base mediante el parámetro `table-name`.

```
aws dynamodb query \
  --table-name MusicCollection \
  --index-name AlbumTitleIndex \
  --key-condition-expression "Artist = :v1" \
  --expression-attribute-values file://expression-attributes.json \
  --select ALL_PROJECTED_ATTRIBUTES \
  --return-consumed-capacity INDEXES
```

Contenidos de `expression-attributes.json`:

```
{
  ":v1": {"S": "No One You Know"}
}
```

Salida:

```
{
  "Items": [
    {
      "AlbumTitle": {
        "S": "Blue Sky Blues"
      },
      "Artist": {
        "S": "No One You Know"
      },
      "SongTitle": {
        "S": "Scared of My Shadow"
      }
    },
  ],
}
```

```
{
  "AlbumTitle": {
    "S": "Somewhat Famous"
  },
  "Artist": {
    "S": "No One You Know"
  },
  "SongTitle": {
    "S": "Call Me Today"
  }
},
"Count": 2,
"ScannedCount": 2,
"ConsumedCapacity": {
  "TableName": "MusicCollection",
  "CapacityUnits": 0.5,
  "Table": {
    "CapacityUnits": 0.0
  },
  "LocalSecondaryIndexes": {
    "AlbumTitleIndex": {
      "CapacityUnits": 0.5
    }
  }
}
}
```

Para obtener más información, consulte [Uso de consultas en DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [Query](#) in AWS CLI Command Reference.

## restore-table-from-backup

En el siguiente ejemplo de código, se muestra cómo usar `restore-table-from-backup`.

### AWS CLI

Para restaurar una tabla de DynamoDB a partir de una copia de seguridad existente

El siguiente `restore-table-from-backup` ejemplo restaura la tabla especificada a partir de una copia de seguridad existente.

```
aws dynamodb restore-table-from-backup \  
  --target-table-name MusicCollection \  
  --backup-arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection/  
backup/01576616366715-b4e58d3a
```

Salida:

```
{  
  "TableDescription": {  
    "AttributeDefinitions": [  
      {  
        "AttributeName": "Artist",  
        "AttributeType": "S"  
      },  
      {  
        "AttributeName": "SongTitle",  
        "AttributeType": "S"  
      }  
    ],  
    "TableName": "MusicCollection2",  
    "KeySchema": [  
      {  
        "AttributeName": "Artist",  
        "KeyType": "HASH"  
      },  
      {  
        "AttributeName": "SongTitle",  
        "KeyType": "RANGE"  
      }  
    ],  
    "TableStatus": "CREATING",  
    "CreationDateTime": 1576618274.326,  
    "ProvisionedThroughput": {  
      "NumberOfDecreasesToday": 0,  
      "ReadCapacityUnits": 5,  
      "WriteCapacityUnits": 5  
    },  
    "TableSizeBytes": 0,  
    "ItemCount": 0,  
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/  
MusicCollection2",  
    "TableId": "114865c9-5ef3-496c-b4d1-c4cbdd2d44fb",  
    "BillingModeSummary": {
```



```

        "BillingMode": "PROVISIONED"
    },
    "RestoreSummary": {
        "SourceBackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01576616366715-b4e58d3a",
        "SourceTableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
        "RestoreDateTime": 1576616366.715,
        "RestoreInProgress": true
    }
}
}
}

```

Para obtener más información, consulte [Backup and Restore on Demand para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [RestoreTableFromBackup](#).AWS CLI

## restore-table-to-point-in-time

En el siguiente ejemplo de código, se muestra cómo usar `restore-table-to-point-in-time`.

### AWS CLI

Para restaurar una tabla de DynamoDB a un punto en el tiempo

En el siguiente `restore-table-to-point-in-time` ejemplo, se restaura la `MusicCollection` tabla en el momento especificado.

```

aws dynamodb restore-table-to-point-in-time \
  --source-table-name MusicCollection \
  --target-table-name MusicCollectionRestore \
  --restore-date-time 1576622404.0

```

Salida:

```

{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      }
    ]
  }
}

```

```

    },
    {
      "AttributeName": "SongTitle",
      "AttributeType": "S"
    }
  ],
  "TableName": "MusicCollectionRestore",
  "KeySchema": [
    {
      "AttributeName": "Artist",
      "KeyType": "HASH"
    },
    {
      "AttributeName": "SongTitle",
      "KeyType": "RANGE"
    }
  ],
  "TableStatus": "CREATING",
  "CreationDateTime": 1576623311.86,
  "ProvisionedThroughput": {
    "NumberOfDecreasesToday": 0,
    "ReadCapacityUnits": 5,
    "WriteCapacityUnits": 5
  },
  "TableSizeBytes": 0,
  "ItemCount": 0,
  "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollectionRestore",
  "TableId": "befd9e0e-1843-4dc6-a147-d6d00e85cb1f",
  "BillingModeSummary": {
    "BillingMode": "PROVISIONED"
  },
  "RestoreSummary": {
    "SourceTableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
    "RestoreDateTime": 1576622404.0,
    "RestoreInProgress": true
  }
}
}
}

```

Para obtener más información, consulte [Point-in-Time Recovery for DynamoDB en la Guía para desarrolladores de Amazon DynamoDB](#).

- Para obtener API más información, consulte la Referencia de comandos.  
[RestoreTableToPointInTime](#) AWS CLI

## scan

En el siguiente ejemplo de código, se muestra cómo usar scan.

### AWS CLI

#### Análisis de una tabla

En el siguiente ejemplo de scan se escanea toda la tabla `MusicCollection` y, a continuación, se reducen los resultados a las canciones del artista “No One You Know”. Para cada elemento, solo se devuelven el título del álbum y el título de la canción.

```
aws dynamodb scan \  
  --table-name MusicCollection \  
  --filter-expression "Artist = :a" \  
  --projection-expression "#ST, #AT" \  
  --expression-attribute-names file://expression-attribute-names.json \  
  --expression-attribute-values file://expression-attribute-values.json
```

Contenidos de `expression-attribute-names.json`:

```
{  
  "#ST": "SongTitle",  
  "#AT": "AlbumTitle"  
}
```

Contenidos de `expression-attribute-values.json`:

```
{  
  ":a": {"S": "No One You Know"}  
}
```

Salida:

```
{  
  "Count": 2,  
  "Items": [  
    {  
      "AlbumTitle": "No One You Know",  
      "SongTitle": "No One You Know"  
    }  
  ]  
}
```

```
{
  "SongTitle": {
    "S": "Call Me Today"
  },
  "AlbumTitle": {
    "S": "Somewhat Famous"
  }
},
{
  "SongTitle": {
    "S": "Scared of My Shadow"
  },
  "AlbumTitle": {
    "S": "Blue Sky Blues"
  }
}
],
"ScannedCount": 3,
"ConsumedCapacity": null
}
```

Para obtener más información, consulte [Uso de operaciones de análisis en DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [Scan](#) in AWS CLI Command Reference.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

### AWS CLI

Para añadir etiquetas a un recurso de DynamoDB

En el siguiente tag-resource ejemplo, se agrega un par clave/valor de etiqueta a la tabla MusicCollection

```
aws dynamodb tag-resource \
  --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \
  --tags Key=Owner,Value=blueTeam
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos. [TagResourceAWS CLI](#)

## transact-get-items

En el siguiente ejemplo de código, se muestra cómo usar `transact-get-items`.

### AWS CLI

Para recuperar varios elementos de forma atómica de una o más tablas

El siguiente `transact-get-items` ejemplo recupera varios elementos de forma atómica.

```
aws dynamodb transact-get-items \  
  --transact-items file://transact-items.json \  
  --return-consumed-capacity TOTAL
```

Contenidos de `transact-items.json`:

```
[  
  {  
    "Get": {  
      "Key": {  
        "Artist": {"S": "Acme Band"},  
        "SongTitle": {"S": "Happy Day"}  
      },  
      "TableName": "MusicCollection"  
    }  
  },  
  {  
    "Get": {  
      "Key": {  
        "Artist": {"S": "No One You Know"},  
        "SongTitle": {"S": "Call Me Today"}  
      },  
      "TableName": "MusicCollection"  
    }  
  }  
]
```

**Salida:**

```
{
  "ConsumedCapacity": [
    {
      "TableName": "MusicCollection",
      "CapacityUnits": 4.0,
      "ReadCapacityUnits": 4.0
    }
  ],
  "Responses": [
    {
      "Item": {
        "AlbumTitle": {
          "S": "Songs About Life"
        },
        "Artist": {
          "S": "Acme Band"
        },
        "SongTitle": {
          "S": "Happy Day"
        }
      }
    },
    {
      "Item": {
        "AlbumTitle": {
          "S": "Somewhat Famous"
        },
        "Artist": {
          "S": "No One You Know"
        },
        "SongTitle": {
          "S": "Call Me Today"
        }
      }
    }
  ]
}
```

Para obtener más información, consulte [Gestión de flujos de trabajo complejos con transacciones de DynamoDB en la Guía para desarrolladores](#) de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [TransactGetItems](#).AWS CLI

## transact-write-items

En el siguiente ejemplo de código, se muestra cómo usar `transact-write-items`.

### AWS CLI

Ejemplo 1: escribir elementos de forma atómica en una o más tablas

En el siguiente `transact-write-items` ejemplo, se actualiza un elemento y se elimina otro. La operación falla si alguna de las operaciones falla o si alguno de los elementos contiene un `Rating` atributo.

```
aws dynamodb transact-write-items \  
  --transact-items file://transact-items.json \  
  --return-consumed-capacity TOTAL \  
  --return-item-collection-metrics SIZE
```

Contenido del `transact-items.json` archivo:

```
[  
  {  
    "Update": {  
      "Key": {  
        "Artist": {"S": "Acme Band"},  
        "SongTitle": {"S": "Happy Day"}  
      },  
      "UpdateExpression": "SET AlbumTitle = :newval",  
      "ExpressionAttributeValues": {  
        ":newval": {"S": "Updated Album Title"}  
      },  
      "TableName": "MusicCollection",  
      "ConditionExpression": "attribute_not_exists(Rating)"  
    },  
    {  
      "Delete": {  
        "Key": {  
          "Artist": {"S": "No One You Know"},  
          "SongTitle": {"S": "Call Me Today"}  
        }  
      }  
    }  
  }  
]
```

```

    },
    "TableName": "MusicCollection",
    "ConditionExpression": "attribute_not_exists(Rating)"
  }
}
]

```

**Salida:**

```

{
  "ConsumedCapacity": [
    {
      "TableName": "MusicCollection",
      "CapacityUnits": 10.0,
      "WriteCapacityUnits": 10.0
    }
  ],
  "ItemCollectionMetrics": {
    "MusicCollection": [
      {
        "ItemCollectionKey": {
          "Artist": {
            "S": "No One You Know"
          }
        },
        "SizeEstimateRangeGB": [
          0.0,
          1.0
        ]
      },
      {
        "ItemCollectionKey": {
          "Artist": {
            "S": "Acme Band"
          }
        },
        "SizeEstimateRangeGB": [
          0.0,
          1.0
        ]
      }
    ]
  }
}

```



```
}
```

Para obtener más información, consulte [Gestión de flujos de trabajo complejos con transacciones de DynamoDB en la Guía para desarrolladores](#) de Amazon DynamoDB.

Ejemplo 2: escribir elementos de forma atómica mediante un token de solicitud de un cliente

El siguiente comando usa un token de solicitud del cliente para realizar la llamada a `transact-write-items` idempotent, lo que significa que varias llamadas tienen el mismo efecto que una sola llamada.

```
aws dynamodb transact-write-items \  
  --transact-items file://transact-items.json \  
  --client-request-token abc123
```

Contenido del archivo: `transact-items.json`

```
[  
  {  
    "Update": {  
      "Key": {  
        "Artist": {"S": "Acme Band"},  
        "SongTitle": {"S": "Happy Day"}  
      },  
      "UpdateExpression": "SET AlbumTitle = :newval",  
      "ExpressionAttributeValues": {  
        ":newval": {"S": "Updated Album Title"}  
      },  
      "TableName": "MusicCollection",  
      "ConditionExpression": "attribute_not_exists(Rating)"  
    },  
    {  
      "Delete": {  
        "Key": {  
          "Artist": {"S": "No One You Know"},  
          "SongTitle": {"S": "Call Me Today"}  
        },  
        "TableName": "MusicCollection",  
        "ConditionExpression": "attribute_not_exists(Rating)"  
      }  
    }  
  ]
```

```
] ]
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión de flujos de trabajo complejos con transacciones de DynamoDB en la Guía para desarrolladores](#) de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [TransactWriteItems](#).AWS CLI

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

AWS CLI

Para eliminar una etiqueta de un recurso de DynamoDB

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta con la clave `Owner` de la `MusicCollection` tabla.

```
aws dynamodb untag-resource \  
  --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \  
  --tag-keys Owner
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado para DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos. [UntagResource](#)AWS CLI

## update-continuous-backups

En el siguiente ejemplo de código, se muestra cómo usar `update-continuous-backups`.

AWS CLI

Para actualizar la configuración de copia de seguridad continua de una tabla de DynamoDB

El siguiente `update-continuous-backups` ejemplo permite la point-in-time recuperación de la `MusicCollection` tabla.

```
aws dynamodb update-continuous-backups \  
  --table-name MusicCollection \  
  --point-in-time-recovery-specification PointInTimeRecoveryEnabled=true
```

Salida:

```
{  
  "ContinuousBackupsDescription": {  
    "ContinuousBackupsStatus": "ENABLED",  
    "PointInTimeRecoveryDescription": {  
      "PointInTimeRecoveryStatus": "ENABLED",  
      "EarliestRestorableDateTime": 1576622404.0,  
      "LatestRestorableDateTime": 1576622404.0  
    }  
  }  
}
```

Para obtener más información, consulte [Point-in-Time Recovery for DynamoDB en la Guía para desarrolladores de Amazon DynamoDB](#).

- Para obtener API más información, consulte la Referencia de comandos.  
[UpdateContinuousBackups](#) AWS CLI

## update-contributor-insights

En el siguiente ejemplo de código, se muestra cómo usar `update-contributor-insights`.

AWS CLI

Para habilitar Contributor Insights en una tabla

El siguiente `update-contributor-insights` ejemplo habilita Contributor Insights en la `MusicCollection` tabla y en el índice secundario `AlbumTitle-index` global.

```
aws dynamodb update-contributor-insights \  
  --table-name MusicCollection \  
  --index-name AlbumTitle-index \  
  --contributor-insights-action ENABLE
```

Salida:

```
{
  "TableName": "MusicCollection",
  "IndexName": "AlbumTitle-index",
  "ContributorInsightsStatus": "ENABLING"
}
```

Para obtener más información, consulte [Análisis del acceso a los datos mediante CloudWatch Contributor Insights for DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [UpdateContributorInsights](#).AWS CLI

## update-global-table-settings

En el siguiente ejemplo de código, se muestra cómo usar `update-global-table-settings`.

### AWS CLI

Para actualizar la configuración de la capacidad de escritura aprovisionada en una tabla global de DynamoDB

El siguiente `update-global-table-settings` ejemplo establece la capacidad de escritura aprovisionada de la tabla `MusicCollection` global en 15.

```
aws dynamodb update-global-table-settings \
  --global-table-name MusicCollection \
  --global-table-provisioned-write-capacity-units 15
```

Salida:

```
{
  "GlobalTableName": "MusicCollection",
  "ReplicaSettings": [
    {
      "RegionName": "eu-west-1",
      "ReplicaStatus": "UPDATING",
      "ReplicaProvisionedReadCapacityUnits": 10,
      "ReplicaProvisionedReadCapacityAutoScalingSettings": {
        "AutoScalingDisabled": true
      },
      "ReplicaProvisionedWriteCapacityUnits": 10,
      "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
```

```

        "AutoScalingDisabled": true
    },
    {
        "RegionName": "us-east-1",
        "ReplicaStatus": "UPDATING",
        "ReplicaProvisionedReadCapacityUnits": 10,
        "ReplicaProvisionedReadCapacityAutoScalingSettings": {
            "AutoScalingDisabled": true
        },
        "ReplicaProvisionedWriteCapacityUnits": 10,
        "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
            "AutoScalingDisabled": true
        }
    },
    {
        "RegionName": "us-east-2",
        "ReplicaStatus": "UPDATING",
        "ReplicaProvisionedReadCapacityUnits": 10,
        "ReplicaProvisionedReadCapacityAutoScalingSettings": {
            "AutoScalingDisabled": true
        },
        "ReplicaProvisionedWriteCapacityUnits": 10,
        "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
            "AutoScalingDisabled": true
        }
    }
]
}

```

Para obtener más información, consulte las tablas [globales de DynamoDB en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [UpdateGlobalTableSettings](#).AWS CLI

## update-global-table

En el siguiente ejemplo de código, se muestra cómo usar `update-global-table`.

### AWS CLI

Para actualizar una tabla global de DynamoDB

En el siguiente `update-global-table` ejemplo, se agrega una réplica de la región especificada a la tabla `MusicCollection` global.

```
aws dynamodb update-global-table \  
  --global-table-name MusicCollection \  
  --replica-updates Create={RegionName=eu-west-1}
```

Salida:

```
{  
  "GlobalTableDescription": {  
    "ReplicationGroup": [  
      {  
        "RegionName": "eu-west-1"  
      },  
      {  
        "RegionName": "us-east-2"  
      },  
      {  
        "RegionName": "us-east-1"  
      }  
    ],  
    "GlobalTableArn": "arn:aws:dynamodb::123456789012:global-table/  
MusicCollection",  
    "CreationDateTime": 1576625818.532,  
    "GlobalTableStatus": "ACTIVE",  
    "GlobalTableName": "MusicCollection"  
  }  
}
```

Para obtener más información, consulte las tablas [globales de DynamoDB en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [UpdateGlobalTable](#).AWS CLI

## update-item

En el siguiente ejemplo de código, se muestra cómo usar `update-item`.

## AWS CLI

### Ejemplo 1: Actualización de un elemento de una tabla

En el siguiente ejemplo de `update-item`, se actualiza un elemento de la tabla `MusicCollection`. Añade un nuevo atributo (`Year`) y modifica el atributo `AlbumTitle`. En la respuesta se muestran todos los atributos del elemento, tal como aparecen después de la actualización.

```
aws dynamodb update-item \  
  --table-name MusicCollection \  
  --key file://key.json \  
  --update-expression "SET #Y = :y, #AT = :t" \  
  --expression-attribute-names file://expression-attribute-names.json \  
  --expression-attribute-values file://expression-attribute-values.json \  
  --return-values ALL_NEW \  
  --return-consumed-capacity TOTAL \  
  --return-item-collection-metrics SIZE
```

Contenidos de `key.json`:

```
{  
  "Artist": {"S": "Acme Band"},  
  "SongTitle": {"S": "Happy Day"}  
}
```

Contenidos de `expression-attribute-names.json`:

```
{  
  "#Y": "Year", "#AT": "AlbumTitle"  
}
```

Contenidos de `expression-attribute-values.json`:

```
{  
  ":y": {"N": "2015"},  
  ":t": {"S": "Louder Than Ever"}  
}
```

Salida:

```
{
  "Attributes": {
    "AlbumTitle": {
      "S": "Louder Than Ever"
    },
    "Awards": {
      "N": "10"
    },
    "Artist": {
      "S": "Acme Band"
    },
    "Year": {
      "N": "2015"
    },
    "SongTitle": {
      "S": "Happy Day"
    }
  },
  "ConsumedCapacity": {
    "TableName": "MusicCollection",
    "CapacityUnits": 3.0
  },
  "ItemCollectionMetrics": {
    "ItemCollectionKey": {
      "Artist": {
        "S": "Acme Band"
      }
    },
    "SizeEstimateRangeGB": [
      0.0,
      1.0
    ]
  }
}
```

Para obtener más información, consulte [Escritura de un elemento](#) en la Guía para desarrolladores de Amazon DynamoDB.

## Ejemplo 2: Actualización de un elemento de forma condicional

En el siguiente ejemplo se actualiza un elemento de la tabla `MusicCollection`, pero solo si el elemento existente aún no tiene un atributo `Year`.



```
aws dynamodb update-item \  
  --table-name MusicCollection \  
  --key file://key.json \  
  --update-expression "SET #Y = :y, #AT = :t" \  
  --expression-attribute-names file://expression-attribute-names.json \  
  --expression-attribute-values file://expression-attribute-values.json \  
  --condition-expression "attribute_not_exists(#Y)"
```

Contenidos de `key.json`:

```
{  
  "Artist": {"S": "Acme Band"},  
  "SongTitle": {"S": "Happy Day"}  
}
```

Contenidos de `expression-attribute-names.json`:

```
{  
  "#Y": "Year",  
  "#AT": "AlbumTitle"  
}
```

Contenidos de `expression-attribute-values.json`:

```
{  
  ":y": {"N": "2015"},  
  ":t": {"S": "Louder Than Ever"}  
}
```

Si el elemento ya tiene un atributo `Year`, DynamoDB devuelve el siguiente resultado.

```
An error occurred (ConditionalCheckFailedException) when calling the UpdateItem  
operation: The conditional request failed
```

Para obtener más información, consulte [Escritura de un elemento](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [UpdateItem](#) la Referencia de AWS CLI comandos.

## update-table-replica-auto-scaling

En el siguiente ejemplo de código, se muestra cómo usar `update-table-replica-auto-scaling`.

### AWS CLI

Para actualizar la configuración de escalado automático en todas las réplicas de una tabla global

El siguiente `update-table-replica-auto-scaling` ejemplo actualiza la configuración de escalado automático de la capacidad de escritura en todas las réplicas de la tabla global especificada.

```
aws dynamodb update-table-replica-auto-scaling \  
  --table-name MusicCollection \  
  --provisioned-write-capacity-auto-scaling-update file://auto-scaling-policy.json
```

Contenidos de `auto-scaling-policy.json`:

```
{  
  "MinimumUnits": 10,  
  "MaximumUnits": 100,  
  "AutoScalingDisabled": false,  
  "ScalingPolicyUpdate": {  
    "PolicyName": "DynamoDBWriteCapacityUtilization:table/MusicCollection",  
    "TargetTrackingScalingPolicyConfiguration": {  
      "TargetValue": 80  
    }  
  }  
}
```

Salida:

```
{  
  "TableAutoScalingDescription": {  
    "TableName": "MusicCollection",  
    "TableStatus": "ACTIVE",  
    "Replicas": [  
      {  
        "RegionName": "eu-central-1",  
        "GlobalSecondaryIndexes": [],  
        "ReplicaProvisionedReadCapacityAutoScalingSettings": {
```

```
        "MinimumUnits": 5,
        "MaximumUnits": 40000,
        "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
        "ScalingPolicies": [
            {
                "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
                "TargetTrackingScalingPolicyConfiguration": {
                    "TargetValue": 70.0
                }
            }
        ],
    },
    "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
        "MinimumUnits": 10,
        "MaximumUnits": 100,
        "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
        "ScalingPolicies": [
            {
                "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
                "TargetTrackingScalingPolicyConfiguration": {
                    "TargetValue": 80.0
                }
            }
        ]
    },
    "ReplicaStatus": "ACTIVE"
},
{
    "RegionName": "us-east-1",
    "GlobalSecondaryIndexes": [],
    "ReplicaProvisionedReadCapacityAutoScalingSettings": {
        "MinimumUnits": 5,
        "MaximumUnits": 40000,
        "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
        "ScalingPolicies": [
            {
```

```

        "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
        "TargetTrackingScalingPolicyConfiguration": {
            "TargetValue": 70.0
        }
    }
]
},
"ReplicaProvisionedWriteCapacityAutoScalingSettings": {
    "MinimumUnits": 10,
    "MaximumUnits": 100,
    "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
    "ScalingPolicies": [
        {
            "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
            "TargetTrackingScalingPolicyConfiguration": {
                "TargetValue": 80.0
            }
        }
    ]
},
"ReplicaStatus": "ACTIVE"
},
{
    "RegionName": "us-east-2",
    "GlobalSecondaryIndexes": [],
    "ReplicaProvisionedReadCapacityAutoScalingSettings": {
        "MinimumUnits": 5,
        "MaximumUnits": 40000,
        "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
        "ScalingPolicies": [
            {
                "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
                "TargetTrackingScalingPolicyConfiguration": {
                    "TargetValue": 70.0
                }
            }
        ]
    }
}
]

```

```

    },
    "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
      "MinimumUnits": 10,
      "MaximumUnits": 100,
      "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
      "ScalingPolicies": [
        {
          "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
          "TargetTrackingScalingPolicyConfiguration": {
            "TargetValue": 80.0
          }
        }
      ]
    },
    "ReplicaStatus": "ACTIVE"
  }
]
}

```

Para obtener más información, consulte las tablas [globales de DynamoDB en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [UpdateTableReplicaAutoScaling](#).AWS CLI

## update-table

En el siguiente ejemplo de código, se muestra cómo usar `update-table`.

### AWS CLI

Ejemplo 1: para modificar el modo de facturación de una tabla

El siguiente ejemplo de `update-table` aumenta la capacidad de lectura y escritura aprovisionada en la tabla `MusicCollection`.

```

aws dynamodb update-table \
  --table-name MusicCollection \
  --billing-mode PROVISIONED \

```

```
--provisioned-throughput ReadCapacityUnits=15,WriteCapacityUnits=10
```

Salida:

```
{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "AlbumTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],
    "TableName": "MusicCollection",
    "KeySchema": [
      {
        "AttributeName": "Artist",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "SongTitle",
        "KeyType": "RANGE"
      }
    ],
    "TableStatus": "UPDATING",
    "CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
    "ProvisionedThroughput": {
      "LastIncreaseDateTime": "2020-07-28T13:18:18.921000-07:00",
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 15,
      "WriteCapacityUnits": 10
    },
    "TableSizeBytes": 182,
    "ItemCount": 2,
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
    "TableId": "abcd0123-01ab-23cd-0123-abcdef123456",
  }
}
```

```

    "BillingModeSummary": {
      "BillingMode": "PROVISIONED",
      "LastUpdateToPayPerRequestDateTime": "2020-07-28T13:14:48.366000-07:00"
    }
  }
}

```

Para obtener más información, consulte [Actualización de una tabla](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 5: crear un índice secundario global

En el siguiente ejemplo se añade un índice secundario global a la tabla `MusicCollection`.

```

aws dynamodb update-table \
  --table-name MusicCollection \
  --attribute-definitions AttributeName=AlbumTitle,AttributeType=S \
  --global-secondary-index-updates file://gsi-updates.json

```

Contenidos de `gsi-updates.json`:

```

[
  {
    "Create": {
      "IndexName": "AlbumTitle-index",
      "KeySchema": [
        {
          "AttributeName": "AlbumTitle",
          "KeyType": "HASH"
        }
      ],
      "ProvisionedThroughput": {
        "ReadCapacityUnits": 10,
        "WriteCapacityUnits": 10
      },
      "Projection": {
        "ProjectionType": "ALL"
      }
    }
  }
]

```

Salida:

```
{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "AlbumTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],
    "TableName": "MusicCollection",
    "KeySchema": [
      {
        "AttributeName": "Artist",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "SongTitle",
        "KeyType": "RANGE"
      }
    ],
    "TableStatus": "UPDATING",
    "CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
    "ProvisionedThroughput": {
      "LastIncreaseDateTime": "2020-07-28T12:59:17.537000-07:00",
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 15,
      "WriteCapacityUnits": 10
    },
    "TableSizeBytes": 182,
    "ItemCount": 2,
    "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
    "TableId": "abcd0123-01ab-23cd-0123-abcdef123456",
    "BillingModeSummary": {
      "BillingMode": "PROVISIONED",
      "LastUpdateToPayPerRequestDateTime": "2020-07-28T13:14:48.366000-07:00"
    }
  },
}
```



```

    "GlobalSecondaryIndexes": [
      {
        "IndexName": "AlbumTitle-index",
        "KeySchema": [
          {
            "AttributeName": "AlbumTitle",
            "KeyType": "HASH"
          }
        ],
        "Projection": {
          "ProjectionType": "ALL"
        },
        "IndexStatus": "CREATING",
        "Backfilling": false,
        "ProvisionedThroughput": {
          "NumberOfDecreasesToday": 0,
          "ReadCapacityUnits": 10,
          "WriteCapacityUnits": 10
        },
        "IndexSizeBytes": 0,
        "ItemCount": 0,
        "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitle-index"
      }
    ]
  }
}

```

Para obtener más información, consulte [Actualización de una tabla](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 3: activar DynamoDB Streams en una tabla

El siguiente comando activa DynamoDB Streams en la tabla `MusicCollection`.

```

aws dynamodb update-table \
  --table-name MusicCollection \
  --stream-specification StreamEnabled=true,StreamViewType=NEW_IMAGE

```

Salida:

```

{
  "TableDescription": {

```

```
"AttributeDefinitions": [
  {
    "AttributeName": "AlbumTitle",
    "AttributeType": "S"
  },
  {
    "AttributeName": "Artist",
    "AttributeType": "S"
  },
  {
    "AttributeName": "SongTitle",
    "AttributeType": "S"
  }
],
"TableName": "MusicCollection",
"KeySchema": [
  {
    "AttributeName": "Artist",
    "KeyType": "HASH"
  },
  {
    "AttributeName": "SongTitle",
    "KeyType": "RANGE"
  }
],
"TableStatus": "UPDATING",
"CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
"ProvisionedThroughput": {
  "LastIncreaseDateTime": "2020-07-28T12:59:17.537000-07:00",
  "NumberOfDecreasesToday": 0,
  "ReadCapacityUnits": 15,
  "WriteCapacityUnits": 10
},
"TableSizeBytes": 182,
"ItemCount": 2,
"TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
"TableId": "abcd0123-01ab-23cd-0123-abcdef123456",
"BillingModeSummary": {
  "BillingMode": "PROVISIONED",
  "LastUpdateToPayPerRequestDateTime": "2020-07-28T13:14:48.366000-07:00"
},
"LocalSecondaryIndexes": [
  {
    "IndexName": "AlbumTitleIndex",
```

```
    "KeySchema": [
      {
        "AttributeName": "Artist",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "AlbumTitle",
        "KeyType": "RANGE"
      }
    ],
    "Projection": {
      "ProjectionType": "INCLUDE",
      "NonKeyAttributes": [
        "Year",
        "Genre"
      ]
    },
    "IndexSizeBytes": 139,
    "ItemCount": 2,
    "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitleIndex"
  }
],
"GlobalSecondaryIndexes": [
  {
    "IndexName": "AlbumTitle-index",
    "KeySchema": [
      {
        "AttributeName": "AlbumTitle",
        "KeyType": "HASH"
      }
    ],
    "Projection": {
      "ProjectionType": "ALL"
    },
    "IndexStatus": "ACTIVE",
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 10,
      "WriteCapacityUnits": 10
    },
    "IndexSizeBytes": 0,
    "ItemCount": 0,
```

```

        "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitle-index"
    }
  ],
  "StreamSpecification": {
    "StreamEnabled": true,
    "StreamViewType": "NEW_IMAGE"
  },
  "LatestStreamLabel": "2020-07-28T21:53:39.112",
  "LatestStreamArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/stream/2020-07-28T21:53:39.112"
}
}

```

Para obtener más información, consulte [Actualización de una tabla](#) en la Guía para desarrolladores de Amazon DynamoDB.

Ejemplo 4: activar el cifrado en el lado del servidor

El siguiente ejemplo activa el cifrado del lado del servidor en la tabla `MusicCollection`.

```

aws dynamodb update-table \
  --table-name MusicCollection \
  --sse-specification Enabled=true,SSEType=KMS

```

Salida:

```

{
  "TableDescription": {
    "AttributeDefinitions": [
      {
        "AttributeName": "AlbumTitle",
        "AttributeType": "S"
      },
      {
        "AttributeName": "Artist",
        "AttributeType": "S"
      },
      {
        "AttributeName": "SongTitle",
        "AttributeType": "S"
      }
    ],

```

```
"TableName": "MusicCollection",
"KeySchema": [
  {
    "AttributeName": "Artist",
    "KeyType": "HASH"
  },
  {
    "AttributeName": "SongTitle",
    "KeyType": "RANGE"
  }
],
"TableStatus": "ACTIVE",
"CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
"ProvisionedThroughput": {
  "LastIncreaseDateTime": "2020-07-28T12:59:17.537000-07:00",
  "NumberOfDecreasesToday": 0,
  "ReadCapacityUnits": 15,
  "WriteCapacityUnits": 10
},
"TableSizeBytes": 182,
"ItemCount": 2,
"TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
"TableId": "abcd0123-01ab-23cd-0123-abcdef123456",
"BillingModeSummary": {
  "BillingMode": "PROVISIONED",
  "LastUpdateToPayPerRequestDateTime": "2020-07-28T13:14:48.366000-07:00"
},
"LocalSecondaryIndexes": [
  {
    "IndexName": "AlbumTitleIndex",
    "KeySchema": [
      {
        "AttributeName": "Artist",
        "KeyType": "HASH"
      },
      {
        "AttributeName": "AlbumTitle",
        "KeyType": "RANGE"
      }
    ],
    "Projection": {
      "ProjectionType": "INCLUDE",
      "NonKeyAttributes": [
        "Year",
```

```

        "Genre"
      ]
    },
    "IndexSizeBytes": 139,
    "ItemCount": 2,
    "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitleIndex"
  }
],
"GlobalSecondaryIndexes": [
  {
    "IndexName": "AlbumTitle-index",
    "KeySchema": [
      {
        "AttributeName": "AlbumTitle",
        "KeyType": "HASH"
      }
    ],
    "Projection": {
      "ProjectionType": "ALL"
    },
    "IndexStatus": "ACTIVE",
    "ProvisionedThroughput": {
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 10,
      "WriteCapacityUnits": 10
    },
    "IndexSizeBytes": 0,
    "ItemCount": 0,
    "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitle-index"
  }
],
"StreamSpecification": {
  "StreamEnabled": true,
  "StreamViewType": "NEW_IMAGE"
},
"LatestStreamLabel": "2020-07-28T21:53:39.112",
"LatestStreamArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/stream/2020-07-28T21:53:39.112",
"SSEDescription": {
  "Status": "UPDATING"
}
}

```

```
}
```

Para obtener más información, consulte [Actualización de una tabla](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [UpdateTable](#) la Referencia de AWS CLI comandos.

## update-time-to-live

En el siguiente ejemplo de código, se muestra cómo usar `update-time-to-live`.

### AWS CLI

Actualización de la configuración de tiempo de vida en una tabla

El siguiente ejemplo de `update-time-to-live` activa Tiempo de vida en la tabla especificada.

```
aws dynamodb update-time-to-live \  
  --table-name MusicCollection \  
  --time-to-live-specification Enabled=true,AttributeName=t1
```

Salida:

```
{  
  "TimeToLiveSpecification": {  
    "Enabled": true,  
    "AttributeName": "ttl"  
  }  
}
```

Para obtener más información, consulte [Tiempo de vida](#) en la Guía para desarrolladores de Amazon DynamoDB.

- Para API obtener más información, consulte [UpdateTimeToLive](#) la Referencia de AWS CLI comandos.

## Ejemplos de DynamoDB Streams que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante DynamoDB Streams. AWS Command Line Interface

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **describe-stream**

En el siguiente ejemplo de código, se muestra cómo usar `describe-stream`.

AWS CLI

Para obtener información sobre una transmisión de DynamoDB

El siguiente `describe-stream` comando muestra información sobre la transmisión específica de DynamoDB.

```
aws dynamodbstreams describe-stream \  
  --stream-arn arn:aws:dynamodb:us-west-1:123456789012:table/Music/  
stream/2019-10-22T18:02:01.576
```

Salida:

```
{  
  "StreamDescription": {  
    "StreamArn": "arn:aws:dynamodb:us-west-1:123456789012:table/Music/  
stream/2019-10-22T18:02:01.576",  
    "StreamLabel": "2019-10-22T18:02:01.576",  
    "StreamStatus": "ENABLED",  
    "StreamViewType": "NEW_AND_OLD_IMAGES",  
    "CreationRequestDateTime": 1571767321.571,  
    "TableName": "Music",  
    "KeySchema": [  
      {
```



```

        "AttributeName": "Artist",
        "KeyType": "HASH"
    },
    {
        "AttributeName": "SongTitle",
        "KeyType": "RANGE"
    }
],
"Shards": [
    {
        "ShardId": "shardId-00000001571767321804-697ce3d2",
        "SequenceNumberRange": {
            "StartingSequenceNumber": "4000000000000642977831",
            "EndingSequenceNumber": "4000000000000642977831"
        }
    },
    {
        "ShardId": "shardId-00000001571780995058-40810d86",
        "SequenceNumberRange": {
            "StartingSequenceNumber": "75740000000005655171150"
        },
        "ParentShardId": "shardId-00000001571767321804-697ce3d2"
    }
]
}
}

```

Para obtener más información, consulte [Capturar la actividad de una tabla con DynamoDB Streams en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [DescribeStream](#).AWS CLI

## get-records

En el siguiente ejemplo de código, se muestra cómo usar `get-records`.

### AWS CLI

Para obtener registros de una transmisión de Dynamodb

El siguiente `get-records` comando recupera los registros mediante el iterador de fragmentos de Amazon DynamoDB especificado.

```
aws dynamodbstreams get-records \
  --shard-iterator "arn:aws:dynamodb:us-west-1:123456789012:table/Music/
stream/2019-10-22T18:02:01.576|1|
AAAAAAAAAAGgM3YZ89vLZZxjmoQeo33r9M4x3+zmmTLsiL86MfrF4+B4EbsByi52InVmi0Nmy6xVW4IRcIIbs1z07MNI
+CjNPLqQjnyRSAnf0wWmKhL1/KNParWSfz2odf780o00bIDIWRRMkt7+Hyzh9SD
+hFxFAWR5C7QILOXPc8mRBfNIazfrVCjJK8/jsjCzsQnyXKzJbhh+GXCoxYN
+Kpmg4nyj1EAsYhbGL35muvHFoHjcyuynbsczbWaXNfThDwRAYvoTmc8XhHKtAWUbJiaVd8ZPtQwDsThCrmDRPIdmTRG
+w/LEGS05ha1qNP+VL4+tuHz2TRnhnJo/pny9GI/yGpce97mWvSPR5KPwy+DtcM5BHayBs
+PVYHITaTLiInFLT
+LCwvaz1QH3MY3b8A05Z800wjpkM60iQqtMeDwN4NX6FrcxR34JoFKGsgR8XkHVJzz2xr1xqSJ12ycpNTyHnndusw==
```

Salida:

```
{
  "Records": [
    {
      "eventID": "c3b5d798eef6215d42f8137b19a88e50",
      "eventName": "INSERT",
      "eventVersion": "1.1",
      "eventSource": "aws:dynamodb",
      "awsRegion": "us-west-1",
      "dynamodb": {
        "ApproximateCreationDateTime": 1571849028.0,
        "Keys": {
          "Artist": {
            "S": "No One You Know"
          },
          "SongTitle": {
            "S": "Call Me Today"
          }
        },
        "NewImage": {
          "AlbumTitle": {
            "S": "Somewhat Famous"
          },
          "Artist": {
            "S": "No One You Know"
          },
          "Awards": {
            "N": "1"
          },
          "SongTitle": {
            "S": "Call Me Today"
          }
        }
      }
    }
  ]
}
```

```

    }
  },
  "SequenceNumber": "700000000013256296913",
  "SizeBytes": 119,
  "StreamViewType": "NEW_AND_OLD_IMAGES"
}
},
{
  "eventID": "878960a6967867e2da16b27380a27328",
  "eventName": "INSERT",
  "eventVersion": "1.1",
  "eventSource": "aws:dynamodb",
  "awsRegion": "us-west-1",
  "dynamodb": {
    "ApproximateCreationDateTime": 1571849029.0,
    "Keys": {
      "Artist": {
        "S": "Acme Band"
      },
      "SongTitle": {
        "S": "Happy Day"
      }
    },
    "NewImage": {
      "AlbumTitle": {
        "S": "Songs About Life"
      },
      "Artist": {
        "S": "Acme Band"
      },
      "Awards": {
        "N": "10"
      },
      "SongTitle": {
        "S": "Happy Day"
      }
    },
    "SequenceNumber": "800000000013256297217",
    "SizeBytes": 100,
    "StreamViewType": "NEW_AND_OLD_IMAGES"
  }
},
{
  "eventID": "520fabde080e159fc3710b15ee1d4daa",

```

```
"eventName": "MODIFY",
"eventVersion": "1.1",
"eventSource": "aws:dynamodb",
"awsRegion": "us-west-1",
"dynamodb": {
  "ApproximateCreationDateTime": 1571849734.0,
  "Keys": {
    "Artist": {
      "S": "Acme Band"
    },
    "SongTitle": {
      "S": "Happy Day"
    }
  },
  "NewImage": {
    "AlbumTitle": {
      "S": "Updated Album Title"
    },
    "Artist": {
      "S": "Acme Band"
    },
    "Awards": {
      "N": "10"
    },
    "SongTitle": {
      "S": "Happy Day"
    }
  },
  "OldImage": {
    "AlbumTitle": {
      "S": "Songs About Life"
    },
    "Artist": {
      "S": "Acme Band"
    },
    "Awards": {
      "N": "10"
    },
    "SongTitle": {
      "S": "Happy Day"
    }
  },
  "SequenceNumber": "900000000013256687845",
  "SizeBytes": 170,
```

```

        "StreamViewType": "NEW_AND_OLD_IMAGES"
      }
    }
  ],
  "NextShardIterator": "arn:aws:dynamodb:us-west-1:123456789012:table/
Music/stream/2019-10-23T16:41:08.740|1|AAAAAAAAAAAAEhEI04jkFLW
+LK0wivjT8d/IHEh3iExV2xK00aTxEzVy1C1C7Kbb5+Z0W6bT9VQ2n1/
mrs7+PRia0ZCHJu7JHJVW7zlsq0i/ges3fw8GYEymyL+piEk35cx67rQqwKKyq
+Q6w9JyjreI0j4F2lWLV26lBwRTrIYC4IB7C3BZZK4715QwYdDxNdVHiSBRZX8UqoS6W0t0F87xZLNB9F/
NhYBLXi/wcGvAcBcC0TNI0H+N0Nqwt0B/
FGckNrf8YZ0xRoNN6RgGuVWHF3px0hxEJeFZoSoJTIKeG9YcYxzi5Ci/
mhdTm7tBXnbw5c6xmsGsBqTirNjldyJLcWl8Cl0U0LX63Ufo/5QlitztcjEbKsQe28x8LM8o7VH1Is0fF/
ITt8awSA4igyJS0P87GN8Qri8kj8iaE35805jBHWf2wvwT6Iy2xGrR2r2HzYps9dwG0arVdEITaJfWzNoL4HajMhmREZ
+V04i1YIeHMXJfcwetNRuIbdQXfJht2NQZa4PVV6iknY6d19MrdbSTMKoqAuvp6g3Q2jH4t7GKCLWgodcPAn8g5+43Da
}

```

Para obtener más información, consulte [Capturar la actividad de una tabla con DynamoDB Streams en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [GetRecords](#).AWS CLI

## get-shard-iterator

En el siguiente ejemplo de código, se muestra cómo usar `get-shard-iterator`.

AWS CLI

Para obtener un iterador de fragmentos

El siguiente `get-shard-iterator` comando recupera un iterador de fragmentos para el fragmento especificado.

```

aws dynamodbstreams get-shard-iterator \
  --stream-arn arn:aws:dynamodb:us-west-1:12356789012:table/Music/  

stream/2019-10-22T18:02:01.576 \  

  --shard-id shardId-00000001571780995058-40810d86 \  

  --shard-iterator-type LATEST

```

Salida:

```
{
```

```

    "ShardIterator": "arn:aws:dynamodb:us-west-1:123456789012:table/Music/
stream/2019-10-22T18:02:01.576|1|
AAAAAAAAAAGgM3YZ89vLZZxjmoQeo33r9M4x3+zmmTLsiL86MfrF4+B4EbsByi52InVmi0Nmy6xVW4IRcIIbs1z07MNI
+CjNPlqQjnyRSAnf0wWmKhL1/KNParWSfz2odf780o00bIDIWRRMkt7+Hyzh9SD
+hFxFAWR5C7QI10XPc8mRBfNIazfrVCjJK8/jsjCzsqNyXKzJbhh+GXCoxYN
+Kpmg4nyj1EAsYhbGL35muvHFoHjcyuynbsczbWaXNfThDwRAYvoTmc8XhHKtAWUbJiaVd8ZPtQwDsThCrmDRPIdmTRG
+w/1EGS05ha1qNP+V14+tuhz2TRnhnJo/pny9GI/yGpce97mWvSPr5KPwy+DtcM5BHayBs
+PVYHITaTliInFlT
+LCwvaz1QH3MY3b8A05Z800wjpktm60iQqtMeDwN4NX6FrcxR34JoFKGsgR8XkHVJzz2xr1xqSJ12ycpNTyHndusw==
}

```

Para obtener más información, consulte [Capturar la actividad de una tabla con DynamoDB Streams en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [GetShardIterator](#).AWS CLI

## list-streams

En el siguiente ejemplo de código, se muestra cómo usar `list-streams`.

### AWS CLI

Para mostrar una lista de transmisiones de DynamoDB

El siguiente `list-streams` comando muestra todas las transmisiones de Amazon DynamoDB existentes en la región predeterminada. AWS

```
aws dynamodbstreams list-streams
```

Salida:

```

{
  "Streams": [
    {
      "StreamArn": "arn:aws:dynamodb:us-west-1:123456789012:table/Music/
stream/2019-10-22T18:02:01.576",
      "TableName": "Music",
      "StreamLabel": "2019-10-22T18:02:01.576"
    }
  ]
}

```

Para obtener más información, consulte [Capturar la actividad de una tabla con DynamoDB Streams en la Guía](#) para desarrolladores de Amazon DynamoDB.

- Para obtener API más información, consulte la Referencia de comandos [ListStreams](#).AWS CLI

## EC2Ejemplos de Amazon que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante AmazonEC2.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **accept-address-transfer**

En el siguiente ejemplo de código, se muestra cómo usar `accept-address-transfer`.

AWS CLI

Para aceptar la transferencia de una dirección IP elástica a su cuenta

El siguiente `accept-address-transfer` ejemplo acepta la transferencia de la dirección IP elástica especificada a su cuenta.

```
aws ec2 accept-address-transfer \  
  --address 100.21.184.216
```

Salida:

```
{  
  "AddressTransfer": {
```

```
"PublicIp": "100.21.184.216",
"AllocationId": "eipalloc-09ad461b0d03f6aaf",
"TransferAccountId": "123456789012",
"TransferOfferExpirationTimestamp": "2023-02-22T20:51:10.000Z",
"TransferOfferAcceptedTimestamp": "2023-02-22T22:52:54.000Z",
"AddressTransferStatus": "accepted"
}
}
```

Para obtener más información, consulta [Transferir direcciones IP elásticas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [AcceptAddressTransfer](#) la Referencia de AWS CLI comandos.

## accept-reserved-instances-exchange-quote

En el siguiente ejemplo de código, se muestra cómo usar `accept-reserved-instances-exchange-quote`.

### AWS CLI

Para realizar un intercambio de instancias reservadas convertibles

En este ejemplo, se realiza un intercambio de las instancias reservadas convertibles especificadas.

Comando:

```
aws ec2 accept-reserved-instances-exchange-quote --reserved-
instance-ids 7b8750c3-397e-4da4-bbcb-a45ebexample --target-
configurations OfferingId=b747b472-423c-48f3-8cee-679bcexample
```

Salida:

```
{
  "ExchangeId": "riex-e68ed3c1-8bc8-4c17-af77-811afexample"
}
```

- Para API obtener más información, consulte [AcceptReservedInstancesExchangeQuote](#) la Referencia de AWS CLI comandos.



## accept-transit-gateway-peering-attachment

En el siguiente ejemplo de código, se muestra cómo usar `accept-transit-gateway-peering-attachment`.

### AWS CLI

Para aceptar un adjunto de peering de Transit Gateway

El siguiente `accept-transit-gateway-peering-attachment` ejemplo acepta el adjunto de interconexión de la pasarela de tránsito especificado. El `--region` parámetro especifica la región en la que se encuentra la puerta de enlace de tránsito aceptadora.

```
aws ec2 accept-transit-gateway-peering-attachment \  
  --transit-gateway-attachment-id tgw-attach-4455667788aabbccd \  
  --region us-east-2
```

Salida:

```
{  
  "TransitGatewayPeeringAttachment": {  
    "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",  
    "RequesterTgwInfo": {  
      "TransitGatewayId": "tgw-123abc05e04123abc",  
      "OwnerId": "123456789012",  
      "Region": "us-west-2"  
    },  
    "AcceptorTgwInfo": {  
      "TransitGatewayId": "tgw-11223344aabbcc112",  
      "OwnerId": "123456789012",  
      "Region": "us-east-2"  
    },  
    "State": "pending",  
    "CreationTime": "2019-12-09T11:38:31.000Z"  
  }  
}
```

Para obtener más información, consulte los [adjuntos de peering de Transit Gateway](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [AcceptTransitGatewayPeeringAttachment](#) de AWS CLI comandos.

## accept-transit-gateway-vpc-attachment

En el siguiente ejemplo de código, se muestra cómo usar `accept-transit-gateway-vpc-attachment`.

### AWS CLI

Para aceptar una solicitud para adjuntar un VPC a una pasarela de tránsito.

En el siguiente `accept-transit-gateway-vpc-attachment` ejemplo, se acepta la solicitud de un adjunto específico.

```
aws ec2 accept-transit-gateway-vpc-attachment \
  --transit-gateway-attachment-id tgw-attach-0a34fe6b4fEXAMPLE
```

Salida:

```
{
  "TransitGatewayVpcAttachment": {
    "TransitGatewayAttachmentId": "tgw-attach-0a34fe6b4fEXAMPLE",
    "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
    "VpcId": "vpc-07e8ffd50fEXAMPLE",
    "VpcOwnerId": "123456789012",
    "State": "pending",
    "SubnetIds": [
      "subnet-0752213d59EXAMPLE"
    ],
    "CreationTime": "2019-07-10T17:33:46.000Z",
    "Options": {
      "DnsSupport": "enable",
      "Ipv6Support": "disable"
    }
  }
}
```

Para obtener más información, consulte [Adjuntos a a de Transit Gateways VPC](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [AcceptTransitGatewayVpcAttachment](#) la Referencia de AWS CLI comandos.

## accept-vpc-endpoint-connections

En el siguiente ejemplo de código, se muestra cómo usar `accept-vpc-endpoint-connections`.

### AWS CLI

Para aceptar una solicitud de conexión de punto final de interfaz

En este ejemplo, se acepta la solicitud de conexión de punto final especificada para el servicio de punto final especificado.

Comando:

```
aws ec2 accept-vpc-endpoint-connections --service-id vpce-svc-03d5ebb7d9579a2b3 --  
vpc-endpoint-ids vpce-0c1308d7312217abc
```

Salida:

```
{  
  "Unsuccessful": []  
}
```

- Para API obtener más información, consulte [AcceptVpcEndpointConnections](#) la Referencia de AWS CLI comandos.

## accept-vpc-peering-connection

En el siguiente ejemplo de código, se muestra cómo usar `accept-vpc-peering-connection`.

### AWS CLI

Para aceptar una conexión VPC entre pares

En este ejemplo se acepta la solicitud de conexión entre VPC pares especificada.

Comando:

```
aws ec2 accept-vpc-peering-connection --vpc-peering-connection-id pcx-1a2b3c4d
```

Salida:

```
{
  "VpcPeeringConnection": {
    "Status": {
      "Message": "Provisioning",
      "Code": "provisioning"
    },
    "Tags": [],
    "AccepterVpcInfo": {
      "OwnerId": "444455556666",
      "VpcId": "vpc-44455566",
      "CidrBlock": "10.0.1.0/28"
    },
    "VpcPeeringConnectionId": "pcx-1a2b3c4d",
    "RequesterVpcInfo": {
      "OwnerId": "444455556666",
      "VpcId": "vpc-111abc45",
      "CidrBlock": "10.0.0.0/28"
    }
  }
}
```

- Para API obtener más información, consulte [AcceptVpcPeeringConnection](#) la Referencia de AWS CLI comandos.

## advertise-byoip-cidr

En el siguiente ejemplo de código, se muestra cómo usar `advertise-byoip-cidr`.

### AWS CLI

Para anunciar un rango de direcciones

En el siguiente `advertise-byoip-cidr` ejemplo, se anuncia el rango de IPv4 direcciones públicas especificado.

```
aws ec2 advertise-byoip-cidr \
  --cidr 203.0.113.25/24
```

Salida:

```
{
```

```
"ByoipCidr": {
  "Cidr": "203.0.113.25/24",
  "StatusMessage": "ipv4pool-ec2-1234567890abcdef0",
  "State": "provisioned"
}
```

- Para API obtener más información, consulte [AdvertiseByoipCidr](#) la Referencia de AWS CLI comandos.

## allocate-address

En el siguiente ejemplo de código, se muestra cómo usar `allocate-address`.

### AWS CLI

Ejemplo 1: Asignar una dirección IP elástica del grupo de direcciones de Amazon

En el siguiente ejemplo de `allocate-address`, se asigna una dirección IP elástica. Amazon EC2 selecciona la dirección del conjunto de direcciones de Amazon.

```
aws ec2 allocate-address
```

Salida:

```
{
  "PublicIp": "70.224.234.241",
  "AllocationId": "eipalloc-01435ba59eEXAMPLE",
  "PublicIpv4Pool": "amazon",
  "NetworkBorderGroup": "us-west-2",
  "Domain": "vpc"
}
```

Para obtener más información, consulte [Direcciones IP elásticas](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Asignar una dirección IP elástica y asociarla a un grupo fronterizo de red

En el siguiente ejemplo de `allocate-address`, se asigna una dirección IP elástica y se la asocia al grupo fronterizo de red especificado.

```
aws ec2 allocate-address \  
  --network-border-group us-west-2-lax-1
```

Salida:

```
{  
  "PublicIp": "70.224.234.241",  
  "AllocationId": "eipalloc-e03dd489ceEXAMPLE",  
  "PublicIpv4Pool": "amazon",  
  "NetworkBorderGroup": "us-west-2-lax-1",  
  "Domain": "vpc"  
}
```

Para obtener más información, consulte [Direcciones IP elásticas](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 3: Asignar una dirección IP elástica desde un grupo de direcciones propio

En el siguiente ejemplo de `allocate-address`, se asigna una dirección IP elástica desde un grupo de direcciones que usted trajo a su cuenta de Amazon Web Services. Amazon EC2 selecciona la dirección del conjunto de direcciones.

```
aws ec2 allocate-address \  
  --public-ipv4-pool ipv4pool-ec2-1234567890abcdef0
```

Salida:

```
{  
  "AllocationId": "eipalloc-02463d08ceEXAMPLE",  
  "NetworkBorderGroup": "us-west-2",  
  "CustomerOwnedIp": "18.218.95.81",  
  "CustomerOwnedIpv4Pool": "ipv4pool-ec2-1234567890abcdef0",  
  "Domain": "vpc"  
  "NetworkBorderGroup": "us-west-2",  
}
```

Para obtener más información, consulte [Direcciones IP elásticas](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [AllocateAddress](#) la Referencia de AWS CLI comandos.

## allocate-hosts

En el siguiente ejemplo de código, se muestra cómo usar `allocate-hosts`.

### AWS CLI

Ejemplo 1: Para asignar un host dedicado

En el siguiente `allocate-hosts` ejemplo, se asigna un único host dedicado en la zona de `eu-west-1a` disponibilidad, en el que se pueden lanzar `m5.large` instancias. De forma predeterminada, el host dedicado solo acepta el lanzamiento de instancias de destino y no admite la recuperación del host.

```
aws ec2 allocate-hosts \  
  --instance-type m5.large \  
  --availability-zone eu-west-1a \  
  --quantity 1
```

Salida:

```
{  
  "HostIds": [  
    "h-07879acf49EXAMPLE"  
  ]  
}
```

Ejemplo 2: Para asignar un host dedicado con la ubicación automática y la recuperación del host habilitadas

El siguiente `allocate-hosts` ejemplo asigna un único host dedicado en la zona de `eu-west-1a` disponibilidad con la ubicación automática y la recuperación del host habilitadas.

```
aws ec2 allocate-hosts \  
  --instance-type m5.large \  
  --availability-zone eu-west-1a \  
  --auto-placement on \  
  --host-recovery on \  
  --quantity 1
```

Salida:

```
{
```

```

    "HostIds": [
      "h-07879acf49EXAMPLE"
    ]
  }

```

Ejemplo 3: Para asignar un host dedicado con etiquetas

El siguiente `allocate-hosts` ejemplo asigna un único host dedicado y aplica una etiqueta con una clave denominada `purpose` y un valor de `production`

```

aws ec2 allocate-hosts \
  --instance-type m5.large \
  --availability-zone eu-west-1a \
  --quantity 1 \
  --tag-specifications 'ResourceType=dedicated-
host,Tags={Key=purpose,Value=production}'

```

Salida:

```

{
  "HostIds": [
    "h-07879acf49EXAMPLE"
  ]
}

```

Para obtener más información, consulte [Asignación de hosts dedicados](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [AllocateHosts](#) la Referencia de AWS CLI comandos.

## **allocate-ipam-pool-cidr**

En el siguiente ejemplo de código, se muestra cómo usar `allocate-ipam-pool-cidr`.

### AWS CLI

Para asignar un elemento CIDR de un IPAM grupo

En el siguiente `allocate-ipam-pool-cidr` ejemplo, se asigna un elemento CIDR de un IPAM grupo.



(Linux):

```
aws ec2 allocate-ipam-pool-cidr \  
  --ipam-pool-id ipam-pool-0533048da7d823723 \  
  --netmask-length 24
```

(Windows):

```
aws ec2 allocate-ipam-pool-cidr ^  
  --ipam-pool-id ipam-pool-0533048da7d823723 ^  
  --netmask-length 24
```

Salida:

```
{  
  "IpamPoolAllocation": {  
    "Cidr": "10.0.0.0/24",  
    "IpamPoolAllocationId": "ipam-pool-alloc-018ecc28043b54ba38e2cd99943cebfbfd",  
    "ResourceType": "custom",  
    "ResourceOwner": "123456789012"  
  }  
}
```

Para obtener más información, consulta [Asignar manualmente un CIDR a un grupo para reservar espacio de direcciones IP](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [AllocateIpamPoolCidr](#) la Referencia de AWS CLI comandos.

## **apply-security-groups-to-client-vpn-target-network**

En el siguiente ejemplo de código, se muestra cómo usar `apply-security-groups-to-client-vpn-target-network`.

### AWS CLI

Para aplicar grupos de seguridad a una red de destino para un VPN punto final del cliente

El siguiente `apply-security-groups-to-client-vpn-target-network` ejemplo aplica un grupo de seguridad `sg-01f6e627a89f4db32` a la asociación entre la red de destino especificada y el VPN punto final del cliente.

```
aws ec2 apply-security-groups-to-client-vpn-target-network \  
  --security-group-ids sg-01f6e627a89f4db32 \  
  --vpc-id vpc-0e2110c2f324332e0 \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Salida:

```
{  
  "SecurityGroupIds": [  
    "sg-01f6e627a89f4db32"  
  ]  
}
```

Para obtener más información, consulte [Target Networks](#) en la Guía del VPN administrador de AWS clientes.

- Para API obtener más información, consulte [ApplySecurityGroupsToClientVpnTargetNetwork](#) la Referencia de AWS CLI comandos.

## assign-ipv6-addresses

En el siguiente ejemplo de código, se muestra cómo usar `assign-ipv6-addresses`.

### AWS CLI

Para asignar IPv6 direcciones específicas a una interfaz de red

En este ejemplo, se asignan las IPv6 direcciones especificadas a la interfaz de red especificada.

Comando:

```
aws ec2 assign-ipv6-addresses --network-interface-id eni-38664473 --ipv6-  
addresses 2001:db8:1234:1a00:3304:8879:34cf:4071 2001:db8:1234:1a00:9691:9503:25ad:1761
```

Salida:

```
{  
  "AssignedIpv6Addresses": [  
    "2001:db8:1234:1a00:3304:8879:34cf:4071",  
    "2001:db8:1234:1a00:9691:9503:25ad:1761"  
  ]  
}
```

```
],  
  "NetworkInterfaceId": "eni-38664473"  
}
```

Para asignar IPv6 las direcciones que Amazon selecciona a una interfaz de red

En este ejemplo se asignan dos IPv6 direcciones a la interfaz de red especificada. Amazon asigna automáticamente estas IPv6 direcciones a partir de las IPv6 direcciones disponibles en el rango de IPv6 CIDR bloques de la subred.

Comando:

```
aws ec2 assign-ipv6-addresses --network-interface-id eni-38664473 --ipv6-address-  
count 2
```

Salida:

```
{  
  "AssignedIpv6Addresses": [  
    "2001:db8:1234:1a00:3304:8879:34cf:4071",  
    "2001:db8:1234:1a00:9691:9503:25ad:1761"  
  ],  
  "NetworkInterfaceId": "eni-38664473"  
}
```

- Para API obtener más información, consulte [AssignIpv6 direcciones](#) en AWS CLI la referencia de comandos.

## assign-private-ip-addresses

En el siguiente ejemplo de código, se muestra cómo usar `assign-private-ip-addresses`.

AWS CLI

Para asignar una dirección IP privada secundaria específica a una interfaz de red

En este ejemplo, se asigna la dirección IP privada secundaria especificada a la interfaz de red especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 assign-private-ip-addresses --network-interface-id eni-e5aa89a3 --private-  
ip-addresses 10.0.0.82
```

Para asignar direcciones IP privadas secundarias que Amazon EC2 seleccione a una interfaz de red

En este ejemplo, se asignan dos direcciones IP privadas secundarias a la interfaz de red especificada. Amazon asigna EC2 automáticamente estas direcciones IP a partir de las direcciones IP disponibles en el rango de CIDR bloques de la subred a la que está asociada la interfaz de red. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 assign-private-ip-addresses --network-interface-id eni-e5aa89a3 --secondary-  
private-ip-address-count 2
```

- Para API obtener más información, consulte la Referencia [AssignPrivateIpAddresses](#) de AWS CLI comandos.

## assign-private-nat-gateway-address

En el siguiente ejemplo de código, se muestra cómo usar `assign-private-nat-gateway-address`.

### AWS CLI

Para asignar direcciones IP privadas a su NAT puerta de enlace privada

El siguiente `assign-private-nat-gateway-address` ejemplo asigna dos direcciones IP privadas a la NAT puerta de enlace privada especificada.

```
aws ec2 assign-private-nat-gateway-address \  
--nat-gateway-id nat-1234567890abcdef0 \  
--private-ip-address-count 2
```

Salida:

```
{  
  "NatGatewayId": "nat-1234567890abcdef0",  
  "NatGatewayAddresses": [  

```

```
    {
      "NetworkInterfaceId": "eni-0065a61b324d1897a",
      "IsPrimary": false,
      "Status": "assigning"
    },
    {
      "NetworkInterfaceId": "eni-0065a61b324d1897a",
      "IsPrimary": false,
      "Status": "assigning"
    }
  ]
}
```

Para obtener más información, consulta [NATlas pasarelas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [AssignPrivateNatGatewayAddress](#) la Referencia de AWS CLI comandos.

## associate-address

En el siguiente ejemplo de código, se muestra cómo usar `associate-address`.

### AWS CLI

Para asociar una dirección IP elástica en EC2 -Classic

En este ejemplo, se asocia una dirección IP elástica a una instancia de EC2 -Classic. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 associate-address --instance-id i-07ffe74c7330ebf53 --public-ip 198.51.100.0
```

Para asociar una dirección IP elástica en - EC2 VPC

En este ejemplo, se asocia una dirección IP elástica a una instancia de unVPC.

Comando:

```
aws ec2 associate-address --instance-id i-0b263919b6498b123 --allocation-id eipalloc-64d5890a
```

Salida:

```
{
  "AssociationId": "eipassoc-2bebb745"
}
```

En este ejemplo, se asocia una dirección IP elástica a una interfaz de red.

Comando:

```
aws ec2 associate-address --allocation-id eipalloc-64d5890a --network-interface-id eni-1a2b3c4d
```

En este ejemplo, se asocia una dirección IP elástica a una dirección IP privada asociada a una interfaz de red.

Comando:

```
aws ec2 associate-address --allocation-id eipalloc-64d5890a --network-interface-id eni-1a2b3c4d --private-ip-address 10.0.0.85
```

- Para API obtener más información, consulte [AssociateAddress](#) la Referencia de AWS CLI comandos.

## **associate-client-vpn-target-network**

En el siguiente ejemplo de código, se muestra cómo usar `associate-client-vpn-target-network`.

AWS CLI

Para asociar una red de destino a un VPN punto final del cliente

El siguiente `associate-client-vpn-target-network` ejemplo asocia una subred al VPN punto final del cliente especificado.

```
aws ec2 associate-client-vpn-target-network \
  --subnet-id subnet-0123456789abcabca \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Salida:

```
{
  "AssociationId": "cvpn-assoc-12312312312312312",
  "Status": {
    "Code": "associating"
  }
}
```

Para obtener más información, consulte [Target Networks](#) en la Guía del VPN administrador de AWS clientes.

- Para API obtener más información, consulte [AssociateClientVpnTargetNetwork](#) la Referencia de AWS CLI comandos.

## associate-dhcp-options

En el siguiente ejemplo de código, se muestra cómo usar `associate-dhcp-options`.

AWS CLI

Para asociar un conjunto DHCP de opciones a su VPC

En este ejemplo, se asocia el conjunto de DHCP opciones especificado al especificado VPC. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 associate-dhcp-options --dhcp-options-id dopt-d9070ebb --vpc-id vpc-a01106c2
```

Para asociar el conjunto de DHCP opciones por defecto a su VPC

En este ejemplo, se asocia el conjunto de DHCP opciones por defecto al especificado VPC. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 associate-dhcp-options --dhcp-options-id default --vpc-id vpc-a01106c2
```

- Para API obtener más información, consulte [AssociateDhcpOptions](#) la Referencia de AWS CLI comandos.

## associate-iam-instance-profile

En el siguiente ejemplo de código, se muestra cómo usar `associate-iam-instance-profile`.

### AWS CLI

Para asociar un perfil de IAM instancia a una instancia

En este ejemplo, se asocia un perfil de IAM instancia denominado `admin-role` a la instancia `i-123456789abcde123`.

Comando:

```
aws ec2 associate-iam-instance-profile --instance-id i-123456789abcde123 --iam-instance-profile Name=admin-role
```

Salida:

```
{
  "IamInstanceProfileAssociation": {
    "InstanceId": "i-123456789abcde123",
    "State": "associating",
    "AssociationId": "iip-assoc-0e7736511a163c209",
    "IamInstanceProfile": {
      "Id": "AIPAJBLK7RKJKWDXVHIEC",
      "Arn": "arn:aws:iam::123456789012:instance-profile/admin-role"
    }
  }
}
```

- Para API obtener más información, consulte [AssociateIamInstanceProfile](#) la Referencia de AWS CLI comandos.

## associate-instance-event-window

En el siguiente ejemplo de código, se muestra cómo usar `associate-instance-event-window`.

### AWS CLI

Ejemplo 1: Para asociar una o más instancias a una ventana de eventos



El siguiente `associate-instance-event-window` ejemplo asocia una o más instancias a una ventana de eventos.

```
aws ec2 associate-instance-event-window \  
  --region us-east-1 \  
  --instance-event-window-id iew-0abcdef1234567890 \  
  --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

Salida:

```
{  
  "InstanceEventWindow": {  
    "InstanceEventWindowId": "iew-0abcdef1234567890",  
    "Name": "myEventWindowName",  
    "CronExpression": "* 21-23 * * 2,3",  
    "AssociationTarget": {  
      "InstanceIds": [  
        "i-1234567890abcdef0",  
        "i-0598c7d356eba48d7"  
      ],  
      "Tags": [],  
      "DedicatedHostIds": []  
    },  
    "State": "creating"  
  }  
}
```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para asociar etiquetas de instancia a una ventana de eventos

En el siguiente `associate-instance-event-window` ejemplo, se asocian etiquetas de instancia a una ventana de eventos. Introduzca un `instance-event-window-id` parámetro para especificar la ventana de eventos. Para asociar etiquetas de instancia, especifique el `association-target` parámetro y, para el valor del parámetro, especifique una o más etiquetas.

```
aws ec2 associate-instance-event-window \  
  --region us-east-1 \  
  --instance-event-window-id iew-0abcdef1234567890 \  
  --association-target "Tags=tag:Name=example-tag"
```

```
--association-target "InstanceTags=[{Key=k2, Value=v2}, {Key=k1, Value=v1}]"
```

Salida:

```
{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [],
      "Tags": [
        {
          "Key": "k2",
          "Value": "v2"
        },
        {
          "Key": "k1",
          "Value": "v1"
        }
      ],
      "DedicatedHostIds": []
    },
    "State": "creating"
  }
}
```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

Ejemplo 3: Para asociar un anfitrión dedicado a una ventana de eventos

El siguiente `associate-instance-event-window` ejemplo asocia un host dedicado a una ventana de eventos. Introduzca un `instance-event-window-id` parámetro para especificar la ventana de eventos. Para asociar un host dedicado, especifique el `--association-target` parámetro y, para los valores de los parámetros, especifique uno o más hosts dedicadosIDs.

```
aws ec2 associate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target "DedicatedHostIds=h-029fa35a02b99801d"
```

Salida:

```
{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [],
      "Tags": [],
      "DedicatedHostIds": [
        "h-029fa35a02b99801d"
      ]
    },
    "State": "creating"
  }
}
```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [AssociateInstanceEventWindow](#) la Referencia de AWS CLI comandos.

## **associate-ipam-resource-discovery**

En el siguiente ejemplo de código, se muestra cómo usar `associate-ipam-resource-discovery`.

### AWS CLI

Para asociar un descubrimiento de recursos con un IPAM

En este ejemplo, usted es un administrador IPAM delegado y otra AWS cuenta ha creado y compartido con usted una función de descubrimiento de recursos para que pueda utilizarla IPAM para gestionar y supervisar los recursos que CIDRs son propiedad de la otra cuenta.

### Nota

Para completar esta solicitud, necesitará el ID de descubrimiento de recursos que puede obtener [describe-ipam-resource-discoveries](#) y el IPAM ID que puede obtener con [describe-ipams](#). El descubrimiento de recursos que está asociando debe haberse compartido primero con su cuenta

mediante AWS RAM. El que introduzca `--region` debe coincidir con la región de origen a la IPAM que lo está asociando.

El siguiente `associate-ipam-resource-discovery` ejemplo asocia un descubrimiento de recursos con un IPAM

```
aws ec2 associate-ipam-resource-discovery \
  --ipam-id ipam-005f921c17ebd5107 \
  --ipam-resource-discovery-id ipam-res-disco-03e0406de76a044ee \
  --tag-specifications 'ResourceType=ipam-resource-discovery,Tags=[{Key=cost-center,Value=cc123}]' \
  --region us-east-1
```

Salida:

```
{
  {
    "IpamResourceDiscoveryAssociation": {
      "OwnerId": "320805250157",
      "IpamResourceDiscoveryAssociationId": "ipam-res-disco-
assoc-04382a6346357cf82",
      "IpamResourceDiscoveryAssociationArn": "arn:aws:ec2::320805250157:ipam-
resource-discovery-association/ipam-res-disco-assoc-04382a6346357cf82",
      "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
      "IpamId": "ipam-005f921c17ebd5107",
      "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",
      "IpamRegion": "us-east-1",
      "IsDefault": false,
      "ResourceDiscoveryStatus": "active",
      "State": "associate-in-progress",
      "Tags": []
    }
  }
}
```

Una vez que asocie una detección de recursos, puede supervisar o administrar las direcciones IP de los recursos creados por las demás cuentas. Para obtener más información, consulta la [sección Integración IPAM con cuentas ajenas a tu organización](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [AssociateIpamResourceDiscovery](#) la Referencia de AWS CLI comandos.

## associate-nat-gateway-address

En el siguiente ejemplo de código, se muestra cómo usar `associate-nat-gateway-address`.

### AWS CLI

Para asociar una dirección IP elástica a una NAT puerta de enlace pública

El siguiente `associate-nat-gateway-address` ejemplo asocia la dirección IP elástica especificada con la NAT puerta de enlace pública especificada. AWS asigna automáticamente una IPv4 dirección privada secundaria.

```
aws ec2 associate-nat-gateway-address \
  --nat-gateway-id nat-1234567890abcdef0 \
  --allocation-ids eipalloc-0be6ecac95EXAMPLE
```

Salida:

```
{
  "NatGatewayId": "nat-1234567890abcdef0",
  "NatGatewayAddresses": [
    {
      "AllocationId": "eipalloc-0be6ecac95EXAMPLE",
      "NetworkInterfaceId": "eni-09cc4b2558794f7f9",
      "IsPrimary": false,
      "Status": "associating"
    }
  ]
}
```

Para obtener más información, consulta [NATlas pasarelas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [AssociateNatGatewayAddress](#) la Referencia de AWS CLI comandos.

## associate-route-table

En el siguiente ejemplo de código, se muestra cómo usar `associate-route-table`.

## AWS CLI

Para asociar una tabla de rutas a una subred

En este ejemplo, se asocia la tabla de rutas especificada a la subred especificada.

Comando:

```
aws ec2 associate-route-table --route-table-id rtb-22574640 --subnet-id subnet-9d4a7b6c
```

Salida:

```
{
  "AssociationId": "rtbassoc-781d0d1a"
}
```

- Para API obtener más información, consulte [AssociateRouteTable](#) la Referencia de AWS CLI comandos.

## associate-subnet-cidr-block

En el siguiente ejemplo de código, se muestra cómo usar `associate-subnet-cidr-block`.

## AWS CLI

Para asociar un IPv6 CIDR bloque a una subred

En este ejemplo, se asocia un IPv6 CIDR bloque a la subred especificada.

Comando:

```
aws ec2 associate-subnet-cidr-block --subnet-id subnet-5f46ec3b --ipv6-cidr-block 2001:db8:1234:1a00::/64
```

Salida:

```
{
  "SubnetId": "subnet-5f46ec3b",
  "Ipv6CidrBlockAssociation": {
    "Ipv6CidrBlock": "2001:db8:1234:1a00::/64",
    "AssociationId": "subnet-cidr-assoc-3aa54053",
  }
}
```

```

    "Ipv6CidrBlockState": {
      "State": "associating"
    }
  }
}

```

- Para API obtener más información, consulte [AssociateSubnetCidrBlock](#) la Referencia de AWS CLI comandos.

## associate-transit-gateway-multicast-domain

En el siguiente ejemplo de código, se muestra cómo usar `associate-transit-gateway-multicast-domain`.

### AWS CLI

Para asociar una puerta de enlace de tránsito a un dominio de multidifusión

El siguiente `associate-transit-gateway-multicast-domain` ejemplo asocia la subred y el adjunto especificados al dominio de multidifusión especificado.

```

aws ec2 associate-transit-gateway-multicast-domain \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
  --transit-gateway-attachment-id tgw-attach-028c1dd0f8f5cbe8e \
  --subnet-ids subnet-000de86e3b49c932a \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE

```

Salida:

```

{
  "Associations": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
    "TransitGatewayAttachmentId": "tgw-attach-028c1dd0f8f5cbe8e",
    "ResourceId": "vpc-01128d2c240c09bd5",
    "ResourceType": "vpc",
    "Subnets": [
      {
        "SubnetId": "subnet-000de86e3b49c932a",
        "State": "associating"
      }
    ]
  }
}

```

```
}
```

Para obtener más información, consulte [Administración de dominios de multidifusión](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [AssociateTransitGatewayMulticastDomain](#) de AWS CLI comandos.

## **associate-transit-gateway-route-table**

En el siguiente ejemplo de código, se muestra cómo usar `associate-transit-gateway-route-table`.

### AWS CLI

Para asociar una tabla de rutas de una puerta de enlace de tránsito con un adjunto de una puerta de enlace de tránsito

El siguiente ejemplo asocia la tabla de rutas de la puerta de enlace de tránsito especificada con el VPC adjunto especificado.

```
aws ec2 associate-transit-gateway-route-table \  
  --transit-gateway-route-table-id tgw-rtb-002573ed1eEXAMPLE \  
  --transit-gateway-attachment-id tgw-attach-0b5968d3b6EXAMPLE
```

Salida:

```
{  
  "Association": {  
    "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",  
    "TransitGatewayAttachmentId": "tgw-attach-0b5968d3b6EXAMPLE",  
    "ResourceId": "vpc-0065acced4EXAMPLE",  
    "ResourceType": "vpc",  
    "State": "associating"  
  }  
}
```

Para obtener más información, consulte [Asociar una tabla de rutas de Transit Gateway](#) en la Guía de AWS Transit Gateways.

- Para API obtener más información, consulte [AssociateTransitGatewayRouteTable](#) la Referencia de AWS CLI comandos.



## associate-vpc-cidr-block

En el siguiente ejemplo de código, se muestra cómo usar `associate-vpc-cidr-block`.

### AWS CLI

Ejemplo 1: Para asociar un IPv6 CIDR bloque proporcionado por Amazon a un VPC

El siguiente `associate-vpc-cidr-block` ejemplo asocia un IPv6 CIDR bloque al especificado. VPC :

```
aws ec2 associate-vpc-cidr-block \  
  --amazon-provided-ipv6-cidr-block \  
  --ipv6-cidr-block-network-border-group us-west-2-lax-1 \  
  --vpc-id vpc-8EXAMPLE
```

Salida:

```
{  
  "Ipv6CidrBlockAssociation": {  
    "AssociationId": "vpc-cidr-assoc-0838ce7d9dEXAMPLE",  
    "Ipv6CidrBlockState": {  
      "State": "associating"  
    },  
    "NetworkBorderGroup": "us-west-2-lax-1"  
  },  
  "VpcId": "vpc-8EXAMPLE"  
}
```

Ejemplo 2: Para asociar un IPv4 CIDR bloque adicional a un VPC

En el siguiente `associate-vpc-cidr-block` ejemplo, se asocia el IPv4 CIDR bloque `10.2.0.0/16` al especificado VPC.

```
aws ec2 associate-vpc-cidr-block \  
  --vpc-id vpc-1EXAMPLE \  
  --cidr-block 10.2.0.0/16
```

Salida:

```
{
```

```
"CidrBlockAssociation": {
  "AssociationId": "vpc-cidr-assoc-2EXAMPLE",
  "CidrBlock": "10.2.0.0/16",
  "CidrBlockState": {
    "State": "associating"
  }
},
"VpcId": "vpc-1EXAMPLE"
}
```

- Para API obtener más información, consulte [AssociateVpcCidrBlock](#) la Referencia de AWS CLI comandos.

## attach-classic-link-vpc

En el siguiente ejemplo de código, se muestra cómo usar `attach-classic-link-vpc`.

### AWS CLI

Para vincular (adjuntar) una instancia EC2 de -Classic a un VPC

En este ejemplo, se vincula la instancia `i-1234567890abcdef0` con la `vpc-88888888` a través del grupo de seguridad `sg-12312312`. VPC VPC

Comando:

```
aws ec2 attach-classic-link-vpc --instance-id i-1234567890abcdef0 --vpc-id vpc-88888888 --groups sg-12312312
```

Salida:

```
{
  "Return": true
}
```

- API Para obtener [AttachClassicLinkVpc](#) más información, consulte AWS CLI la Referencia de comandos.

## attach-internet-gateway

En el siguiente ejemplo de código, se muestra cómo usar `attach-internet-gateway`.

## AWS CLI

Para adjuntar una puerta de enlace de Internet a su VPC

En el siguiente `attach-internet-gateway` ejemplo, se adjunta la puerta de enlace de Internet especificada a la específica VPC.

```
aws ec2 attach-internet-gateway \  
  --internet-gateway-id igw-0d0fb496b3EXAMPLE \  
  --vpc-id vpc-0a60eb65b4EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [las pasarelas de Internet](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [AttachInternetGateway](#) la Referencia de AWS CLI comandos.

## `attach-network-interface`

En el siguiente ejemplo de código, se muestra cómo usar `attach-network-interface`.

### AWS CLI

Ejemplo 1: Para adjuntar una interfaz de red a una instancia

En el siguiente `attach-network-interface` ejemplo, se adjunta la interfaz de red especificada a la instancia especificada.

```
aws ec2 attach-network-interface \  
  --network-interface-id eni-0dc56a8d4640ad10a \  
  --instance-id i-1234567890abcdef0 \  
  --device-index 1
```

Salida:

```
{  
  "AttachmentId": "eni-attach-01a8fc87363f07cf9"  
}
```

Para obtener más información, consulte [Interfaces de red elásticas](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para conectar una interfaz de red a una instancia con varias tarjetas de red

En el siguiente `attach-network-interface` ejemplo, se conecta la interfaz de red especificada a la instancia y a la tarjeta de red especificadas.

```
aws ec2 attach-network-interface \  
  --network-interface-id eni-07483b1897541ad83 \  
  --instance-id i-01234567890abcdef \  
  --network-card-index 1 \  
  --device-index 1
```

Salida:

```
{  
  "AttachmentId": "eni-attach-0fbd7ee87a88cd06c"  
}
```

Para obtener más información, consulte [Interfaces de red elásticas](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [AttachNetworkInterface](#) la Referencia de AWS CLI comandos.

## **attach-verified-access-trust-provider**

En el siguiente ejemplo de código, se muestra cómo usar `attach-verified-access-trust-provider`.

AWS CLI

Para adjuntar un proveedor de confianza a una instancia

En el siguiente `attach-verified-access-trust-provider` ejemplo, se adjunta el proveedor de confianza de acceso verificado especificado a la instancia de acceso verificado especificada.

```
aws ec2 attach-verified-access-trust-provider \  
  --verified-access-instance-id vai-0ce000c0b7643abea \  
  --verified-access-trust-provider-id vatp-0c000000000000000000000000000000
```

```
--verified-access-trust-provider-id vatp-0bb32de759a3e19e7
```

Salida:

```
{
  "VerifiedAccessTrustProvider": {
    "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
    "Description": "",
    "TrustProviderType": "user",
    "UserTrustProviderType": "iam-identity-center",
    "PolicyReferenceName": "idc",
    "CreationTime": "2023-08-25T19:00:38",
    "LastUpdatedTime": "2023-08-25T19:00:38"
  },
  "VerifiedAccessInstance": {
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "Description": "",
    "VerifiedAccessTrustProviders": [
      {
        "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
        "TrustProviderType": "user",
        "UserTrustProviderType": "iam-identity-center"
      }
    ],
    "CreationTime": "2023-08-25T18:27:56",
    "LastUpdatedTime": "2023-08-25T18:27:56"
  }
}
```

Para obtener más información, consulte las [instancias de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [AttachVerifiedAccessTrustProvider](#) la Referencia de AWS CLI comandos.

## attach-volume

En el siguiente ejemplo de código, se muestra cómo usar attach-volume.

### AWS CLI

Para adjuntar un volumen a una instancia

Este comando de ejemplo adjunta un volumen (`vol-1234567890abcdef0`) a una instancia (`i-01474ef662b89480`) como `/dev/sdf`.

Comando:

```
aws ec2 attach-volume --volume-id vol-1234567890abcdef0 --instance-id i-01474ef662b89480 --device /dev/sdf
```

Salida:

```
{
  "AttachTime": "YYYY-MM-DDTHH:MM:SS.000Z",
  "InstanceId": "i-01474ef662b89480",
  "VolumeId": "vol-1234567890abcdef0",
  "State": "attaching",
  "Device": "/dev/sdf"
}
```

- Para API obtener más información, consulte [AttachVolume](#) la Referencia de AWS CLI comandos.

## attach-vpn-gateway

En el siguiente ejemplo de código, se muestra cómo usar `attach-vpn-gateway`.

AWS CLI

Para adjuntar una puerta de enlace privada virtual a su VPC

En el siguiente `attach-vpn-gateway` ejemplo, se adjunta la puerta de enlace privada virtual especificada a la especificada VPC.

```
aws ec2 attach-vpn-gateway \  
  --vpn-gateway-id vgw-9a4cacf3 \  
  --vpc-id vpc-a01106c2
```

Salida:

```
{
  "VpcAttachment": {
```

```

    "State": "attaching",
    "VpcId": "vpc-a01106c2"
  }
}

```

- Para API obtener más información, consulte [AttachVpnGateway](#) la Referencia de AWS CLI comandos.

## authorize-client-vpn-ingress

En el siguiente ejemplo de código, se muestra cómo usar `authorize-client-vpn-ingress`.

### AWS CLI

Para añadir una regla de autorización para un VPN punto final del cliente

El siguiente `authorize-client-vpn-ingress` ejemplo agrega una regla de autorización de entrada que permite a todos los clientes acceder a Internet (`0.0.0.0/0`).

```

aws ec2 authorize-client-vpn-ingress \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
  --target-network-cidr 0.0.0.0/0 \
  --authorize-all-groups

```

Salida:

```

{
  "Status": {
    "Code": "authorizing"
  }
}

```

Para obtener más información, consulte [las reglas de autorización](#) en la Guía del VPN administrador de AWS clientes.

- Para API obtener más información, consulte [AuthorizeClientVpnIngress](#) la Referencia de AWS CLI comandos.

## authorize-security-group-egress

En el siguiente ejemplo de código, se muestra cómo usar `authorize-security-group-egress`.

## AWS CLI

Para agregar una regla que permita el tráfico saliente a un rango de direcciones específico

Este comando de ejemplo agrega una regla que otorga acceso a los rangos de direcciones especificados en el TCP puerto 80.

Comando (Linux):

```
aws ec2 authorize-security-group-egress --group-id sg-1a2b3c4d --ip-permissions IpProtocol=tcp,FromPort=80,ToPort=80,IpRanges='[{{CidrIp=10.0.0.0/16}}]'
```

Comando (Windows):

```
aws ec2 authorize-security-group-egress --group-id sg-1a2b3c4d --ip-permissions IpProtocol=tcp,FromPort=80,ToPort=80,IpRanges=[{{CidrIp=10.0.0.0/16}}]
```

Para agregar una regla que permita el tráfico saliente a un grupo de seguridad específico

Este comando de ejemplo agrega una regla que otorga acceso al grupo de seguridad especificado en el TCP puerto 80.

Comando (Linux):

```
aws ec2 authorize-security-group-egress --group-id sg-1a2b3c4d --ip-permissions IpProtocol=tcp,FromPort=80,ToPort=80,UserIdGroupPairs='[{{GroupId=sg-4b51a32f}}]'
```

Comando (Windows):

```
aws ec2 authorize-security-group-egress --group-id sg-1a2b3c4d --ip-permissions IpProtocol=tcp,FromPort=80,ToPort=80,UserIdGroupPairs=[{{GroupId=sg-4b51a32f}}]
```

- Para API obtener más información, consulte [AuthorizeSecurityGroupEgress](#) la Referencia de AWS CLI comandos.

## authorize-security-group-ingress

En el siguiente ejemplo de código, se muestra cómo usar `authorize-security-group-ingress`.



## AWS CLI

Ejemplo 1: Para añadir una regla que permita el tráfico entrante SSH

El siguiente `authorize-security-group-ingress` ejemplo agrega una regla que permite el tráfico entrante en el TCP puerto 22 ()SSH.

```
aws ec2 authorize-security-group-ingress \  
  --group-id sg-1234567890abcdef0 \  
  --protocol tcp \  
  --port 22 \  
  --cidr 203.0.113.0/24
```

Salida:

```
{  
  "Return": true,  
  "SecurityGroupRules": [  
    {  
      "SecurityGroupRuleId": "sgr-01afa97ef3e1bedfc",  
      "GroupId": "sg-1234567890abcdef0",  
      "GroupOwnerId": "123456789012",  
      "IsEgress": false,  
      "IpProtocol": "tcp",  
      "FromPort": 22,  
      "ToPort": 22,  
      "CidrIpv4": "203.0.113.0/24"  
    }  
  ]  
}
```

Ejemplo 2: Para agregar una regla que permita el HTTP tráfico entrante desde otro grupo de seguridad

El siguiente `authorize-security-group-ingress` ejemplo agrega una regla que permite el acceso entrante al TCP puerto 80 desde el grupo de seguridad de origen. `sg-1a2b3c4d` El grupo de origen debe estar en el mismo grupo VPC o en un grupo homólogo VPC (requiere una conexión de VPC emparejamiento). Se permite el tráfico entrante según las direcciones IP privadas de las instancias asociadas al grupo de seguridad de origen (y no la dirección IP pública o la dirección IP elástica).

```
aws ec2 authorize-security-group-ingress \  
  --group-id sg-1a2b3c4d \  
  --group-owner-id 123456789012 \  
  --protocol tcp \  
  --port 80 \  
  --source-group sg-1a2b3c4d
```

```
--group-id sg-1234567890abcdef0 \  
--protocol tcp \  
--port 80 \  
--source-group sg-1a2b3c4d
```

Salida:

```
{  
  "Return": true,  
  "SecurityGroupRules": [  
    {  
      "SecurityGroupRuleId": "sgr-01f4be99110f638a7",  
      "GroupId": "sg-1234567890abcdef0",  
      "GroupOwnerId": "123456789012",  
      "IsEgress": false,  
      "IpProtocol": "tcp",  
      "FromPort": 80,  
      "ToPort": 80,  
      "ReferencedGroupInfo": {  
        "GroupId": "sg-1a2b3c4d",  
        "UserId": "123456789012"  
      }  
    }  
  ]  
}
```

Ejemplo 3: Agregar varias reglas en la misma llamada

En el siguiente `authorize-security-group-ingress` ejemplo, se usa el `ip-permissions` parámetro para agregar dos reglas de entrada, una que permite el acceso entrante en el TCP puerto 3389 (RDP) y otra que habilita el comando ping/. ICMP

```
aws ec2 authorize-security-group-ingress --group-id sg-1234567890abcdef0 --ip-permissions  
IpProtocol =tcp, FromPort =3389, ToPort =3389, IpRanges = "[{CidrIp=172.31.0.0/16}]" IpProtocol  
=icmp, FromPort ToPort =-1, IpRanges = "[{CidrIp=172.31.0.0/16}]"
```

Salida:

```
{  
  "Return": true,  
  "SecurityGroupRules": [  
    {  
      "SecurityGroupRuleId": "sgr-00e06e5d3690f29f3",
```

```

    "GroupId": "sg-1234567890abcdef0",
    "GroupOwnerId": "123456789012",
    "IsEgress": false,
    "IpProtocol": "tcp",
    "FromPort": 3389,
    "ToPort": 3389,
    "CidrIpv4": "172.31.0.0/16"
  },
  {
    "SecurityGroupId": "sgr-0a133dd4493944b87",
    "GroupId": "sg-1234567890abcdef0",
    "GroupOwnerId": "123456789012",
    "IsEgress": false,
    "IpProtocol": "tcp",
    "FromPort": -1,
    "ToPort": -1,
    "CidrIpv4": "172.31.0.0/16"
  }
]
}

```

#### Ejemplo 4: Para agregar ICMP una regla de tráfico

En el siguiente `authorize-security-group-ingress` ejemplo, se utiliza el `ip-permissions` parámetro para agregar una regla de entrada que permita ICMP enviar mensajes `Destination Unreachable: Fragmentation Needed and Don't Fragment was Set` (tipo 3, código 4) desde cualquier lugar.

```
aws ec2 authorize-security-group-ingress --group-id sg-1234567890abcdef0 --ip-permissions =icmp, =3, =4, = [{"=0.0.0.0/0}] IpProtocol» FromPort ToPort IpRanges CidrIp
```

Salida:

```

{
  "Return": true,
  "SecurityGroupRules": [
    {
      "SecurityGroupId": "sgr-0de3811019069b787",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "123456789012",
      "IsEgress": false,
      "IpProtocol": "icmp",
      "FromPort": 3,

```

```

        "ToPort": 4,
        "CidrIpv4": "0.0.0.0/0"
    }
]
}

```

### Ejemplo 5: Para agregar una regla IPv6 de tráfico

En el siguiente `authorize-security-group-ingress` ejemplo, se utiliza el `ip-permissions` parámetro para agregar una regla de entrada que permita el SSH acceso (puerto 22) desde el IPv6 rango `2001:db8:1234:1a00::/64`.

```
aws ec2 authorize-security-group-ingress --group-id sg-1234567890abcdef0 --ip-permissions
IpProtocol=tcp,=22,=22,Ipv6Ranges=["{{6=2001:db8:1234:1a00::/64}}» FromPort ToPort
CidrIpv
```

Salida:

```

{
  "Return": true,
  "SecurityGroupRules": [
    {
      "SecurityGroupRuleId": "sgr-0455bc68b60805563",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "123456789012",
      "IsEgress": false,
      "IpProtocol": "tcp",
      "FromPort": 22,
      "ToPort": 22,
      "CidrIpv6": "2001:db8:1234:1a00::/64"
    }
  ]
}

```

### Ejemplo 6: Para añadir una ICMPv6 regla de tráfico

En el siguiente `authorize-security-group-ingress` ejemplo, se utiliza el `ip-permissions` parámetro para agregar una regla de entrada que permita el ICMPv6 tráfico desde cualquier lugar.

```
aws ec2 authorize-security-group-ingress --group-id sg-1234567890abcdef0 --ip-permissions
=icmpv6, Ipv6Ranges=["{{6=:/0}}» IpProtocol CidrIpv
```

**Salida:**

```
{
  "Return": true,
  "SecurityGroupRules": [
    {
      "SecurityGroupId": "sgr-04b612d9363ab6327",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "123456789012",
      "IsEgress": false,
      "IpProtocol": "icmpv6",
      "FromPort": -1,
      "ToPort": -1,
      "CidrIpv6": "::/0"
    }
  ]
}
```

**Ejemplo 7: Agregar una regla con una descripción**

En el `authorize-security-group-ingress` siguiente ejemplo, se usa el parámetro para agregar una regla `ip-permissions` de entrada que permita el tráfico desde el rango de direcciones especificado RDP. IPv4 La regla incluye una descripción que lo ayudará a identificarla posteriormente.

```
aws ec2 authorize-security-group-ingress --group-id sg-1234567890abcdef0 --ip-permissions
IpProtocol=tcp, =3389, =3389, = "[{=203.0.113.0/24, Description='acceso desde una oficina de
Nueva York'}FromPort]» ToPort IpRanges CidrIp RDP
```

**Salida:**

```
{
  "Return": true,
  "SecurityGroupRules": [
    {
      "SecurityGroupId": "sgr-0397bbcc01e974db3",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "123456789012",
      "IsEgress": false,
      "IpProtocol": "tcp",
      "FromPort": 3389,
      "ToPort": 3389,

```

```

        "CidrIpv4": "203.0.113.0/24",
        "Description": "RDP access from NY office"
    }
]
}

```

### Ejemplo 8: Agregar una regla de entrada que use una lista de prefijos

En `authorize-security-group-ingress` el siguiente ejemplo, se usa el parámetro `ip-permissions` para agregar una `ip-permissions` regla de entrada que permita todo el tráfico en los rangos de la lista CIDR de prefijos especificada.

```
aws ec2 authorize-security-group-ingress --group-id sg-04a351bfe432d4e71 --ip-permissions =all,
= [{"pl-002dc3ec097de1514}» IpProtocol PrefixListIds PrefixListId
```

Salida:

```

{
  "Return": true,
  "SecurityGroupRules": [
    {
      "SecurityGroupRuleId": "sgr-09c74b32f677c6c7c",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "123456789012",
      "IsEgress": false,
      "IpProtocol": "-1",
      "FromPort": -1,
      "ToPort": -1,
      "PrefixListId": "pl-0721453c7ac4ec009"
    }
  ]
}

```

Para obtener más información, consulte [Grupos de seguridad](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [AuthorizeSecurityGroupIngress](#) la Referencia de AWS CLI comandos.

## bundle-instance

En el siguiente ejemplo de código, se muestra cómo usar `bundle-instance`.

## AWS CLI

Para empaquetar una instancia

En este ejemplo, se agrupa `i-1234567890abcdef0` la instancia en un bucket llamado `bundletasks`. Antes de especificar valores para la clave de acceso IDs, revise y siga las instrucciones de las prácticas recomendadas para administrar las claves de AWS acceso.

Comando:

```
aws ec2 bundle-instance --instance-id i-1234567890abcdef0 --bucket bundletasks --  
prefix winami --owner-akid AK12AJEXAMPLE --owner-sak example123example
```

Salida:

```
{  
  "BundleTask": {  
    "UpdateTime": "2015-09-15T13:30:35.000Z",  
    "InstanceId": "i-1234567890abcdef0",  
    "Storage": {  
      "S3": {  
        "Prefix": "winami",  
        "Bucket": "bundletasks"  
      }  
    },  
    "State": "pending",  
    "StartTime": "2015-09-15T13:30:35.000Z",  
    "BundleId": "bun-294e041f"  
  }  
}
```

- Para API obtener más información, consulte [BundleInstance](#) la Referencia de AWS CLI comandos.

## cancel-bundle-task

En el siguiente ejemplo de código, se muestra cómo usar `cancel-bundle-task`.

## AWS CLI

Para cancelar una tarea agrupada

En este ejemplo se cancela la tarea de agrupamiento. `bun-2a4e041c`

Comando:

```
aws ec2 cancel-bundle-task --bundle-id bun-2a4e041c
```

Salida:

```
{
  "BundleTask": {
    "UpdateTime": "2015-09-15T13:27:40.000Z",
    "InstanceId": "i-1234567890abcdef0",
    "Storage": {
      "S3": {
        "Prefix": "winami",
        "Bucket": "bundletasks"
      }
    },
    "State": "cancelling",
    "StartTime": "2015-09-15T13:24:35.000Z",
    "BundleId": "bun-2a4e041c"
  }
}
```

- Para API obtener más información, consulte [CancelBundleTask](#) la Referencia de AWS CLI comandos.

## cancel-capacity-reservation-fleets

En el siguiente ejemplo de código, se muestra cómo usar `cancel-capacity-reservation-fleets`.

AWS CLI

Para cancelar una flota de reserva de capacidad

El siguiente `cancel-capacity-reservation-fleets` ejemplo cancela la flota de reserva de capacidad especificada y la capacidad que reserva. Al cancelar una flota, su estado cambia a `cancelled` y ya no puede crear nuevas reservas de capacidad. Además, se cancelan todas las reservas de capacidad individuales de la flota y las instancias que anteriormente se ejecutaban en la capacidad reservada seguirán funcionando normalmente en la capacidad compartida.



```
aws ec2 cancel-capacity-reservation-fleets \  
--capacity-reservation-fleet-ids crf-abcdef01234567890
```

Salida:

```
{  
  "SuccessfulFleetCancellations": [  
    {  
      "CurrentFleetState": "cancelling",  
      "PreviousFleetState": "active",  
      "CapacityReservationFleetId": "crf-abcdef01234567890"  
    }  
  ],  
  "FailedFleetCancellations": []  
}
```

Para obtener más información sobre las flotas de reserva de capacidad, consulta [Flotas de reserva de capacidad](#) en la Guía EC2 del usuario de Amazon.

- Para API obtener más información, consulte la Referencia [CancelCapacityReservationFleets](#) de AWS CLI comandos.

## cancel-capacity-reservation

En el siguiente ejemplo de código, se muestra cómo usar `cancel-capacity-reservation`.

AWS CLI

Para cancelar una reserva de capacidad

El siguiente `cancel-capacity-reservation` ejemplo cancela la reserva de capacidad especificada.

```
aws ec2 cancel-capacity-reservation \  
--capacity-reservation-id cr-1234abcd56EXAMPLE
```

Salida:

```
{  
  "Return": true  
}
```

Para obtener más información, consulte [Cancelar una reserva de capacidad](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [CancelCapacityReservation](#) la Referencia de AWS CLI comandos.

## cancel-conversion-task

En el siguiente ejemplo de código, se muestra cómo usar `cancel-conversion-task`.

### AWS CLI

Para cancelar una conversión activa de una instancia o un volumen

En este ejemplo, se cancela la carga asociada al ID de tarea `import-i-fh 95npoc`. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 cancel-conversion-task --conversion-task-id import-i-fh95npoc
```

- Para obtener API más información, consulte la Referencia de comandos [CancelConversionTask](#).AWS CLI

## cancel-export-task

En el siguiente ejemplo de código, se muestra cómo usar `cancel-export-task`.

### AWS CLI

Para cancelar una tarea de exportación activa

En este ejemplo se cancela una tarea de exportación activa con el identificador de tarea `export-i-fgelt0i7`. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 cancel-export-task --export-task-id export-i-fgelt0i7
```

- Para API obtener más información, consulte la Referencia de [CancelExportTask](#) comandos AWS CLI .

## cancel-image-launch-permission

En el siguiente ejemplo de código, se muestra cómo usar `cancel-image-launch-permission`.

### AWS CLI

Para cancelar la AMI compartición con tu cuenta de Amazon Web Services

En el siguiente `cancel-image-launch-permission` ejemplo, se elimina tu cuenta de los permisos AMI de lanzamiento de la persona especificada.

```
aws ec2 cancel-image-launch-permission \  
  --image-id ami-0123456789example \  
  --region us-east-1
```

Salida:

```
{  
  "Return": true  
}
```

Para obtener más información, consulta [Cancelar la AMI compartición con tu cuenta de Amazon Web Services](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [CancelImageLaunchPermission](#) la Referencia de AWS CLI comandos.

## cancel-import-task

En el siguiente ejemplo de código, se muestra cómo usar `cancel-import-task`.

### AWS CLI

Para cancelar una tarea de importación

El siguiente `cancel-import-task` ejemplo cancela la tarea de importación de imágenes especificada.

```
aws ec2 cancel-import-task \  
  --import-task-id import-ami-1234567890abcdef0
```

Salida:

```
{
  "ImportTaskId": "import-ami-1234567890abcdef0",
  "PreviousState": "active",
  "State": "deleting"
}
```

- Para API obtener más información, consulte [CancelImportTask](#) la Referencia de AWS CLI comandos.

## cancel-reserved-instances-listing

En el siguiente ejemplo de código, se muestra cómo usar `cancel-reserved-instances-listing`.

AWS CLI

Para cancelar un listado de instancias reservadas

En el siguiente `cancel-reserved-instances-listing` ejemplo, se cancela la lista de instancias reservadas especificada.

```
aws ec2 cancel-reserved-instances-listing \
  --reserved-instances-listing-id 5ec28771-05ff-4b9b-aa31-9e57dexample
```

- Para API obtener más información, consulte [CancelReservedInstancesListing](#) la Referencia de AWS CLI comandos.

## cancel-spot-fleet-requests

En el siguiente ejemplo de código, se muestra cómo usar `cancel-spot-fleet-requests`.

AWS CLI

Ejemplo 1: cancelar una solicitud de flota de Spot y cancelar las instancias asociadas

En el siguiente `cancel-spot-fleet-requests` ejemplo, se cancela una solicitud de flota puntual y se cancelan las instancias puntuales y bajo demanda asociadas.

```
aws ec2 cancel-spot-fleet-requests \  
  --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \  
  --terminate-instances
```

Salida:

```
{  
  "SuccessfulFleetRequests": [  
    {  
      "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",  
      "CurrentSpotFleetRequestState": "cancelled_terminating",  
      "PreviousSpotFleetRequestState": "active"  
    }  
  ],  
  "UnsuccessfulFleetRequests": []  
}
```

Para obtener más información, consulte [Cancelar una solicitud de flota puntual](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

Ejemplo 2: Para cancelar una solicitud de flota puntual sin cancelar las instancias asociadas

En el siguiente `cancel-spot-fleet-requests` ejemplo, se cancela una solicitud de flota puntual sin cancelar las instancias puntuales y bajo demanda asociadas.

```
aws ec2 cancel-spot-fleet-requests \  
  --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \  
  --no-terminate-instances
```

Salida:

```
{  
  "SuccessfulFleetRequests": [  
    {  
      "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",  
      "CurrentSpotFleetRequestState": "cancelled_running",  
      "PreviousSpotFleetRequestState": "active"  
    }  
  ],  
  "UnsuccessfulFleetRequests": []  
}
```

Para obtener más información, consulte [Cancelar una solicitud de flota puntual](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [CancelSpotFleetRequests](#) la Referencia de AWS CLI comandos.

## **cancel-spot-instance-requests**

En el siguiente ejemplo de código, se muestra cómo usar `cancel-spot-instance-requests`.

### AWS CLI

Para cancelar las solicitudes de instancias puntuales

Este comando de ejemplo cancela una solicitud de instancia de spot.

Comando:

```
aws ec2 cancel-spot-instance-requests --spot-instance-request-ids sir-08b93456
```

Salida:

```
{
  "CancelledSpotInstanceRequests": [
    {
      "State": "cancelled",
      "SpotInstanceRequestId": "sir-08b93456"
    }
  ]
}
```

- Para API obtener más información, consulte [CancelSpotInstanceRequests](#) la Referencia de AWS CLI comandos.

## **confirm-product-instance**

En el siguiente ejemplo de código, se muestra cómo usar `confirm-product-instance`.

### AWS CLI

Para confirmar la instancia del producto

En este ejemplo se determina si el código de producto especificado está asociado a la instancia especificada.

Comando:

```
aws ec2 confirm-product-instance --product-code 774F4FF8 --instance-id i-1234567890abcdef0
```

Salida:

```
{  
  "OwnerId": "123456789012"  
}
```

- Para API obtener más información, consulte [ConfirmProductInstance](#) la Referencia de AWS CLI comandos.

## copy-fpga-image

En el siguiente ejemplo de código, se muestra cómo usar `copy-fpga-image`.

AWS CLI

Para copiar una FPGA imagen de Amazon

En este ejemplo, se copia lo especificado AFI de la `us-east-1` región a la región actual (`eu-west-1`).

Comando:

```
aws ec2 copy-fpga-image --name copy-afi --source-fpga-image-id afi-0d123e123bfc85abc --source-region us-east-1 --region eu-west-1
```

Salida:

```
{  
  "FpgaImageId": "afi-06b12350a123fbabc"  
}
```

- Para API obtener más información, consulte [CopyFpgaImage](#) la Referencia de AWS CLI comandos.

## copy-image

En el siguiente ejemplo de código, se muestra cómo usar `copy-image`.

### AWS CLI

Ejemplo 1: Para copiar un objeto AMI a otra región

El siguiente comando de `copy-image` ejemplo copia lo especificado AMI de la `us-west-2` región a la `us-east-1` región y añade una breve descripción.

```
aws ec2 copy-image \  
  --region us-east-1 \  
  --name ami-name \  
  --source-region us-west-2 \  
  --source-image-id ami-066877671789bd71b \  
  --description "This is my copied image."
```

Salida:

```
{  
  "ImageId": "ami-0123456789abcdefg"  
}
```

Para obtener más información, consulta [Copiar un AMI](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para copiar un archivo AMI a otra región y cifrar la instantánea de respaldo

El siguiente `copy-image` comando copia lo especificado AMI de la `us-west-2` región a la región actual y cifra la instantánea de respaldo con la clave especificada KMS.

```
aws ec2 copy-image \  
  --source-region us-west-2 \  
  --name ami-name \  
  --source-image-id ami-066877671789bd71b \  
  --encrypted \  
  --kms-key-id alias/my-kms-key
```

Salida:

```
{
```



```
"ImageId": "ami-0123456789abcdefg"
}
```

Para obtener más información, consulta [Copiar un AMI](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 3: Para incluir las AMI etiquetas definidas por el usuario al copiar un AMI

El siguiente `copy-image` comando utiliza el `--copy-image-tags` parámetro para copiar las AMI etiquetas definidas por el usuario al copiar. AMI

```
aws ec2 copy-image \
  --region us-east-1 \
  --name ami-name \
  --source-region us-west-2 \
  --source-image-id ami-066877671789bd71b \
  --description "This is my copied image." \
  --copy-image-tags
```

Salida:

```
{
  "ImageId": "ami-0123456789abcdefg"
}
```

Para obtener más información, consulta [Copiar un AMI](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [CopyImage](#) la Referencia de AWS CLI comandos.

## copy-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `copy-snapshot`.

### AWS CLI

Ejemplo 1: Para copiar una instantánea a otra región

El siguiente comando de `copy-snapshot` ejemplo copia la instantánea especificada de la `us-west-2` región a la `us-east-1` región y añade una breve descripción.

```
aws ec2 copy-snapshot \
  --region us-east-1 \
```

```
--source-region us-west-2 \  
--source-snapshot-id snap-066877671789bd71b \  
--description "This is my copied snapshot."
```

Salida:

```
{  
  "SnapshotId": "snap-066877671789bd71b"  
}
```

Para obtener más información, consulta [Copiar una EBS instantánea de Amazon](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para copiar una instantánea no cifrada y cifrar la nueva instantánea

El siguiente `copy-snapshot` comando copia la instantánea no cifrada especificada de la `us-west-2` región a la región actual y cifra la nueva instantánea con la clave especificada. KMS

```
aws ec2 copy-snapshot \  
  --source-region us-west-2 \  
  --source-snapshot-id snap-066877671789bd71b \  
  --encrypted \  
  --kms-key-id alias/my-kms-key
```

Salida:

```
{  
  "SnapshotId": "snap-066877671789bd71b"  
}
```

Para obtener más información, consulta [Copiar una EBS instantánea de Amazon](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [CopySnapshot](#) la Referencia de AWS CLI comandos.

## **create-capacity-reservation-fleet**

En el siguiente ejemplo de código, se muestra cómo usar `create-capacity-reservation-fleet`.

## AWS CLI

Para crear una flota de reserva de capacidad

En el siguiente `create-capacity-reservation-fleet` ejemplo, se crea una flota de reserva de capacidad para el tipo de instancia especificado en la solicitud, hasta la capacidad objetivo total especificada. El número de instancias para las que la flota reserva capacidad depende de la capacidad de destino total y de las ponderaciones de tipo de instancia que especifique en la solicitud. Especifica los tipos de instancias que se van a usar y una prioridad para cada uno de los tipos de instancias designados.

```
aws ec2 create-capacity-reservation-fleet \  
--total-target-capacity 24 \  
--allocation-strategy prioritized \  
--instance-match-criteria open \  
--tenancy default \  
--end-date 2022-12-31T23:59:59.000Z \  
--instance-type-specifications file://instanceTypeSpecification.json
```

Contenidos de `instanceTypeSpecification.json`:

```
[  
  {  
    "InstanceType": "m5.xlarge",  
    "InstancePlatform": "Linux/UNIX",  
    "Weight": 3.0,  
    "AvailabilityZone": "us-east-1a",  
    "EbsOptimized": true,  
    "Priority" : 1  
  }  
]
```

Salida:

```
{  
  "Status": "submitted",  
  "TotalFulfilledCapacity": 0.0,  
  "CapacityReservationFleetId": "crf-abcdef01234567890",  
  "TotalTargetCapacity": 24  
}
```

Para obtener más información sobre las flotas de reserva de capacidad, consulta [Flotas de reserva de capacidad](#) en la Guía EC2 del usuario de Amazon.

Para obtener más información sobre el peso del tipo de instancia y la capacidad objetivo total, consulta el artículo [Peso del tipo de instancia](#) y [capacidad objetivo total](#) en la Guía del EC2 usuario de Amazon.

Para obtener más información sobre la designación de prioridades para tipos de instancias específicos, consulte [Estrategia de asignación](#) y [prioridad de tipo de instancia](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [CreateCapacityReservationFleet](#) la Referencia de AWS CLI comandos.

## create-capacity-reservation

En el siguiente ejemplo de código, se muestra cómo usar `create-capacity-reservation`.

### AWS CLI

Ejemplo 1: Para crear una reserva de capacidad

El siguiente `create-capacity-reservation` ejemplo crea una reserva de capacidad en la zona de `eu-west-1a` disponibilidad, en la que puede lanzar tres `t2.medium` instancias que ejecuten un sistema operativo Linux/Unix. De forma predeterminada, la reserva de capacidad se crea con criterios de coincidencia de instancias abiertas y no admite el almacenamiento efímero, y permanece activa hasta que la canceles manualmente.

```
aws ec2 create-capacity-reservation \
  --availability-zone eu-west-1a \
  --instance-type t2.medium \
  --instance-platform Linux/UNIX \
  --instance-count 3
```

Salida:

```
{
  "CapacityReservation": {
    "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
    "EndDateType": "unlimited",
    "AvailabilityZone": "eu-west-1a",
    "InstanceMatchCriteria": "open",
```

```

    "EphemeralStorage": false,
    "CreateDate": "2019-08-16T09:27:35.000Z",
    "AvailableInstanceCount": 3,
    "InstancePlatform": "Linux/UNIX",
    "TotalInstanceCount": 3,
    "State": "active",
    "Tenancy": "default",
    "EbsOptimized": false,
    "InstanceType": "t2.medium"
  }
}

```

Ejemplo 2: Para crear una reserva de capacidad que finalice automáticamente en una fecha y hora especificadas

El siguiente `create-capacity-reservation` ejemplo crea una reserva de capacidad en la zona de `eu-west-1a` disponibilidad, en la que puede lanzar tres `m5.large` instancias que ejecuten un sistema operativo Linux/Unix. Esta reserva de capacidad finaliza automáticamente el 31 de agosto de 2019 a las 23:59:59.

```

aws ec2 create-capacity-reservation \
  --availability-zone eu-west-1a \
  --instance-type m5.large \
  --instance-platform Linux/UNIX \
  --instance-count 3 \
  --end-date-type Limited \
  --end-date 2019-08-31T23:59:59Z

```

Salida:

```

{
  "CapacityReservation": {
    "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
    "EndDateType": "limited",
    "AvailabilityZone": "eu-west-1a",
    "EndDate": "2019-08-31T23:59:59.000Z",
    "InstanceMatchCriteria": "open",
    "EphemeralStorage": false,
    "CreateDate": "2019-08-16T10:15:53.000Z",
    "AvailableInstanceCount": 3,
    "InstancePlatform": "Linux/UNIX",
    "TotalInstanceCount": 3,
  }
}

```

```
    "State": "active",
    "Tenancy": "default",
    "EbsOptimized": false,
    "InstanceType": "m5.large"
  }
}
```

Ejemplo 3: Para crear una reserva de capacidad que solo acepte lanzamientos de instancias segmentadas

En el siguiente `create-capacity-reservation` ejemplo, se crea una reserva de capacidad que solo acepta los lanzamientos de instancias de destino.

```
aws ec2 create-capacity-reservation \
  --availability-zone eu-west-1a \
  --instance-type m5.large \
  --instance-platform Linux/UNIX \
  --instance-count 3 \
  --instance-match-criteria targeted
```

Salida:

```
{
  "CapacityReservation": {
    "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
    "EndDateType": "unlimited",
    "AvailabilityZone": "eu-west-1a",
    "InstanceMatchCriteria": "targeted",
    "EphemeralStorage": false,
    "CreateDate": "2019-08-16T10:21:57.000Z",
    "AvailableInstanceCount": 3,
    "InstancePlatform": "Linux/UNIX",
    "TotalInstanceCount": 3,
    "State": "active",
    "Tenancy": "default",
    "EbsOptimized": false,
    "InstanceType": "m5.large"
  }
}
```

Para obtener más información, consulte [Creación de una reserva de capacidad](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [CreateCapacityReservation](#) la Referencia de AWS CLI comandos.

## create-carrier-gateway

En el siguiente ejemplo de código, se muestra cómo usar `create-carrier-gateway`.

### AWS CLI

Para crear una puerta de enlace portadora

El siguiente `create-carrier-gateway` ejemplo crea una puerta de enlace de operador para el operador especificado VPC.

```
aws ec2 create-carrier-gateway \  
--vpc-id vpc-0c529aEXAMPLE1111
```

Salida:

```
{  
  "CarrierGateway": {  
    "CarrierGatewayId": "cagw-0465cdEXAMPLE1111",  
    "VpcId": "vpc-0c529aEXAMPLE1111",  
    "State": "pending",  
    "OwnerId": "123456789012"  
  }  
}
```

Para obtener más información, consulte [Carrier Gateways](#) en la Guía del usuario AWS de Wavelength.

- Para API obtener más información, consulte [CreateCarrierGateway](#) la Referencia de AWS CLI comandos.

## create-client-vpn-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `create-client-vpn-endpoint`.

### AWS CLI

Para crear un VPN punto final de cliente

En el siguiente `create-client-vpn-endpoint` ejemplo, se crea un VPN punto final de cliente que utiliza la autenticación mutua y se especifica un valor para el CIDR bloque de cliente.

```
aws ec2 create-client-vpn-endpoint \
  --client-cidr-block "172.31.0.0/16" \
  --server-certificate-arn arn:aws:acm:ap-south-1:123456789012:certificate/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
  --authentication-options Type=certificate-
authentication,MutualAuthentication={ClientRootCertificateChainArn=arn:aws:acm:ap-
south-1:123456789012:certificate/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE} \
  --connection-log-options Enabled=false
```

Salida:

```
{
  "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
  "Status": {
    "Code": "pending-associate"
  },
  "DnsName": "cvpn-endpoint-123456789123abcde.prod.clientvpn.ap-
south-1.amazonaws.com"
}
```

Para obtener más información, consulte los [VPN puntos finales del cliente](#) en la Guía del VPN administrador de AWS clientes.

- Para API obtener más información, consulte [CreateClientVpnEndpoint](#) la Referencia de AWS CLI comandos.

## create-client-vpn-route

En el siguiente ejemplo de código, se muestra cómo usar `create-client-vpn-route`.

AWS CLI

Para crear una ruta para un VPN punto final del cliente

El siguiente `create-client-vpn-route` ejemplo agrega una ruta a Internet (`0.0.0.0/0`) para la subred especificada del VPN punto final del cliente.

```
aws ec2 create-client-vpn-route \
```



```
--client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \  
--destination-cidr-block 0.0.0.0/0 \  
--target-vpc-subnet-id subnet-0123456789abcabca
```

Salida:

```
{  
  "Status": {  
    "Code": "creating"  
  }  
}
```

Para obtener más información, consulte [Rutas](#) en la Guía del VPN administrador del AWS cliente.

- Para API obtener más información, consulte [CreateClientVpnRoute](#) la Referencia de AWS CLI comandos.

## create-coip-cidr

En el siguiente ejemplo de código, se muestra cómo usar `create-coip-cidr`.

### AWS CLI

Para crear un rango de direcciones IP (CoIP) propiedad del cliente

En el siguiente `create-coip-cidr` ejemplo, se crea el rango especificado de direcciones de CoIP en el grupo de CoIP especificado.

```
aws ec2 create-coip-cidr \  
--cidr 15.0.0.0/24 \  
--coip-pool-id ipv4pool-coip-1234567890abcdefg
```

Salida:

```
{  
  "CoipCidr": {  
    "Cidr": "15.0.0.0/24",  
    "CoipPoolId": "ipv4pool-coip-1234567890abcdefg",  
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890"  
  }  
}
```

Para obtener más información, consulte [Direcciones IP propiedad del cliente](#) en la Guía del usuario de AWS .

- Para API obtener más información, consulte la Referencia [CreateCoipCidr](#) de AWS CLI comandos.

## create-coip-pool

En el siguiente ejemplo de código, se muestra cómo usar `create-coip-pool`.

### AWS CLI

Para crear un conjunto de direcciones IP (CoIP) propiedad del cliente

El siguiente `create-coip-pool` ejemplo crea un grupo de CoIP para las direcciones de CoIP en la tabla de rutas de la puerta de enlace local especificada.

```
aws ec2 create-coip-pool \
  --local-gateway-route-table-id lgw-rtb-abcdefg1234567890
```

Salida:

```
{
  "CoipPool": {
    "PoolId": "ipv4pool-coip-1234567890abcdefg",
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",
    "PoolArn": "arn:aws:ec2:us-west-2:123456789012:coip-pool/ipv4pool-
coip-1234567890abcdefg"
  }
}
```

Para obtener más información, consulte [Direcciones IP propiedad del cliente](#) en la Guía del usuario de AWS .

- Para API obtener más información, consulte la Referencia [CreateCoipPool](#) de AWS CLI comandos.

## create-customer-gateway

En el siguiente ejemplo de código, se muestra cómo usar `create-customer-gateway`.

## AWS CLI

Para crear una pasarela de clientes

En este ejemplo, se crea una puerta de enlace para el cliente con la dirección IP especificada para su interfaz exterior.

Comando:

```
aws ec2 create-customer-gateway --type ipsec.1 --public-ip 12.1.2.3 --bgp-asn 65534
```

Salida:

```
{
  "CustomerGateway": {
    "CustomerGatewayId": "cgw-0e11f167",
    "IpAddress": "12.1.2.3",
    "State": "available",
    "Type": "ipsec.1",
    "BgpAsn": "65534"
  }
}
```

- Para API obtener más información, consulte [CreateCustomerGateway](#) la Referencia de AWS CLI comandos.

## create-default-subnet

En el siguiente ejemplo de código, se muestra cómo usar create-default-subnet.

## AWS CLI

Para crear una subred predeterminada

En este ejemplo, se crea una subred predeterminada en Availability Zone. us-east-2a

Comando:

```
aws ec2 create-default-subnet --availability-zone us-east-2a

{
  "Subnet": {
    "AvailabilityZone": "us-east-2a",
```

```
    "Tags": [],
    "AvailableIpAddressCount": 4091,
    "DefaultForAz": true,
    "Ipv6CidrBlockAssociationSet": [],
    "VpcId": "vpc-1a2b3c4d",
    "State": "available",
    "MapPublicIpOnLaunch": true,
    "SubnetId": "subnet-1122aabb",
    "CidrBlock": "172.31.32.0/20",
    "AssignIpv6AddressOnCreation": false
  }
}
```

- Para API obtener más información, consulte [CreateDefaultSubnet](#) la Referencia de AWS CLI comandos.

## create-default-vpc

En el siguiente ejemplo de código, se muestra cómo usar `create-default-vpc`.

### AWS CLI

Para crear un valor por defecto VPC

En este ejemplo se crea un valor predeterminado VPC.

Comando:

```
aws ec2 create-default-vpc
```

Salida:

```
{
  "Vpc": {
    "VpcId": "vpc-8eaae5ea",
    "InstanceTenancy": "default",
    "Tags": [],
    "Ipv6CidrBlockAssociationSet": [],
    "State": "pending",
    "DhcpOptionsId": "dopt-af0c32c6",
    "CidrBlock": "172.31.0.0/16",
    "IsDefault": true
  }
}
```

```
}
```

- Para API obtener más información, consulte [CreateDefaultVpca](#) Referencia de AWS CLI comandos.

## create-dhcp-options

En el siguiente ejemplo de código, se muestra cómo usar `create-dhcp-options`.

### AWS CLI

Para crear un conjunto de DHCP opciones

El siguiente `create-dhcp-options` ejemplo crea un conjunto de DHCP opciones que especifica el nombre de dominio, los servidores de nombres de dominio y el tipo de BIOS nodo de red.

```
aws ec2 create-dhcp-options \  
  --dhcp-configuration \  
    "Key=domain-name-servers,Values=10.2.5.1,10.2.5.2" \  
    "Key=domain-name,Values=example.com" \  
    "Key=netbios-node-type,Values=2"
```

Salida:

```
{  
  "DhcpOptions": {  
    "DhcpConfigurations": [  
      {  
        "Key": "domain-name",  
        "Values": [  
          {  
            "Value": "example.com"  
          }  
        ]  
      },  
      {  
        "Key": "domain-name-servers",  
        "Values": [  
          {  
            "Value": "10.2.5.1"  
          }  
        ],  
      }  
    ]  
  }  
}
```

```

        {
            "Value": "10.2.5.2"
        }
    ],
    {
        "Key": "netbios-node-type",
        "Values": [
            {
                "Value": "2"
            }
        ]
    }
],
"DhcpOptionsId": "dopt-06d52773eff4c55f3"
}
}

```

- Para API obtener más información, consulte [CreateDhcpOptions](#) la Referencia de AWS CLI comandos.

## create-egress-only-internet-gateway

En el siguiente ejemplo de código, se muestra cómo usar `create-egress-only-internet-gateway`.

### AWS CLI

Para crear una puerta de enlace a Internet de solo salida

En este ejemplo, se crea una puerta de enlace de Internet de solo salida para el objeto especificado. VPC

Comando:

```
aws ec2 create-egress-only-internet-gateway --vpc-id vpc-0c62a468
```

Salida:

```

{
  "EgressOnlyInternetGateway": {
    "EgressOnlyInternetGatewayId": "eigw-015e0e244e24dfe8a",

```

```

    "Attachments": [
      {
        "State": "attached",
        "VpcId": "vpc-0c62a468"
      }
    ]
  }
}

```

- Para API obtener más información, consulte la Referencia de [CreateEgressOnlyInternetGateway](#) comandos AWS CLI .

## create-fleet

En el siguiente ejemplo de código, se muestra cómo usar `create-fleet`.

### AWS CLI

Ejemplo 1: Para crear una EC2 flota que lance instancias puntuales como modelo de compra predeterminado

En el siguiente `create-fleet` ejemplo, se crea una EC2 flota con los parámetros mínimos necesarios para lanzar una flota: una plantilla de lanzamiento, una capacidad objetivo y un modelo de compra predeterminado. La plantilla de inicialización se identifica mediante su ID y número de versión. La capacidad objetivo de la flota es de 2 instancias y el modelo de compra predeterminado `spot`, lo que hace que la flota lance 2 instancias puntuales.

Al crear una EC2 flota, utilice un JSON archivo para especificar la información sobre las instancias que se van a lanzar.

```

aws ec2 create-fleet \
  --cli-input-json file://file_name.json

```

Contenido de `file_name.json`:

```

{
  "LaunchTemplateConfigs": [
    {
      "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-0e8c754449b27161c",
        "Version": "1"
      }
    }
  ]
}

```

```

    }
  }
],
"TargetCapacitySpecification": {
  "TotalTargetCapacity": 2,
  "DefaultTargetCapacityType": "spot"
}
}

```

Salida:

```

{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}

```

Ejemplo 2: Para crear una EC2 flota que lance instancias bajo demanda como modelo de compra predeterminado

En el siguiente `create-fleet` ejemplo, se crea una EC2 flota con los parámetros mínimos necesarios para lanzar una flota: una plantilla de lanzamiento, una capacidad objetivo y un modelo de compra predeterminado. La plantilla de inicialización se identifica mediante su ID y número de versión. La capacidad objetivo de la flota es de 2 instancias y el modelo de compra predeterminado es `on-demand`, lo que hace que la flota lance 2 instancias bajo demanda.

Al crear una EC2 flota, utilice un JSON archivo para especificar la información sobre las instancias que se van a lanzar.

```

aws ec2 create-fleet \
  --cli-input-json file://file_name.json

```

Contenido de `file_name.json`:

```

{
  "LaunchTemplateConfigs": [
    {
      "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-0e8c754449b27161c",
        "Version": "1"
      }
    }
  ],
  "TargetCapacitySpecification": {

```



```
"TotalTargetCapacity": 2,  
"DefaultTargetCapacityType": "on-demand"  
}  
}
```

Salida:

```
{  
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"  
}
```

Ejemplo 3: Para crear una EC2 flota que lance instancias bajo demanda como capacidad principal

En el siguiente `create-fleet` ejemplo, se crea una EC2 flota que especifica la capacidad objetivo total de 2 instancias para la flota y una capacidad objetivo de 1 instancia bajo demanda. El modelo de compra predeterminado es `spot`. La flota lanza 1 instancia bajo demanda según lo especificado, pero necesita lanzar una instancia más para alcanzar la capacidad objetivo total. El modelo de compra para la diferencia se calcula como `TotalTargetCapacity - OnDemandTargetCapacity = DefaultTargetCapacityType`, lo que da como resultado que la flota lance 1 instancia puntual.

Al crear una EC2 flota, utilice un JSON archivo para especificar la información sobre las instancias que se van a lanzar.

```
aws ec2 create-fleet \  
  --cli-input-json file://file_name.json
```

Contenido de `file_name.json`:

```
{  
  "LaunchTemplateConfigs": [  
    {  
      "LaunchTemplateSpecification": {  
        "LaunchTemplateId": "lt-0e8c754449b27161c",  
        "Version": "1"  
      }  
    }  
  ],  
  "TargetCapacitySpecification": {  
    "TotalTargetCapacity": 2,  
    "OnDemandTargetCapacity": 1,  
  }  
}
```

```

    "DefaultTargetCapacityType": "spot"
  }
}

```

Salida:

```

{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}

```

Ejemplo 4: Para crear una EC2 flota que lance instancias puntuales mediante la estrategia de asignación de precios más bajos

Si la estrategia de asignación de instancias de spot no se especifica, se usa la estrategia predeterminada, que es `lowest-price`. En el siguiente `create-fleet` ejemplo, se crea una EC2 flota mediante la estrategia de `lowest-price` asignación. Las tres especificaciones de inicialización, que invalidan la plantilla de inicialización, tienen tipos de instancias diferentes, pero las mismas capacidad ponderada y subred. La capacidad objetivo total es de 2 instancias y el modelo de compra predeterminado `esspot`. La EC2 flota lanza 2 instancias puntuales utilizando el tipo de instancia de la especificación de lanzamiento con el precio más bajo.

Al crear una EC2 flota, utilice un JSON archivo para especificar la información sobre las instancias que se van a lanzar.

```

aws ec2 create-fleet \
  --cli-input-json file://file_name.jsonContents of file_name.json::
{
  "LaunchTemplateConfigs": [
    {
      "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-0e8c754449b27161c",
        "Version": "1"
      },
      "Overrides": [
        {
          "InstanceType": "c4.large",
          "WeightedCapacity": 1,
          "SubnetId": "subnet-a4f6c5d3"
        },
        {
          "InstanceType": "c3.large",

```

```

        "WeightedCapacity": 1,
        "SubnetId": "subnet-a4f6c5d3"
    },
    {
        "InstanceType": "c5.large",
        "WeightedCapacity": 1,
        "SubnetId": "subnet-a4f6c5d3"
    }
]
},
"TargetCapacitySpecification": {
    "TotalTargetCapacity": 2,
    "DefaultTargetCapacityType": "spot"
}
}

```

Salida:

```

{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}

```

- Para API obtener más información, consulte [CreateFleet](#) la Referencia de AWS CLI comandos.

## create-flow-logs

En el siguiente ejemplo de código, se muestra cómo usar `create-flow-logs`.

### AWS CLI

Ejemplo 1: Para crear un registro de flujo

El siguiente `create-flow-logs` ejemplo crea un registro de flujo que captura todo el tráfico rechazado para la interfaz de red especificada. Los registros de flujo se envían a un grupo de CloudWatch registros en Logs mediante los permisos de la IAM función especificada.

```

aws ec2 create-flow-logs \
  --resource-type NetworkInterface \
  --resource-ids eni-11223344556677889 \
  --traffic-type REJECT \

```

```
--log-group-name my-flow-logs \  
--deliver-logs-permission-arn arn:aws:iam::123456789101:role/publishFlowLogs
```

Salida:

```
{  
  "ClientToken": "so0eNA2uSHUN1HI0S2cJ305GuIX1CezaRdGtexample",  
  "FlowLogIds": [  
    "fl-12345678901234567"  
  ],  
  "Unsuccessful": []  
}
```

Para obtener más información, consulta [VPCFlow Logs](#) en la Guía del VPC usuario de Amazon.

Ejemplo 2: Para crear un registro de flujo con un formato personalizado

El siguiente `create-flow-logs` ejemplo crea un registro de flujo que captura todo el tráfico del especificado VPC y entrega los registros de flujo a un bucket de Amazon S3. El parámetro `--log-format` especifica un formato personalizado para las entradas de registros de flujo. Para ejecutar este comando en Windows, cambie las comillas simples (') por comillas dobles («).

```
aws ec2 create-flow-logs \  
  --resource-type VPC \  
  --resource-ids vpc-00112233344556677 \  
  --traffic-type ALL \  
  --log-destination-type s3 \  
  --log-destination arn:aws:s3:::flow-log-bucket/my-custom-flow-logs/ \  
  --log-format '${version} ${vpc-id} ${subnet-id} ${instance-id} ${srcaddr} ${dstaddr} ${srcport} ${dstport} ${protocol} ${tcp-flags} ${type} ${pkt-srcaddr} ${pkt-dstaddr}'
```

Para obtener más información, consulta [VPCFlow Logs](#) en la Guía del VPC usuario de Amazon.

Ejemplo 3: Para crear un registro de flujo con un intervalo de agregación máximo de un minuto

El siguiente `create-flow-logs` ejemplo crea un registro de flujo que captura todo el tráfico del especificado VPC y entrega los registros de flujo a un bucket de Amazon S3. El `--max-aggregation-interval` parámetro especifica un intervalo de agregación máximo de 60 segundos (1 minuto).

```
aws ec2 create-flow-logs \  
  --max-aggregation-interval 60
```

```
--resource-type VPC \  
--resource-ids vpc-00112233344556677 \  
--traffic-type ALL \  
--log-destination-type s3 \  
--log-destination arn:aws:s3:::flow-log-bucket/my-custom-flow-logs/ \  
--max-aggregation-interval 60
```

Para obtener más información, consulta [VPCFlow Logs](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [CreateFlowLogs](#) la Referencia de AWS CLI comandos.

## create-fpga-image

En el siguiente ejemplo de código, se muestra cómo usar `create-fpga-image`.

### AWS CLI

Para crear una FPGA imagen de Amazon

En este ejemplo, se crea una AFI a partir del tarball especificado en el bucket especificado.

Comando:

```
aws ec2 create-fpga-image --name my-afi --description test-afi --input-storage-  
location Bucket=my-fpga-bucket,Key=dcp/17_12_22-103226.Developer_CL.tar --logs-  
storage-location Bucket=my-fpga-bucket,Key=logs
```

Salida:

```
{  
  "FpgaImageId": "afi-0d123e123bfc85abc",  
  "FpgaImageGlobalId": "agfi-123cb27b5e84a0abc"  
}
```

- Para API obtener más información, consulte [CreateFpgaImage](#) la Referencia de AWS CLI comandos.

## create-image

En el siguiente ejemplo de código, se muestra cómo usar `create-image`.

## AWS CLI

Ejemplo 1: Para crear una a AMI partir de una instancia EBS respaldada por Amazon

En el siguiente `create-image` ejemplo, se crea una AMI a partir de la instancia especificada.

```
aws ec2 create-image \  
  --instance-id i-1234567890abcdef0 \  
  --name "My server" \  
  --description "An AMI for my server"
```

Salida:

```
{  
  "ImageId": "ami-abcdef01234567890"  
}
```

Para obtener más información sobre cómo especificar un mapeo de dispositivos de bloques para ustedAMI, consulte [Especificar un mapeo de dispositivos de bloques para un mapeo](#) de dispositivos de bloques AMI en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para crear una AMI desde una instancia EBS respaldada por Amazon sin reiniciar

En el siguiente `create-image` ejemplo, se crea un parámetro `--no-reboot` AMI y se establece el mismo para que la instancia no se reinicie antes de crear la imagen.

```
aws ec2 create-image \  
  --instance-id i-1234567890abcdef0 \  
  --name "My server" \  
  --no-reboot
```

Salida:

```
{  
  "ImageId": "ami-abcdef01234567890"  
}
```

Para obtener más información sobre cómo especificar un mapeo de dispositivos de bloques para ustedAMI, consulte [Especificar un mapeo de dispositivos de bloques para un mapeo](#) de dispositivos de bloques AMI en la Guía del EC2 usuario de Amazon.

### Ejemplo 3: etiquetar una imagen AMI y una instantánea al crearla

El siguiente `create-image` ejemplo crea una AMI y etiqueta las instantáneas AMI y las mismas con la misma etiqueta `cost-center=cc123`

```
aws ec2 create-image \  
  --instance-id i-1234567890abcdef0 \  
  --name "My server" \  
  --tag-specifications "ResourceType=image,Tags=[{Key=cost-center,Value=cc123}]" "ResourceType=snapshot,Tags=[{Key=cost-center,Value=cc123}]"
```

Salida:

```
{  
  "ImageId": "ami-abcdef01234567890"  
}
```

Para obtener más información sobre cómo etiquetar tus recursos al crearlos, consulta [Añadir etiquetas al crear recursos en](#) la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [CreateImage](#) la Referencia de AWS CLI comandos.

### `create-instance-connect-endpoint`

En el siguiente ejemplo de código, se muestra cómo usar `create-instance-connect-endpoint`.

AWS CLI

Para crear un punto final de EC2 Instance Connect

En el siguiente `create-instance-connect-endpoint` ejemplo, se crea un punto final EC2 Instance Connect en la subred especificada.

```
aws ec2 create-instance-connect-endpoint \  
  --region us-east-1 \  
  --subnet-id subnet-0123456789example
```

Salida:

```
{
```

```

    "VpcId": "vpc-0123abcd",
    "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-
connect-endpoint/eice-0123456789example",
    "AvailabilityZone": "us-east-1a",
    "NetworkInterfaceIds": [
        "eni-0123abcd"
    ],
    "PreserveClientIp": true,
    "Tags": [],
    "FipsDnsName": "eice-0123456789example.0123abcd.fips.ec2-instance-connect-
endpoint.us-east-1.amazonaws.com",
    "StateMessage": "",
    "State": "create-complete",
    "DnsName": "eice-0123456789example.0123abcd.ec2-instance-connect-endpoint.us-
east-1.amazonaws.com",
    "SubnetId": "subnet-0123abcd",
    "OwnerId": "111111111111",
    "SecurityGroupIds": [
        "sg-0123abcd"
    ],
    "InstanceConnectEndpointId": "eice-0123456789example",
    "CreatedAt": "2023-04-07T15:43:53.000Z"
}

```

Para obtener más información, consulte [Create an EC2 Instance Connect Endpoint](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [CreateInstanceConnectEndpoint](#) la Referencia de AWS CLI comandos.

## create-instance-event-window

En el siguiente ejemplo de código, se muestra cómo usar `create-instance-event-window`.

### AWS CLI

Ejemplo 1: Para crear una ventana de eventos con un rango de tiempo

El siguiente `create-instance-event-window` ejemplo crea una ventana de eventos con un rango de tiempo. No puede especificar el parámetro `cron-expression`.

```

aws ec2 create-instance-event-window \
  --region us-east-1 \

```



```

--time-range StartWeekDay=monday, StartHour=2, EndWeekDay=wednesday, EndHour=8 \
--tag-specifications "ResourceType=instance-event-
window, Tags=[{Key=K1, Value=V1}]" \
--name myEventWindowName

```

Salida:

```

{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "TimeRanges": [
      {
        "StartWeekDay": "monday",
        "StartHour": 2,
        "EndWeekDay": "wednesday",
        "EndHour": 8
      }
    ],
    "Name": "myEventWindowName",
    "State": "creating",
    "Tags": [
      {
        "Key": "K1",
        "Value": "V1"
      }
    ]
  }
}

```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para crear una ventana de eventos con una expresión cron

El siguiente `create-instance-event-window` ejemplo crea una ventana de eventos con una expresión cron. No puede especificar el parámetro `time-range`.

```

aws ec2 create-instance-event-window \
--region us-east-1 \
--cron-expression "* 21-23 * * 2,3" \
--tag-specifications "ResourceType=instance-event-
window, Tags=[{Key=K1, Value=V1}]" \

```

```
--name myEventWindowName
```

Salida:

```
{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "State": "creating",
    "Tags": [
      {
        "Key": "K1",
        "Value": "V1"
      }
    ]
  }
}
```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [CreateInstanceEventWindow](#) la Referencia de AWS CLI comandos.

## create-instance-export-task

En el siguiente ejemplo de código, se muestra cómo usar `create-instance-export-task`.

AWS CLI

Para exportar una instancia

Este comando de ejemplo crea una tarea para exportar la instancia `i-1234567890abcdef0` al bucket `myexportbucket` de Amazon S3.

Comando:

```
aws ec2 create-instance-export-task --description "RHEL5 instance" --
instance-id i-1234567890abcdef0 --target-environment vmware --export-to-s3-
task DiskImageFormat=vmdk, ContainerFormat=ova, S3Bucket=myexportbucket, S3Prefix=RHEL5
```

Salida:

```
{
  "ExportTask": {
    "State": "active",
    "InstanceExportDetails": {
      "InstanceId": "i-1234567890abcdef0",
      "TargetEnvironment": "vmware"
    },
    "ExportToS3Task": {
      "S3Bucket": "myexportbucket",
      "S3Key": "RHEL5export-i-fh8sjjsq.ova",
      "DiskImageFormat": "vmdk",
      "ContainerFormat": "ova"
    },
    "Description": "RHEL5 instance",
    "ExportTaskId": "export-i-fh8sjjsq"
  }
}
```

- Para [CreateInstanceExportTask](#) obtener AWS CLI más información, consulte la Referencia de comandos. API

## create-internet-gateway

En el siguiente ejemplo de código, se muestra cómo usar `create-internet-gateway`.

AWS CLI

Para crear una puerta de enlace a Internet

En el siguiente `create-internet-gateway` ejemplo, se crea una puerta de enlace a Internet con la etiqueta `Name=my-igw`.

```
aws ec2 create-internet-gateway \
  --tag-specifications ResourceType=internet-gateway,Tags=[{Key=Name,Value=my-igw}]
```

Salida:

```
{
```

```

    "InternetGateway": {
      "Attachments": [],
      "InternetGatewayId": "igw-0d0fb496b3994d755",
      "OwnerId": "123456789012",
      "Tags": [
        {
          "Key": "Name",
          "Value": "my-igw"
        }
      ]
    }
  }
}

```

Para obtener más información, consulta [las pasarelas de Internet](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [CreateInternetGateway](#) la Referencia de AWS CLI comandos.

## create-ipam-pool

En el siguiente ejemplo de código, se muestra cómo usar `create-ipam-pool`.

### AWS CLI

Para crear un IPAM grupo

En el siguiente `create-ipam-pool` ejemplo, se crea un IPAM grupo.

(Linux):

```

aws ec2 create-ipam-pool \
  --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 \
  --address-family ipv4 \
  --auto-import \
  --allocation-min-netmask-length 16 \
  --allocation-max-netmask-length 26 \
  --allocation-default-netmask-length 24 \
  --allocation-resource-tags "Key=Environment, Value=Preprod" \
  --tag-specifications 'ResourceType=ipam-pool, Tags=[{Key=Name, Value="Preprod pool"}]'

```

(Windows):

```
aws ec2 create-ipam-pool ^
  --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 ^
  --address-family ipv4 ^
  --auto-import ^
  --allocation-min-netmask-length 16 ^
  --allocation-max-netmask-length 26 ^
  --allocation-default-netmask-length 24 ^
  --allocation-resource-tags "Key=Environment,Value=Preprod" ^
  --tag-specifications ResourceType=ipam-pool,Tags=[{Key=Name,Value="Preprod
pool"}]
```

Salida:

```
{
  "IpamPool": {
    "OwnerId": "123456789012",
    "IpamPoolId": "ipam-pool-0533048da7d823723",
    "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0533048da7d823723",
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-02fc38cd4c48e7d38",
    "IpamScopeType": "private",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
    "IpamRegion": "us-east-1",
    "Locale": "None",
    "PoolDepth": 1,
    "State": "create-in-progress",
    "AutoImport": true,
    "AddressFamily": "ipv4",
    "AllocationMinNetmaskLength": 16,
    "AllocationMaxNetmaskLength": 26,
    "AllocationDefaultNetmaskLength": 24,
    "AllocationResourceTags": [
      {
        "Key": "Environment",
        "Value": "Preprod"
      }
    ],
    "Tags": [
      {
        "Key": "Name",
        "Value": "Preprod pool"
      }
    ]
  }
}
```

```
    ]  
  }  
}
```

Para obtener más información, consulta Cómo [planificar el aprovisionamiento de direcciones IP](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [CreateIpamPool](#) la Referencia de AWS CLI comandos.

## **create-ipam-resource-discovery**

En el siguiente ejemplo de código, se muestra cómo usar `create-ipam-resource-discovery`.

### AWS CLI

Para crear un descubrimiento de recursos

En este ejemplo, usted es un IPAM administrador delegado que desea crear y compartir una detección de recursos con el IPAM administrador de otra AWS organización para que el administrador de la otra organización pueda administrar y supervisar las direcciones IP de los recursos de su organización.

### Importante

Este ejemplo incluye ambas `--operating-regions` opciones porque, si bien son opcionales, deben configurarse de una forma determinada para integrar correctamente la detección de recursos con una IPAM. \* `--operating-regions` debe coincidir con las regiones en las que tiene los recursos que desea IPAM descubrir. `--region` Si hay regiones en las que no IPAM desea administrar las direcciones IP (por ejemplo, por motivos de conformidad), no las incluya. \* `--region` debe coincidir con la región de origen a la IPAM que desee asociarlo. Debe crear la detección de recursos en la misma región en la que IPAM se creó. Por ejemplo, IPAM si la opción a la que se está asociando se creó en `us-east-1`, `--region us-east-1` inclúyala en la solicitud. Si no las especifica, ambas opciones son `--region --operating-regions` las predeterminadas de la región en la que se ejecuta el comando.

En este ejemplo, las regiones operativas de las que IPAM estamos integrando incluyen, y. `us-west-1 us-west-2 ap-south-1` Cuando creamos la detección de recursos, queremos IPAM descubrir las direcciones IP de los recursos en `us-west-1` y `us-west-2` no `ap-south-1`.

Por lo tanto, solo las incluimos `--operating-regions RegionName='us-west-1' RegionName='us-west-2'` en la solicitud.

El siguiente `create-ipam-resource-discovery` ejemplo crea un descubrimiento IPAM de recursos.

```
aws ec2 create-ipam-resource-discovery \
  --description 'Example-resource-discovery' \
  --tag-specifications 'ResourceType=ipam-resource-discovery,Tags=[{Key=cost-center,Value=cc123}]' \
  --operating-regions RegionName='us-west-1' RegionName='us-west-2' \
  --region us-east-1
```

Salida:

```
{
  "IpamResourceDiscovery": {
    "OwnerId": "149977607591",
    "IpamResourceDiscoveryId": "ipam-res-disco-0257046d8aa78b8bc",
    "IpamResourceDiscoveryArn": "arn:aws:ec2::149977607591:ipam-resource-discovery/ipam-res-disco-0257046d8aa78b8bc",
    "IpamResourceDiscoveryRegion": "us-east-1",
    "Description": "'Example-resource-discovery'",
    "OperatingRegions": [
      {"RegionName": "us-west-1"},
      {"RegionName": "us-west-2"},
      {"RegionName": "us-east-1"}
    ],
    "IsDefault": false,
    "State": "create-in-progress",
    "Tags": [
      {
        "Key": "cost-center",
        "Value": "cc123"
      }
    ]
  }
}
```

Una vez que haya creado una detección de recursos, puede que desee compartirla con otro administrador IPAM delegado, lo que puede hacer con [create-resource-share](#)ella. Para obtener más información, consulta la [sección Integración IPAM con cuentas ajenas a tu organización](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [CreatelpamResourceDiscovery](#) la Referencia de AWS CLI comandos.

## create-ipam-scope

En el siguiente ejemplo de código, se muestra cómo usar `create-ipam-scope`.

### AWS CLI

Para crear un IPAM ámbito

En el siguiente `create-ipam-scope` ejemplo, se crea un IPAM ámbito.

(Linux):

```
aws ec2 create-ipam-scope \  
  --ipam-id ipam-08440e7a3acde3908 \  
  --description "Example description" \  
  --tag-specifications 'ResourceType=ipam-scope,Tags=[{Key=Name,Value="Example  
name value"}]'
```

(Windows):

```
aws ec2 create-ipam-scope ^  
  --ipam-id ipam-08440e7a3acde3908 ^  
  --description "Example description" ^  
  --tag-specifications ResourceType=ipam-scope,Tags=[{Key=Name,Value="Example name  
value"}]
```

Salida:

```
{  
  "IpamScope": {  
    "OwnerId": "123456789012",  
    "IpamScopeId": "ipam-scope-01c1ebab2b63bd7e4",  
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-  
scope-01c1ebab2b63bd7e4",  
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",  
    "IpamRegion": "us-east-1",  
    "IpamScopeType": "private",  
    "IsDefault": false,  
    "Description": "Example description",
```



```

    "PoolCount": 0,
    "State": "create-in-progress",
    "Tags": [
      {
        "Key": "Name",
        "Value": "Example name value"
      }
    ]
  }
}

```

Para obtener más información, consulta [Crear ámbitos adicionales](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [CreateIpamScope](#) la Referencia de AWS CLI comandos.

## create-ipam

En el siguiente ejemplo de código, se muestra cómo usar create-ipam.

### AWS CLI

Para crear un IPAM

El siguiente create-ipam ejemplo crea un IPAM.

(Linux):

```

aws ec2 create-ipam \
  --description "Example description" \
  --operating-regions "RegionName=us-east-2" "RegionName=us-west-1" \
  --tag-specifications 'ResourceType=ipam,Tags=[{Key=Name,Value=ExampleIPAM}]'

```

(Windows):

```

aws ec2 create-ipam ^
  --description "Example description" ^
  --operating-regions "RegionName=us-east-2" "RegionName=us-west-1" ^
  --tag-specifications ResourceType=ipam,Tags=[{Key=Name,Value=ExampleIPAM}]

```

Salida:

```
{
  "Ipam": {
    "OwnerId": "123456789012",
    "IpamId": "ipam-036486dfa6af58ee0",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-036486dfa6af58ee0",
    "IpamRegion": "us-east-1",
    "PublicDefaultScopeId": "ipam-scope-071b8042b0195c183",
    "PrivateDefaultScopeId": "ipam-scope-0807405dece705a30",
    "ScopeCount": 2,
    "OperatingRegions": [
      {
        "RegionName": "us-east-2"
      },
      {
        "RegionName": "us-west-1"
      },
      {
        "RegionName": "us-east-1"
      }
    ],
    "State": "create-in-progress",
    "Tags": [
      {
        "Key": "Name",
        "Value": "ExampleIPAM"
      }
    ]
  }
}
```

Para obtener más información, consulta [Crear una IPAM](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [CreateIpam](#) la Referencia de AWS CLI comandos.

## create-key-pair

En el siguiente ejemplo de código, se muestra cómo usar `create-key-pair`.

### AWS CLI

Crear un par de claves

En este ejemplo, se crea un par de claves denominado `MyKeyPair`.

Comando:

```
aws ec2 create-key-pair --key-name MyKeyPair
```

El resultado es una ASCII versión de la clave privada y la huella digital de la clave. Debe guardar la clave en un archivo.

Para obtener más información, consulte [Uso del par de claves](#) en la Guía del usuario de la Interfaz de la línea de comandos de AWS .

- Para API obtener más información, consulte [CreateKeyPair](#) la Referencia de AWS CLI comandos.

## **create-launch-template-version**

En el siguiente ejemplo de código, se muestra cómo usar `create-launch-template-version`.

### AWS CLI

Para crear una versión de plantilla de lanzamiento

En este ejemplo, se crea una nueva versión de la plantilla de lanzamiento basada en la versión 1 de la plantilla de lanzamiento y se especifica un AMI identificador diferente.

Comando:

```
aws ec2 create-launch-template-version --launch-template-id lt-0abcd290751193123  
--version-description WebVersion2 --source-version 1 --launch-template-data  
'{"ImageId": "ami-c998b6b2"}'
```

Salida:

```
{  
  "LaunchTemplateVersion": {  
    "VersionDescription": "WebVersion2",  
    "LaunchTemplateId": "lt-0abcd290751193123",  
    "LaunchTemplateName": "WebServers",  
    "VersionNumber": 2,  
    "CreatedBy": "arn:aws:iam::123456789012:root",  
    "LaunchTemplateData": {  
      "ImageId": "ami-c998b6b2",
```

```

    "InstanceType": "t2.micro",
    "NetworkInterfaces": [
      {
        "Ipv6Addresses": [
          {
            "Ipv6Address": "2001:db8:1234:1a00::123"
          }
        ],
        "DeviceIndex": 0,
        "SubnetId": "subnet-7b16de0c",
        "AssociatePublicIpAddress": true
      }
    ],
    "DefaultVersion": false,
    "CreateTime": "2017-12-01T13:35:46.000Z"
  }
}

```

- Para API obtener más información, consulte [CreateLaunchTemplateVersion](#) la Referencia de AWS CLI comandos.

## create-launch-template

En el siguiente ejemplo de código, se muestra cómo usar `create-launch-template`.

### AWS CLI

Ejemplo 1: Crear una plantilla de lanzamiento

En el siguiente `create-launch-template` ejemplo, se crea una plantilla de lanzamiento que especifica la subred en la que se va a lanzar la instancia, se asigna una dirección IP pública y una IPv6 dirección a la instancia y se crea una etiqueta para la instancia.

```

aws ec2 create-launch-template \
  --launch-template-name TemplateForWebServer \
  --version-description WebVersion1 \
  --launch-template-data '{"NetworkInterfaces":
  [{"AssociatePublicIpAddress":true,"DeviceIndex":0,"Ipv6AddressCount":1,"SubnetId":"subnet-7b
  [{"ResourceType":"instance","Tags":[{"Key":"purpose","Value":"webserver"}]}]}'

```

Salida:

```
{
  "LaunchTemplate": {
    "LatestVersionNumber": 1,
    "LaunchTemplateId": "lt-01238c059e3466abc",
    "LaunchTemplateName": "TemplateForWebServer",
    "DefaultVersionNumber": 1,
    "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
    "CreateTime": "2019-01-27T09:13:24.000Z"
  }
}
```

Para obtener más información, consulte Lanzamiento de una instancia desde una plantilla de lanzamiento en la Guía del usuario de Amazon Elastic Compute Cloud. Para obtener información sobre los parámetros con JSON formato de comillas, consulta Cómo citar cadenas en la Guía del usuario de la AWS interfaz de línea de comandos.

Ejemplo 2: Para crear una plantilla de lanzamiento para Amazon EC2 Auto Scaling

En el siguiente `create-launch-template` ejemplo, se crea una plantilla de lanzamiento con varias etiquetas y un mapeo de dispositivos de bloques para especificar un EBS volumen adicional cuando se lanza una instancia. Especifique un valor `Groups` que corresponda a los grupos de seguridad en los VPC que su grupo de Auto Scaling lanzará las instancias. Especifique las subredes VPC y como propiedades del grupo Auto Scaling.

```
aws ec2 create-launch-template \
  --launch-template-name TemplateForAutoScaling \
  --version-description AutoScalingVersion1 \
  --launch-template-data '{"NetworkInterfaces":
  [{"DeviceIndex":0,"AssociatePublicIpAddress":true,"Groups":
  ["sg-7c227019,sg-903004f8"],"DeleteOnTermination":true}],"ImageId":"ami-
  b42209de","InstanceType":"m4.large","TagSpecifications":
  [{"ResourceType":"instance","Tags":[{"Key":"environment","Value":"production"},
  {"Key":"purpose","Value":"webserver"}]},{"ResourceType":"volume","Tags":
  [{"Key":"environment","Value":"production"}, {"Key":"cost-
  center","Value":"cc123"}]}],"BlockDeviceMappings":[{"DeviceName":"/dev/sda1","Ebs":
  {"VolumeSize":100}}]}' --region us-east-1
```

Salida:

```
{
  "LaunchTemplate": {
```

```

    "LatestVersionNumber": 1,
    "LaunchTemplateId": "lt-0123c79c33a54e0abc",
    "LaunchTemplateName": "TemplateForAutoScaling",
    "DefaultVersionNumber": 1,
    "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
    "CreateTime": "2019-04-30T18:16:06.000Z"
  }
}

```

Para obtener más información, consulte [Creación de una plantilla de lanzamiento para un grupo de Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling. Para obtener información sobre los parámetros JSON formateados entre comillas, consulte la sección [Cómo citar cadenas](#) en la Guía del usuario de la interfaz de línea de AWS comandos.

Ejemplo 3: Para crear una plantilla de lanzamiento que especifique el cifrado de volúmenes EBS

En el siguiente `create-launch-template` ejemplo, se crea una plantilla de lanzamiento que incluye EBS los volúmenes cifrados creados a partir de una instantánea no cifrada. También etiqueta los volúmenes durante la creación. Si el cifrado está deshabilitado de forma predeterminada, debe especificar la opción `"Encrypted"` que se muestra en el siguiente ejemplo. Si utiliza la `"KmsKeyId"` opción para especificar una gestión por parte del cliente CMK, también debe especificarla aunque el `"Encrypted"` cifrado esté activado de forma predeterminada.

```

aws ec2 create-launch-template \
  --launch-template-name TemplateForEncryption \
  --launch-template-data file://config.json

```

Contenidos de `config.json`:

```

{
  "BlockDeviceMappings": [
    {
      "DeviceName": "/dev/sda1",
      "Ebs": {
        "VolumeType": "gp2",
        "DeleteOnTermination": true,
        "SnapshotId": "snap-066877671789bd71b",
        "Encrypted": true,
        "KmsKeyId": "arn:aws:kms:us-east-1:012345678910:key/abcd1234-a123-456a-a12b-a123b4cd56ef"
      }
    }
  ]
}

```

```

    }
  }
],
"ImageId":"ami-00068cd7555f543d5",
"InstanceType":"c5.large",
"TagSpecifications":[
  {
    "ResourceType":"volume",
    "Tags":[
      {
        "Key":"encrypted",
        "Value":"yes"
      }
    ]
  }
]
}
}
}

```

Salida:

```

{
  "LaunchTemplate": {
    "LatestVersionNumber": 1,
    "LaunchTemplateId": "lt-0d5bd51bcf8530abc",
    "LaunchTemplateName": "TemplateForEncryption",
    "DefaultVersionNumber": 1,
    "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
    "CreateTime": "2020-01-07T19:08:36.000Z"
  }
}

```

Para obtener más información, consulte Restauración de un EBS volumen de Amazon a partir de una instantánea y cifrado de forma predeterminada en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte [CreateLaunchTemplate](#) la Referencia de AWS CLI comandos.

## **create-local-gateway-route-table-virtual-interface-group-association**

En el siguiente ejemplo de código, se muestra cómo usar `create-local-gateway-route-table-virtual-interface-group-association`.

## AWS CLI

Para asociar una tabla de enrutamiento de una puerta de enlace local a un grupo de interfaces virtuales (VIFs)

El siguiente `create-local-gateway-route-table-virtual-interface-group-association` ejemplo crea una asociación entre la tabla de rutas de la puerta de enlace local especificada y el VIF grupo.

```
aws ec2 create-local-gateway-route-table-virtual-interface-group-association \
  --local-gateway-route-table-id lgw-rtb-exampleidabcd1234 \
  --local-gateway-virtual-interface-group-id lgw-vif-grp-exampleid0123abcd
```

Salida:

```
{
  "LocalGatewayRouteTableVirtualInterfaceGroupAssociation": {
    "LocalGatewayRouteTableVirtualInterfaceGroupAssociationId": "lgw-vif-grp-
assoc-exampleid12345678",
    "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-exampleid0123abcd",
    "LocalGatewayId": "lgw-exampleid11223344",
    "LocalGatewayRouteTableId": "lgw-rtb-exampleidabcd1234",
    "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:111122223333:local-
gateway-route-table/lgw-rtb-exampleidabcd1234",
    "OwnerId": "111122223333",
    "State": "pending",
    "Tags": []
  }
}
```

Para obtener más información, consulta [las asociaciones de VIF grupos](#) en la Guía del usuario de AWS Outposts.

- Para API obtener más información, consulte [CreateLocalGatewayRouteTableVirtualInterfaceGroupAssociation](#) la Referencia de AWS CLI comandos.

## **create-local-gateway-route-table-vpc-association**

En el siguiente ejemplo de código, se muestra cómo usar `create-local-gateway-route-table-vpc-association`.



## AWS CLI

Para asociar a a una VPC tabla de enrutamiento

El siguiente `create-local-gateway-route-table-vpc-association` ejemplo asocia lo especificado VPC a la tabla de enrutamiento de la puerta de enlace local especificada.

```
aws ec2 create-local-gateway-route-table-vpc-association \  
  --local-gateway-route-table-id lgw-rtb-059615ef7dEXAMPLE \  
  --vpc-id vpc-07ef66ac71EXAMPLE
```

Salida:

```
{  
  "LocalGatewayRouteTableVpcAssociation": {  
    "LocalGatewayRouteTableVpcAssociationId": "lgw-vpc-assoc-0ee765bcc8EXAMPLE",  
    "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE",  
    "LocalGatewayId": "lgw-09b493aa7cEXAMPLE",  
    "VpcId": "vpc-07ef66ac71EXAMPLE",  
    "State": "associated"  
  }  
}
```

- Para API obtener más información, consulte [CreateLocalGatewayRouteTableVpcAssociation](#) la Referencia de AWS CLI comandos.

## `create-local-gateway-route-table`

En el siguiente ejemplo de código, se muestra cómo usar `create-local-gateway-route-table`.

## AWS CLI

Para crear una tabla de rutas de puerta de enlace local

El siguiente `create-local-gateway-route-table` ejemplo crea una tabla de enrutamiento de puerta de enlace local con el modo de VPC enrutamiento directo.

```
aws ec2 create-local-gateway-route-table \  
  --local-gateway-id lgw-1a2b3c4d5e6f7g8h9 \  
  --mode direct-vpc-routing
```

Salida:

```
{
  "LocalGatewayRouteTable": {
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",
    "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:111122223333:local-
gateway-route-table/lgw-rtb-abcdefg1234567890",
    "LocalGatewayId": "lgw-1a2b3c4d5e6f7g8h9",
    "OutpostArn": "arn:aws:outposts:us-west-2:111122223333:outpost/
op-021345abcdef67890",
    "OwnerId": "111122223333",
    "State": "pending",
    "Tags": [],
    "Mode": "direct-vpc-routing"
  }
}
```

Para obtener más información, consulte [Tablas de enrutamiento de puerta de enlace locales](#) en la Guía del usuario de AWS Outposts.

- Para API obtener más información, consulte [CreateLocalGatewayRouteTable](#) la Referencia de AWS CLI comandos.

## create-local-gateway-route

En el siguiente ejemplo de código, se muestra cómo usar `create-local-gateway-route`.

AWS CLI

Para crear una ruta estática para una tabla de rutas de una puerta de enlace local

El siguiente `create-local-gateway-route` ejemplo crea la ruta especificada en la tabla de rutas de la puerta de enlace local especificada.

```
aws ec2 create-local-gateway-route \
  --destination-cidr-block 0.0.0.0/0 \
  --local-gateway-route-table-id lgw-rtb-059615ef7dEXAMPLE
```

Salida:

```
{
```

```

    "Route": {
      "DestinationCidrBlock": "0.0.0.0/0",
      "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
      "Type": "static",
      "State": "deleted",
      "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE"
    }
  }
}

```

- Para API obtener más información, consulte [CreateLocalGatewayRoute](#) la Referencia de AWS CLI comandos.

## create-managed-prefix-list

En el siguiente ejemplo de código, se muestra cómo usar `create-managed-prefix-list`.

### AWS CLI

Para crear una lista de prefijos

En el siguiente `create-managed-prefix-list` ejemplo, se crea una lista de IPv4 prefijos con un máximo de 10 entradas y se crean 2 entradas en la lista de prefijos.

```

aws ec2 create-managed-prefix-list \
  --address-family IPv4 \
  --max-entries 10 \
  --entries Cidr=10.0.0.0/16,Description=vpc-a Cidr=10.2.0.0/16,Description=vpc-b \
  --prefix-list-name vpc-cidrs

```

Salida:

```

{
  "PrefixList": {
    "PrefixListId": "pl-0123456abcabc1",
    "AddressFamily": "IPv4",
    "State": "create-in-progress",
    "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/pl-0123456abcabc1",
    "PrefixListName": "vpc-cidrs",
    "MaxEntries": 10,
    "Version": 1,
  }
}

```

```

    "Tags": [],
    "OwnerId": "123456789012"
  }
}

```

Para obtener más información, consulta [Listas de prefijos gestionadas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [CreateManagedPrefixList](#) la Referencia de AWS CLI comandos.

## create-nat-gateway

En el siguiente ejemplo de código, se muestra cómo usar `create-nat-gateway`.

### AWS CLI

Ejemplo 1: Para crear una NAT puerta de enlace pública

El siguiente `create-nat-gateway` ejemplo crea una NAT puerta de enlace pública en la subred especificada y asocia la dirección IP elástica con el ID de asignación especificado. Al crear una NAT puerta de enlace pública, debe asociar una dirección IP elástica.

```

aws ec2 create-nat-gateway \
  --subnet-id subnet-0250c25a1fEXAMPLE \
  --allocation-id eipalloc-09ad461b0dEXAMPLE

```

Salida:

```

{
  "NatGateway": {
    "CreateTime": "2021-12-01T22:22:38.000Z",
    "NatGatewayAddresses": [
      {
        "AllocationId": "eipalloc-09ad461b0dEXAMPLE"
      }
    ],
    "NatGatewayId": "nat-0c61bf8a12EXAMPLE",
    "State": "pending",
    "SubnetId": "subnet-0250c25a1fEXAMPLE",
    "VpcId": "vpc-0a60eb65b4EXAMPLE",
    "ConnectivityType": "public"
  }
}

```

```
}
}
```

Para obtener más información, consulta [NATlas pasarelas](#) en la Guía del VPC usuario de Amazon.

Ejemplo 2: Para crear una puerta de enlace privada NAT

El siguiente `create-nat-gateway` ejemplo crea una NAT puerta de enlace privada en la subred especificada. Una NAT puerta de enlace privada no tiene una dirección IP elástica asociada.

```
aws ec2 create-nat-gateway \
  --subnet-id subnet-0250c25a1fEXAMPLE \
  --connectivity-type private
```

Salida:

```
{
  "NatGateway": {
    "CreateTime": "2021-12-01T22:26:00.000Z",
    "NatGatewayAddresses": [
      {}
    ],
    "NatGatewayId": "nat-011b568379EXAMPLE",
    "State": "pending",
    "SubnetId": "subnet-0250c25a1fEXAMPLE",
    "VpcId": "vpc-0a60eb65b4EXAMPLE",
    "ConnectivityType": "private"
  }
}
```

Para obtener más información, consulta [NATlas pasarelas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [CreateNatGateway](#) la Referencia de AWS CLI comandos.

## **create-network-acl-entry**

En el siguiente ejemplo de código, se muestra cómo usar `create-network-acl-entry`.

## AWS CLI

Para crear una ACL entrada de red

En este ejemplo, se crea una entrada para la red especificada ACL. La regla permite la entrada de tráfico desde cualquier IPv4 dirección (0.0.0.0/0) del UDP puerto 53 (DNS) a cualquier subred asociada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 create-network-acl-entry --network-acl-id acl-5fb85d36 --ingress --rule-number 100 --protocol udp --port-range From=53,To=53 --cidr-block 0.0.0.0/0 --rule-action allow
```

En este ejemplo, se crea una regla para la red especificada ACL que permite la entrada de tráfico desde cualquier IPv6 dirección (:: /0) del puerto 80 (). TCP HTTP

Comando:

```
aws ec2 create-network-acl-entry --network-acl-id acl-5fb85d36 --ingress --rule-number 120 --protocol tcp --port-range From=80,To=80 --ipv6-cidr-block ::/0 --rule-action allow
```

- Para API obtener más información, consulte la Referencia [CreateNetworkAclEntry](#) de AWS CLI comandos.

## create-network-acl

En el siguiente ejemplo de código, se muestra cómo usar create-network-acl.

## AWS CLI

Para crear una red ACL

En este ejemplo se crea una red ACL para el especificado VPC.

Comando:

```
aws ec2 create-network-acl --vpc-id vpc-a01106c2
```

Salida:

```
{
  "NetworkAcl": {
    "Associations": [],
    "NetworkAclId": "acl-5fb85d36",
    "VpcId": "vpc-a01106c2",
    "Tags": [],
    "Entries": [
      {
        "CidrBlock": "0.0.0.0/0",
        "RuleNumber": 32767,
        "Protocol": "-1",
        "Egress": true,
        "RuleAction": "deny"
      },
      {
        "CidrBlock": "0.0.0.0/0",
        "RuleNumber": 32767,
        "Protocol": "-1",
        "Egress": false,
        "RuleAction": "deny"
      }
    ],
    "IsDefault": false
  }
}
```

- Para API obtener más información, consulte [CreateNetworkAcl](#) la Referencia de AWS CLI comandos.

## **create-network-insights-access-scope**

En el siguiente ejemplo de código, se muestra cómo usar `create-network-insights-access-scope`.

### AWS CLI

Para crear un ámbito de acceso a la red

El siguiente `create-network-insights-access-scope` ejemplo crea un ámbito de acceso a la red.

```
aws ec2 create-network-insights-access-scope \
```

```
--cli-input-json file://access-scope-file.json
```

Contenidos de `access-scope-file.json`:

```
{
  "MatchPaths": [
    {
      "Source": {
        "ResourceStatement": {
          "Resources": [
            "vpc-abcd12e3"
          ]
        }
      }
    }
  ],
  "ExcludePaths": [
    {
      "Source": {
        "ResourceStatement": {
          "ResourceTypes": [
            "AWS::EC2::InternetGateway"
          ]
        }
      }
    }
  ]
}
```

Salida:

```
{
  "NetworkInsightsAccessScope": {
    "NetworkInsightsAccessScopeId": "nis-123456789abc01234",
    "NetworkInsightsAccessScopeArn": "arn:aws:ec2:us-east-1:123456789012:network-insights-access-scope/nis-123456789abc01234",
    "CreateDate": "2022-01-25T19:20:28.796000+00:00",
    "UpdateDate": "2022-01-25T19:20:28.797000+00:00"
  },
  "NetworkInsightsAccessScopeContent": {
    "NetworkInsightsAccessScopeId": "nis-123456789abc01234",
    "MatchPaths": [
      {
```



```

        "Source": {
            "ResourceStatement": {
                "Resources": [
                    "vpc-abcd12e3"
                ]
            }
        }
    ],
    "ExcludePaths": [
        {
            "Source": {
                "ResourceStatement": {
                    "ResourceTypes": [
                        "AWS::EC2::InternetGateway"
                    ]
                }
            }
        }
    ]
}

```

Para obtener más información, consulte [Introducción al analizador de acceso a la red mediante la Guía del AWS CLI analizador](#) de acceso a la red.

- Para API obtener más información, consulte la Referencia [CreateNetworkInsightsAccessScope](#) de AWS CLI comandos.

## create-network-insights-path

En el siguiente ejemplo de código, se muestra cómo usar `create-network-insights-path`.

### AWS CLI

Para crear una ruta

En el `create-network-insights-path` ejemplo siguiente se crea una ruta. El origen es la puerta de enlace de Internet especificada y el destino es la EC2 instancia especificada. Para determinar si se puede acceder al destino mediante el protocolo y el puerto especificados, analice la ruta mediante el `start-network-insights-analysis` comando.

```
aws ec2 create-network-insights-path \
```

```
--source igw-0797cccdc9d73b0e5 \  
--destination i-0495d385ad28331c7 \  
--destination-port 22 \  
--protocol TCP
```

Salida:

```
{  
  "NetworkInsightsPaths": {  
    "NetworkInsightsPathId": "nip-0b26f224f1d131fa8",  
    "NetworkInsightsPathArn": "arn:aws:ec2:us-east-1:123456789012:network-  
insights-path/nip-0b26f224f1d131fa8",  
    "CreateDate": "2021-01-20T22:43:46.933Z",  
    "Source": "igw-0797cccdc9d73b0e5",  
    "Destination": "i-0495d385ad28331c7",  
    "Protocol": "tcp"  
  }  
}
```

Para obtener más información, consulte [Cómo empezar a utilizar la Guía del AWS CLI Reachability Analyzer](#).

- Para API obtener más información, consulte la Referencia de [CreateNetworkInsightsPath](#) comandos AWS CLI .

## create-network-interface-permission

En el siguiente ejemplo de código, se muestra cómo usar `create-network-interface-permission`.

AWS CLI

Para crear un permiso de interfaz de red

En este ejemplo, se concede permiso a la cuenta 123456789012 para adjuntar la interfaz de red `eni-1a2b3c4d` a una instancia.

Comando:

```
aws ec2 create-network-interface-permission --network-interface-id eni-1a2b3c4d --  
aws-account-id 123456789012 --permission INSTANCE-ATTACH
```

Salida:

```
{
  "InterfacePermission": {
    "PermissionState": {
      "State": "GRANTED"
    },
    "NetworkInterfacePermissionId": "eni-perm-06fd19020ede149ea",
    "NetworkInterfaceId": "eni-1a2b3c4d",
    "Permission": "INSTANCE-ATTACH",
    "AwsAccountId": "123456789012"
  }
}
```

- Para API obtener más información, consulte [CreateNetworkInterfacePermission](#) la Referencia de AWS CLI comandos.

## create-network-interface

En el siguiente ejemplo de código, se muestra cómo usar `create-network-interface`.

AWS CLI

Ejemplo 1: Para especificar una IPv4 dirección para una interfaz de red

El siguiente `create-network-interface` ejemplo crea una interfaz de red para la subred especificada con la IPv4 dirección principal especificada.

```
aws ec2 create-network-interface \
  --subnet-id subnet-00a24d0d67acf6333 \
  --description "my network interface" \
  --groups sg-09dfba7ed20cda78b \
  --private-ip-address 10.0.8.17
```

Salida:

```
{
  "NetworkInterface": {
    "AvailabilityZone": "us-west-2a",
    "Description": "my network interface",
    "Groups": [
      {
```

```

        "GroupName": "my-security-group",
        "GroupId": "sg-09dfba7ed20cda78b"
    }
],
"InterfaceType": "interface",
"Ipv6Addresses": [],
"MacAddress": "06:6a:0f:9a:49:37",
"NetworkInterfaceId": "eni-0492b355f0cf3b3f8",
"OwnerId": "123456789012",
"PrivateDnsName": "ip-10-0-8-18.us-west-2.compute.internal",
"PrivateIpAddress": "10.0.8.17",
"PrivateIpAddresses": [
    {
        "Primary": true,
        "PrivateDnsName": "ip-10-0-8-17.us-west-2.compute.internal",
        "PrivateIpAddress": "10.0.8.17"
    }
],
"RequesterId": "AIDA4Z3Y7GSXTMEXAMPLE",
"RequesterManaged": false,
"SourceDestCheck": true,
"Status": "pending",
"SubnetId": "subnet-00a24d0d67acf6333",
"TagSet": [],
"VpcId": "vpc-02723a0feeeb9d57b"
}
}

```

Ejemplo 2: Para crear una interfaz de red con una IPv4 dirección y una IPv6 dirección

El siguiente `create-network-interface` ejemplo crea una interfaz de red para la subred especificada con una IPv4 dirección y una IPv6 dirección seleccionadas por AmazonEC2.

```

aws ec2 create-network-interface \
  --subnet-id subnet-00a24d0d67acf6333 \
  --description "my dual stack network interface" \
  --ipv6-address-count 1 \
  --groups sg-09dfba7ed20cda78b

```

Salida:

```

{
  "NetworkInterface": {

```

```

    "AvailabilityZone": "us-west-2a",
    "Description": "my dual stack network interface",
    "Groups": [
      {
        "GroupName": "my-security-group",
        "GroupId": "sg-09dfba7ed20cda78b"
      }
    ],
    "InterfaceType": "interface",
    "Ipv6Addresses": [
      {
        "Ipv6Address": "2600:1f13:cfe:3650:a1dc:237c:393a:4ba7",
        "IsPrimaryIpv6": false
      }
    ],
    "MacAddress": "06:b8:68:d2:b2:2d",
    "NetworkInterfaceId": "eni-05da417453f9a84bf",
    "OwnerId": "123456789012",
    "PrivateDnsName": "ip-10-0-8-18.us-west-2.compute.internal",
    "PrivateIpAddress": "10.0.8.18",
    "PrivateIpAddresses": [
      {
        "Primary": true,
        "PrivateDnsName": "ip-10-0-8-18.us-west-2.compute.internal",
        "PrivateIpAddress": "10.0.8.18"
      }
    ],
    "RequesterId": "AIDA4Z3Y7GSXTMEXAMPLE",
    "RequesterManaged": false,
    "SourceDestCheck": true,
    "Status": "pending",
    "SubnetId": "subnet-00a24d0d67acf6333",
    "TagSet": [],
    "VpcId": "vpc-02723a0feeeb9d57b",
    "Ipv6Address": "2600:1f13:cfe:3650:a1dc:237c:393a:4ba7"
  }
}

```

**Ejemplo 3:** Para crear una interfaz de red con opciones de configuración de seguimiento de conexiones

En el siguiente `create-network-interface` ejemplo, se crea una interfaz de red y se configuran los tiempos de espera del seguimiento de las conexiones inactivas.

```
aws ec2 create-network-interface \  
--subnet-id subnet-00a24d0d67acf6333 \  
--groups sg-02e57dbcf0331c1b \  
--connection-tracking-specification TcpEstablishedTimeout=86400,UdpTimeout=60
```

Salida:

```
{  
  "NetworkInterface": {  
    "AvailabilityZone": "us-west-2a",  
    "ConnectionTrackingConfiguration": {  
      "TcpEstablishedTimeout": 86400,  
      "UdpTimeout": 60  
    },  
    "Description": "",  
    "Groups": [  
      {  
        "GroupName": "my-security-group",  
        "GroupId": "sg-02e57dbcf0331c1b"  
      }  
    ],  
    "InterfaceType": "interface",  
    "Ipv6Addresses": [],  
    "MacAddress": "06:4c:53:de:6d:91",  
    "NetworkInterfaceId": "eni-0c133586e08903d0b",  
    "OwnerId": "123456789012",  
    "PrivateDnsName": "ip-10-0-8-94.us-west-2.compute.internal",  
    "PrivateIpAddress": "10.0.8.94",  
    "PrivateIpAddresses": [  
      {  
        "Primary": true,  
        "PrivateDnsName": "ip-10-0-8-94.us-west-2.compute.internal",  
        "PrivateIpAddress": "10.0.8.94"  
      }  
    ],  
    "RequesterId": "AIDA4Z3Y7GSXTMEXAMPLE",  
    "RequesterManaged": false,  
    "SourceDestCheck": true,  
    "Status": "pending",  
    "SubnetId": "subnet-00a24d0d67acf6333",  
    "TagSet": [],  
    "VpcId": "vpc-02723a0feeeb9d57b"  
  }  
}
```

```
}
```

#### Ejemplo 4: Para crear un adaptador Elastic Fabric

El siguiente `create-network-interface` ejemplo crea unEFA.

```
aws ec2 create-network-interface \  
  --interface-type efa \  
  --subnet-id subnet-00a24d0d67acf6333 \  
  --description "my efa" \  
  --groups sg-02e57dbcfe0331c1b
```

Salida:

```
{  
  "NetworkInterface": {  
    "AvailabilityZone": "us-west-2a",  
    "Description": "my efa",  
    "Groups": [  
      {  
        "GroupName": "my-efa-sg",  
        "GroupId": "sg-02e57dbcfe0331c1b"  
      }  
    ],  
    "InterfaceType": "efa",  
    "Ipv6Addresses": [],  
    "MacAddress": "06:d7:a4:f7:4d:57",  
    "NetworkInterfaceId": "eni-034acc2885e862b65",  
    "OwnerId": "123456789012",  
    "PrivateDnsName": "ip-10-0-8-180.us-west-2.compute.internal",  
    "PrivateIpAddress": "10.0.8.180",  
    "PrivateIpAddresses": [  
      {  
        "Primary": true,  
        "PrivateDnsName": "ip-10-0-8-180.us-west-2.compute.internal",  
        "PrivateIpAddress": "10.0.8.180"  
      }  
    ],  
    "RequesterId": "AIDA4Z3Y7GSXTMEXAMPLE",  
    "RequesterManaged": false,  
    "SourceDestCheck": true,  
    "Status": "pending",  
    "SubnetId": "subnet-00a24d0d67acf6333",
```

```
    "TagSet": [],  
    "VpcId": "vpc-02723a0feeeb9d57b"  
  }  
}
```

Para obtener más información, consulte [Interfaces de red elásticas](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [CreateNetworkInterface](#) la Referencia de AWS CLI comandos.

## create-placement-group

En el siguiente ejemplo de código, se muestra cómo usar `create-placement-group`.

### AWS CLI

Para crear un grupo de ubicaciones

Este comando de ejemplo crea un grupo de ubicación con el nombre especificado.

Comando:

```
aws ec2 create-placement-group --group-name my-cluster --strategy cluster
```

Para crear un grupo de colocación de particiones

Este comando de ejemplo crea un grupo de colocación de particiones denominado HDFS-Group-A con cinco particiones.

Comando:

```
aws ec2 create-placement-group --group-name HDFS-Group-A --strategy partition --  
partition-count 5
```

- Para API obtener más información, consulte [CreatePlacementGroup](#) la Referencia de AWS CLI comandos.

## create-replace-root-volume-task

En el siguiente ejemplo de código, se muestra cómo usar `create-replace-root-volume-task`.



## AWS CLI

Ejemplo 1: Para restaurar un volumen raíz a su estado de lanzamiento inicial

El siguiente `create-replace-root-volume-task` ejemplo restaura el volumen raíz de la instancia `i-0123456789abcdefa` a su estado de lanzamiento inicial.

```
aws ec2 create-replace-root-volume-task \  
  --instance-id i-0123456789abcdefa
```

Salida:

```
{  
  "ReplaceRootVolumeTask":  
  {  
    "InstanceId": "i-0123456789abcdefa",  
    "ReplaceRootVolumeTaskId": "replacevol-0111122223333abcd",  
    "TaskState": "pending",  
    "StartTime": "2022-03-14T15:06:38Z",  
    "Tags": []  
  }  
}
```

Para obtener más información, consulte [Reemplazar un volumen raíz](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

Ejemplo 2: Para restaurar un volumen raíz en una instantánea específica

El siguiente `create-replace-root-volume-task` ejemplo restaura el volumen raíz de la instancia `i-0123456789abcdefa` a la instantánea `snap-0abcdef1234567890`.

```
aws ec2 create-replace-root-volume-task \  
  --instance-id i-0123456789abcdefa \  
  --snapshot-id snap-0abcdef1234567890
```

Salida:

```
{  
  "ReplaceRootVolumeTask":  
  {  
    "InstanceId": "i-0123456789abcdefa",
```

```
    "ReplaceRootVolumeTaskId": "replacevol-0555566667777abcd",
    "TaskState": "pending",
    "StartTime": "2022-03-14T15:16:28Z",
    "Tags": []
  }
}
```

Para obtener más información, consulte [Reemplazar un volumen raíz](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte [CreateReplaceRootVolumeTask](#) la Referencia de AWS CLI comandos.

## create-reserved-instances-listing

En el siguiente ejemplo de código, se muestra cómo usar `create-reserved-instances-listing`.

### AWS CLI

Para incluir una instancia reservada en el Marketplace de instancias reservadas

En el siguiente `create-reserved-instances-listing` ejemplo, se crea un listado para la instancia reservada especificada en el Marketplace de instancias reservadas.

```
aws ec2 create-reserved-instances-listing \
  --reserved-instances-id 5ec28771-05ff-4b9b-aa31-9e57dexample \
  --instance-count 3 \
  --price-schedules CurrencyCode=USD,Price=25.50 \
  --client-token 550e8400-e29b-41d4-a716-446655440000
```

- Para API obtener más información, consulte [CreateReservedInstancesListing](#) la Referencia de AWS CLI comandos.

## create-restore-image-task

En el siguiente ejemplo de código, se muestra cómo usar `create-restore-image-task`.

### AWS CLI

Para restaurar un AMI desde un bucket de S3

El siguiente `create-restore-image-task` ejemplo restaura un AMI desde un bucket de S3. Utilice los valores `S3ObjectKey` y `Bucket` de la `describe-store-image-tasks` salida, especifique la clave de objeto AMI y el nombre del bucket de S3 en el que AMI se copió y especifique el nombre del objeto restaurado AMI. El nombre debe ser único AMIs en la región de esta cuenta. La persona restaurada AMI recibirá una nueva AMI identificación.

```
aws ec2 create-restore-image-task \  
  --object-key ami-1234567890abcdef0.bin \  
  --bucket my-ami-bucket \  
  --name "New AMI Name"
```

Salida:

```
{  
  "ImageId": "ami-0eab20fe36f83e1a8"  
}
```

Para obtener más información sobre el almacenamiento y la restauración y el AMI uso de S3, consulte Almacenar y restaurar un AMI uso de S3 <<https://docs.aws.amazon.com/AWS EC2/latest/UserGuide/ami-store-restore.html>> en la Guía del usuario de Amazon. EC2

- Para obtener API más información, consulte la Referencia de comandos.

[CreateRestoreImageTask](#) AWS CLI

## create-route-table

En el siguiente ejemplo de código, se muestra cómo usar `create-route-table`.

AWS CLI

Crear una tabla de enrutamiento

En este ejemplo, se crea una tabla de rutas para lo especificado VPC.

Comando:

```
aws ec2 create-route-table --vpc-id vpc-a01106c2
```

Salida:

```
{
```

```

"RouteTable": {
  "Associations": [],
  "RouteTableId": "rtb-22574640",
  "VpcId": "vpc-a01106c2",
  "PropagatingVgws": [],
  "Tags": [],
  "Routes": [
    {
      "GatewayId": "local",
      "DestinationCidrBlock": "10.0.0.0/16",
      "State": "active"
    }
  ]
}
}

```

- Para API obtener más información, consulte [CreateRouteTable](#) la Referencia de AWS CLI comandos.

## create-route

En el siguiente ejemplo de código, se muestra cómo usar create-route.

### AWS CLI

Para crear una ruta

En este ejemplo, se crea una ruta para la tabla de rutas especificada. La ruta coincide con todo el IPv4 tráfico (0.0.0.0/0) y lo enruta a la puerta de enlace de Internet especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 create-route --route-table-id rtb-22574640 --destination-cidr-block 0.0.0.0/0 --gateway-id igw-c0a643a9
```

Este comando de ejemplo crea una ruta en la tabla de rutas rtb-g8ff4ea2. La ruta coincide con el tráfico del IPv4 CIDR bloque 10.0.0.0/16 y lo enruta a la conexión de emparejamiento, pcx-111aaa22. VPC Esta ruta permite que el tráfico se dirija al par en la conexión de emparejamiento. VPC VPC Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 create-route --route-table-id rtb-g8ff4ea2 --destination-cidr-block 10.0.0.0/16 --vpc-peering-connection-id pcx-1a2b3c4d
```

En este ejemplo, se crea una ruta en la tabla de rutas especificada que coincide con todo el IPv6 tráfico (: : /0) y la enruta a la puerta de enlace de Internet de solo salida especificada.

Comando:

```
aws ec2 create-route --route-table-id rtb-dce620b8 --destination-ipv6-cidr-block ::/0 --egress-only-internet-gateway-id eigw-01eadbd45ecd7943f
```

- Para API obtener más información, consulte la Referencia de [CreateRoute](#) comandos AWS CLI .

## create-security-group

En el siguiente ejemplo de código, se muestra cómo usar create-security-group.

AWS CLI

Para crear un grupo de seguridad para EC2 -Classic

En este ejemplo, se crea un grupo de seguridad denominado MySecurityGroup.

Comando:

```
aws ec2 create-security-group --group-name MySecurityGroup --description "My security group"
```

Salida:

```
{  
  "GroupId": "sg-903004f8"  
}
```

Para crear un grupo de seguridad para - EC2 VPC

En este ejemplo, se crea un grupo de seguridad con el nombre MySecurityGroup especificadoVPC.

Comando:

```
aws ec2 create-security-group --group-name MySecurityGroup --description "My security group" --vpc-id vpc-1a2b3c4d
```

Salida:

```
{  
  "GroupId": "sg-903004f8"  
}
```

Para obtener más información, consulte [Uso de los grupos de seguridad](#) en la Guía del usuario de la Interfaz de la línea de comandos de AWS .

- Para API obtener más información, consulte [CreateSecurityGroup](#) la Referencia de AWS CLI comandos.

## create-snapshot

En el siguiente ejemplo de código, se muestra cómo usar create-snapshot.

AWS CLI

Para crear una instantánea

Este comando de ejemplo crea una instantánea del volumen con un identificador de volumen `vol-1234567890abcdef0` y una breve descripción para identificar la instantánea.

Comando:

```
aws ec2 create-snapshot --volume-id vol-1234567890abcdef0 --description "This is my root volume snapshot"
```

Salida:

```
{  
  "Description": "This is my root volume snapshot",  
  "Tags": [],  
  "Encrypted": false,  
  "VolumeId": "vol-1234567890abcdef0",  
}
```

```
"State": "pending",
"VolumeSize": 8,
"StartTime": "2018-02-28T21:06:01.000Z",
"Progress": "",
"OwnerId": "012345678910",
"SnapshotId": "snap-066877671789bd71b"
}
```

Para crear una instantánea con etiquetas

Este comando de ejemplo crea una instantánea y aplica dos etiquetas: `purpose=prod` y `costcenter=123`.

Comando:

```
aws ec2 create-snapshot --volume-id vol-1234567890abcdef0 --description 'Prod backup' --tag-specifications 'ResourceType=snapshot,Tags=[{Key=purpose,Value=prod},{Key=costcenter,Value=123}]'
```

Salida:

```
{
  "Description": "Prod backup",
  "Tags": [
    {
      "Value": "prod",
      "Key": "purpose"
    },
    {
      "Value": "123",
      "Key": "costcenter"
    }
  ],
  "Encrypted": false,
  "VolumeId": "vol-1234567890abcdef0",
  "State": "pending",
  "VolumeSize": 8,
  "StartTime": "2018-02-28T21:06:06.000Z",
  "Progress": "",
  "OwnerId": "012345678910",
  "SnapshotId": "snap-09ed24a70bc19bbe4"
}
```

- Para API obtener más información, consulte la Referencia de comandos. [CreateSnapshot](#) AWS CLI

## create-snapshots

En el siguiente ejemplo de código, se muestra cómo usar create-snapshots.

### AWS CLI

Ejemplo 1: Para crear una instantánea de varios volúmenes

En el siguiente create-snapshots ejemplo, se crean instantáneas de todos los volúmenes adjuntos a la instancia especificada.

```
aws ec2 create-snapshots \  
  --instance-specification InstanceId=i-1234567890abcdef0 \  
  --description "This is snapshot of a volume from my-instance"
```

Salida:

```
{  
  "Snapshots": [  
    {  
      "Description": "This is a snapshot of a volume from my-instance",  
      "Tags": [],  
      "Encrypted": false,  
      "VolumeId": "vol-0a01d2d5a34697479",  
      "State": "pending",  
      "VolumeSize": 16,  
      "StartTime": "2019-08-05T16:58:19.000Z",  
      "Progress": "",  
      "OwnerId": "123456789012",  
      "SnapshotId": "snap-07f30e3909aa0045e"  
    },  
    {  
      "Description": "This is a snapshot of a volume from my-instance",  
      "Tags": [],  
      "Encrypted": false,  
      "VolumeId": "vol-02d0d4947008cb1a2",  
      "State": "pending",  
      "VolumeSize": 20,  
      "StartTime": "2019-08-05T16:58:19.000Z",
```



```

        "Progress": "",
        "OwnerId": "123456789012",
        "SnapshotId": "snap-0ec20b602264aad48"
    },
    ...
]
}

```

Ejemplo 2: Para crear una instantánea de varios volúmenes con etiquetas del volumen de origen

En el siguiente `create-snapshots` ejemplo, se crean instantáneas de todos los volúmenes adjuntos a la instancia especificada y se copian las etiquetas de cada volumen en la instantánea correspondiente.

```

aws ec2 create-snapshots \
  --instance-specification InstanceId=i-1234567890abcdef0 \
  --copy-tags-from-source volume \
  --description "This is snapshot of a volume from my-instance"

```

Salida:

```

{
  "Snapshots": [
    {
      "Description": "This is a snapshot of a volume from my-instance",
      "Tags": [
        {
          "Key": "Name",
          "Value": "my-volume"
        }
      ],
      "Encrypted": false,
      "VolumeId": "vol-02d0d4947008cb1a2",
      "State": "pending",
      "VolumeSize": 20,
      "StartTime": "2019-08-05T16:53:04.000Z",
      "Progress": "",
      "OwnerId": "123456789012",
      "SnapshotId": "snap-053bfaeb821a458dd"
    }
    ...
  ]
}

```

```
}
```

Ejemplo 3: Para crear una instantánea de varios volúmenes que no incluya el volumen raíz

En el siguiente `create-snapshots` ejemplo, se crea una instantánea de todos los volúmenes adjuntos a la instancia especificada, excepto el volumen raíz.

```
aws ec2 create-snapshots \  
  --instance-specification InstanceId=i-1234567890abcdef0,ExcludeBootVolume=true
```

Consulte el ejemplo 1 para ver una salida de muestra.

Ejemplo 4: Para crear una instantánea de varios volúmenes y añadir etiquetas

En el siguiente `create-snapshots` ejemplo, se crean instantáneas de todos los volúmenes adjuntos a la instancia especificada y se añaden dos etiquetas a cada instantánea.

```
aws ec2 create-snapshots \  
  --instance-specification InstanceId=i-1234567890abcdef0 \  
  --tag-specifications 'ResourceType=snapshot,Tags=[{Key=Name,Value=backup},  
{Key=costcenter,Value=123}]'
```

Consulte el ejemplo 1 para ver una salida de muestra.

- Para API obtener más información, consulte [CreateSnapshots](#) la Referencia de AWS CLI comandos.

## **create-spot-datafeed-subscription**

En el siguiente ejemplo de código, se muestra cómo usar `create-spot-datafeed-subscription`.

AWS CLI

Para crear una fuente de datos de instancias puntuales

El siguiente `create-spot-datafeed-subscription` ejemplo crea una fuente de datos de instancias puntuales.

```
aws ec2 create-spot-datafeed-subscription \  
  --bucket my-bucket \  
  --instance-id i-1234567890abcdef0
```

```
--prefix spot-data-feed
```

Salida:

```
{
  "SpotDatafeedSubscription": {
    "Bucket": "my-bucket",
    "OwnerId": "123456789012",
    "Prefix": "spot-data-feed",
    "State": "Active"
  }
}
```

La fuente de datos se almacena en el bucket de Amazon S3 que especificó. Los nombres de los archivos de esta fuente de datos tienen el siguiente formato.

```
my-bucket.s3.amazonaws.com/spot-data-feed/123456789012.YYYY-MM-DD-HH.n.abcd1234.gz
```

Para obtener más información, consulte la [fuente de datos de instancias puntuales](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [CreateSpotDatafeedSubscription](#) la Referencia de AWS CLI comandos.

## create-store-image-task

En el siguiente ejemplo de código, se muestra cómo usar `create-store-image-task`.

AWS CLI

Para almacenar un AMI en un bucket de S3

El siguiente `create-store-image-task` ejemplo almacena un elemento AMI en un bucket de S3. Especifique el ID AMI y el nombre del depósito de S3 en el que se va a almacenar el AMI.

```
aws ec2 create-store-image-task \
  --image-id ami-1234567890abcdef0 \
  --bucket my-ami-bucket
```

Salida:

```
{
  "ObjectKey": "ami-1234567890abcdef0.bin"
}
```

Para obtener más información, consulte [Almacenar y restaurar y AMI usar S3](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [CreateStoreImageTask](#) la Referencia de AWS CLI comandos.

## create-subnet-cidr-reservation

En el siguiente ejemplo de código, se muestra cómo usar `create-subnet-cidr-reservation`.

### AWS CLI

Para crear una reserva de subred CIDR

En el siguiente `create-subnet-cidr-reservation` ejemplo, se crea una CIDR reserva de subred para la subred y el rango especificados. CIDR

```
aws ec2 create-subnet-cidr-reservation \
  --subnet-id subnet-03c51e2eEXAMPLE \
  --reservation-type prefix \
  --cidr 10.1.0.20/26
```

Salida:

```
{
  "SubnetCidrReservation": {
    "SubnetCidrReservationId": "scr-044f977c4eEXAMPLE",
    "SubnetId": "subnet-03c51e2e6cEXAMPLE",
    "Cidr": "10.1.0.16/28",
    "ReservationType": "prefix",
    "OwnerId": "123456789012"
  }
}
```

Para obtener más información, consulta [CIDRReservas de subred](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [CreateSubnetCidrReservation](#) la Referencia de AWS CLI comandos.

## create-subnet

En el siguiente ejemplo de código, se muestra cómo usar create-subnet.

### AWS CLI

Ejemplo 1: Para crear una subred solo con un IPv4 CIDR bloque

El siguiente create-subnet ejemplo crea una subred en el bloque especificado VPC con el bloque especificado IPv4CIDR.

```
aws ec2 create-subnet \  
  --vpc-id vpc-081ec835f3EXAMPLE \  
  --cidr-block 10.0.0.0/24 \  
  --tag-specifications ResourceType=subnet,Tags=[{Key=Name,Value=my-ipv4-only-  
subnet}]
```

Salida:

```
{  
  "Subnet": {  
    "AvailabilityZone": "us-west-2a",  
    "AvailabilityZoneId": "usw2-az2",  
    "AvailableIpAddressCount": 251,  
    "CidrBlock": "10.0.0.0/24",  
    "DefaultForAz": false,  
    "MapPublicIpOnLaunch": false,  
    "State": "available",  
    "SubnetId": "subnet-0e99b93155EXAMPLE",  
    "VpcId": "vpc-081ec835f3EXAMPLE",  
    "OwnerId": "123456789012",  
    "AssignIpv6AddressOnCreation": false,  
    "Ipv6CidrBlockAssociationSet": [],  
    "Tags": [  
      {  
        "Key": "Name",  
        "Value": "my-ipv4-only-subnet"  
      }  
    ],  
  },  
}
```

```

    "SubnetArn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0e99b93155EXAMPLE"
  }
}

```

Ejemplo 2: Para crear una subred con bloques y IPv4 IPv6 CIDR

En el siguiente `create-subnet` ejemplo, se crea una subred en el espacio especificado VPC con los bloques IPv4 y IPv6 CIDR especificados.

```

aws ec2 create-subnet \
  --vpc-id vpc-081ec835f3EXAMPLE \
  --cidr-block 10.0.0.0/24 \
  --ipv6-cidr-block 2600:1f16:cfe:3660::/64 \
  --tag-specifications ResourceType=subnet,Tags=[{Key=Name,Value=my-ipv4-ipv6-
subnet}]

```

Salida:

```

{
  "Subnet": {
    "AvailabilityZone": "us-west-2a",
    "AvailabilityZoneId": "usw2-az2",
    "AvailableIpAddressCount": 251,
    "CidrBlock": "10.0.0.0/24",
    "DefaultForAz": false,
    "MapPublicIpOnLaunch": false,
    "State": "available",
    "SubnetId": "subnet-0736441d38EXAMPLE",
    "VpcId": "vpc-081ec835f3EXAMPLE",
    "OwnerId": "123456789012",
    "AssignIpv6AddressOnCreation": false,
    "Ipv6CidrBlockAssociationSet": [
      {
        "AssociationId": "subnet-cidr-assoc-06c5f904499fcc623",
        "Ipv6CidrBlock": "2600:1f13:cfe:3660::/64",
        "Ipv6CidrBlockState": {
          "State": "associating"
        }
      }
    ],
    "Tags": [
      {

```

```

        "Key": "Name",
        "Value": "my-ipv4-ipv6-subnet"
    }
],
"SubnetArn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0736441d38EXAMPLE"
}
}

```

Ejemplo 3: Para crear una subred solo con un bloque IPv6 CIDR

El siguiente `create-subnet` ejemplo crea una subred en el bloque especificado VPC con el bloque especificado IPv6CIDR.

```

aws ec2 create-subnet \
  --vpc-id vpc-081ec835f3EXAMPLE \
  --ipv6-native \
  --ipv6-cidr-block 2600:1f16:115:200::/64 \
  --tag-specifications ResourceType=subnet,Tags=[{Key=Name,Value=my-ipv6-only-
subnet}]

```

Salida:

```

{
  "Subnet": {
    "AvailabilityZone": "us-west-2a",
    "AvailabilityZoneId": "usw2-az2",
    "AvailableIpAddressCount": 0,
    "DefaultForAz": false,
    "MapPublicIpOnLaunch": false,
    "State": "available",
    "SubnetId": "subnet-03f720e7deEXAMPLE",
    "VpcId": "vpc-081ec835f3EXAMPLE",
    "OwnerId": "123456789012",
    "AssignIpv6AddressOnCreation": true,
    "Ipv6CidrBlockAssociationSet": [
      {
        "AssociationId": "subnet-cidr-assoc-01ef639edde556709",
        "Ipv6CidrBlock": "2600:1f13:cfe:3660::/64",
        "Ipv6CidrBlockState": {
          "State": "associating"
        }
      }
    ]
  }
}

```

```
    ],
    "Tags": [
      {
        "Key": "Name",
        "Value": "my-ipv6-only-subnet"
      }
    ],
    "SubnetArn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-03f720e7deEXAMPLE"
  }
}
```

Para obtener más información, consulte [VPCs y las subredes](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [CreateSubnet](#) la Referencia de AWS CLI comandos.

## create-tags

En el siguiente ejemplo de código, se muestra cómo usar `create-tags`.

### AWS CLI

Ejemplo 1: Para añadir una etiqueta a un recurso

En el siguiente `create-tags` ejemplo, se agrega la etiqueta `Stack=production` a la imagen especificada o se sobrescribe una etiqueta existente en el AMI lugar donde se encuentra `Stack` la clave de la etiqueta.

```
aws ec2 create-tags \
  --resources ami-1234567890abcdef0 \
  --tags Key=Stack,Value=production
```

Para obtener más información, consulte [Este es el título del tema](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

Ejemplo 2: Para añadir etiquetas a varios recursos

El siguiente `create-tags` ejemplo agrega (o sobrescribe) dos etiquetas para una instancia AMI y una. Una de las etiquetas tiene una clave (`webserver`), pero no tiene valor (el valor



se establece en una cadena vacía). La otra etiqueta tiene una clave (stack) y un valor (Production).

```
aws ec2 create-tags \  
  --resources ami-1a2b3c4d i-1234567890abcdef0 \  
  --tags Key=webserver,Value= Key=stack,Value=Production
```

Para obtener más información, consulte [Este es el título del tema](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

Ejemplo 3: Para añadir etiquetas que contengan caracteres especiales

En el siguiente ejemplo de `create-tags`, se agrega la etiqueta `[Group]=test` a una instancia. Los corchetes (`[ y ]`) son caracteres especiales y deben incluirse en el carácter de escape. En los siguientes ejemplos también se usa el carácter de continuación de línea adecuado para cada entorno.

Si usa Windows, encierre el elemento que tiene caracteres especiales entre comillas dobles (`"`) y, a continuación, preceda cada carácter de comillas dobles con una barra invertida (`\`) de la siguiente manera:

```
aws ec2 create-tags ^  
  --resources i-1234567890abcdef0 ^  
  --tags Key=\"[Group]\",Value=test
```

Si utiliza Windows PowerShell, escriba el elemento con caracteres especiales entre comillas dobles (`«`), coloque una barra invertida (`\`) delante de cada carácter entre comillas dobles y, a continuación, rodee toda la estructura de claves y valores entre comillas simples (`'`), de la siguiente manera:

```
aws ec2 create-tags `\  
  --resources i-1234567890abcdef0 `\  
  --tags 'Key=\"[Group]\",Value=test'
```

Si usa Linux u OS X, encierre el elemento con caracteres especiales entre comillas dobles (`"`) y, a continuación, encierre toda la estructura de clave y valor entre comillas simples (`'`) de la siguiente manera:

```
aws ec2 create-tags \  
  --resources i-1234567890abcdef0 \  
  --tags 'Key=\"[Group]\",Value=test'
```

```
--tags 'Key="[Group]",Value=test'
```

Para obtener más información, consulte [Este es el título del tema](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [CreateTags](#) la Referencia de AWS CLI comandos.

## create-traffic-mirror-filter-rule

En el siguiente ejemplo de código, se muestra cómo usar `create-traffic-mirror-filter-rule`.

### AWS CLI

Para crear una regla de filtrado para el TCP tráfico entrante

En el siguiente `create-traffic-mirror-filter-rule` ejemplo, se crea una regla que se puede utilizar para reflejar todo TCP el tráfico entrante. Antes de ejecutar este comando, utilícelo `create-traffic-mirror-filter` para crear el filtro Traffic Mirror.

```
aws ec2 create-traffic-mirror-filter-rule \  
  --description "TCP Rule" \  
  --destination-cidr-block 0.0.0.0/0 \  
  --protocol 6 \  
  --rule-action accept \  
  --rule-number 1 \  
  --source-cidr-block 0.0.0.0/0 \  
  --traffic-direction ingress \  
  --traffic-mirror-filter-id tmf-04812ff784b25ae67
```

Salida:

```
{  
  "TrafficMirrorFilterRule": {  
    "DestinationCidrBlock": "0.0.0.0/0",  
    "TrafficMirrorFilterId": "tmf-04812ff784b25ae67",  
    "TrafficMirrorFilterRuleId": "tmfr-02d20d996673f3732",  
    "SourceCidrBlock": "0.0.0.0/0",  
    "TrafficDirection": "ingress",  
    "Description": "TCP Rule",  
    "RuleNumber": 1,  
    "RuleAction": "accept",
```

```
    "Protocol": 6
  },
  "ClientToken": "4752b573-40a6-4eac-a8a4-a72058761219"
}
```

Para obtener más información, consulte [Crear un filtro de espejo de tráfico](#) en la Guía de duplicación de AWS tráfico.

- Para API obtener más información, consulte la Referencia [CreateTrafficMirrorFilterRule](#) de AWS CLI comandos.

## create-traffic-mirror-filter

En el siguiente ejemplo de código, se muestra cómo usar `create-traffic-mirror-filter`.

### AWS CLI

Para crear un filtro de espejo de tráfico

El siguiente `create-traffic-mirror-filter` ejemplo crea un filtro Traffic Mirror. Tras crear el filtro, `create-traffic-mirror-filter-rule` utilícelo para añadir reglas al filtro.

```
aws ec2 create-traffic-mirror-filter \
  --description "TCP Filter"
```

Salida:

```
{
  "ClientToken": "28908518-100b-4987-8233-8c744EXAMPLE",
  "TrafficMirrorFilter": {
    "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
    "Description": "TCP Filter",
    "EgressFilterRules": [],
    "IngressFilterRules": [],
    "Tags": [],
    "NetworkServices": []
  }
}
```

Para obtener más información, consulte [Crear un filtro de espejo de tráfico](#) en la Guía de duplicación de AWS tráfico.

- Para API obtener más información, consulte la Referencia [CreateTrafficMirrorFilter](#) de AWS CLI comandos.

## create-traffic-mirror-session

En el siguiente ejemplo de código, se muestra cómo usar `create-traffic-mirror-session`.

### AWS CLI

Para crear una sesión de Traffic Mirror

El siguiente `create-traffic-mirror-session` comando crea una sesión de réplica de tráfico para el origen y el destino especificados para 25 bytes del paquete.

```
aws ec2 create-traffic-mirror-session \  
  --description "example session" \  
  --traffic-mirror-target-id tmt-07f75d8feeEXAMPLE \  
  --network-interface-id eni-070203f901EXAMPLE \  
  --session-number 1 \  
  --packet-length 25 \  
  --traffic-mirror-filter-id tmf-04812ff784EXAMPLE
```

Salida:

```
{  
  "TrafficMirrorSession": {  
    "TrafficMirrorSessionId": "tms-08a33b1214EXAMPLE",  
    "TrafficMirrorTargetId": "tmt-07f75d8feeEXAMPLE",  
    "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",  
    "NetworkInterfaceId": "eni-070203f901EXAMPLE",  
    "OwnerId": "111122223333",  
    "PacketLength": 25,  
    "SessionNumber": 1,  
    "VirtualNetworkId": 7159709,  
    "Description": "example session",  
    "Tags": []  
  },  
  "ClientToken": "5236cffc-ee13-4a32-bb5b-388d9da09d96"  
}
```

Para obtener más información, consulte [Crear una sesión de duplicación de tráfico](#) en la Guía de duplicación de AWS tráfico.

- Para API obtener más información, consulte la Referencia [CreateTrafficMirrorSession](#) de AWS CLI comandos.

## create-traffic-mirror-target

En el siguiente ejemplo de código, se muestra cómo usar `create-traffic-mirror-target`.

### AWS CLI

Para crear un objetivo de réplica de tráfico de Network Load Balancer

En el siguiente `create-traffic-mirror-target` ejemplo, se crea un objetivo de réplica de tráfico de Network Load Balancer.

```
aws ec2 create-traffic-mirror-target \  
  --description "Example Network Load Balancer Target" \  
  --network-load-balancer-arn arn:aws:elasticloadbalancing:us-  
east-1:111122223333:loadbalancer/net/NLB/7cdec873EXAMPLE
```

Salida:

```
{  
  "TrafficMirrorTarget": {  
    "Type": "network-load-balancer",  
    "Tags": [],  
    "Description": "Example Network Load Balancer Target",  
    "OwnerId": "111122223333",  
    "NetworkLoadBalancerArn": "arn:aws:elasticloadbalancing:us-  
east-1:724145273726:loadbalancer/net/NLB/7cdec873EXAMPLE",  
    "TrafficMirrorTargetId": "tmt-0dabe9b0a6EXAMPLE"  
  },  
  "ClientToken": "d5c090f5-8a0f-49c7-8281-72c796a21f72"  
}
```

Para crear un objetivo de Traffic Mirror de red

El siguiente `create-traffic-mirror-target` ejemplo crea un destino de Traffic Mirror de interfaz de red.

```
aws ec2 create-traffic-mirror-target --description «Destino de interfaz de red» -- eni-eni-01f6f631e  
network-interface-id EXAMPLE
```

Salida:

```
{
  "ClientToken": "5289a345-0358-4e62-93d5-47ef3061d65e",
  "TrafficMirrorTarget": {
    "Description": "Network interface target",
    "NetworkInterfaceId": "eni-01f6f631eEXAMPLE",
    "TrafficMirrorTargetId": "tmt-02dcdb2abEXAMPLE",
    "OwnerId": "111122223333",
    "Type": "network-interface",
    "Tags": []
  }
}
```

[Para obtener más información, consulte Crear un objetivo de duplicación de tráfico en la Guía de duplicación de tráfico.AWS](#)

- Para API obtener más información, consulte la Referencia [CreateTrafficMirrorTarget](#) de AWS CLI comandos.

## create-transit-gateway-connect-peer

En el siguiente ejemplo de código, se muestra cómo usar `create-transit-gateway-connect-peer`.

AWS CLI

Para crear un par de Transit Gateway Connect

En el siguiente `create-transit-gateway-connect-peer` ejemplo, se crea un peer de Connect.

```
aws ec2 create-transit-gateway-connect-peer \
  --transit-gateway-attachment-id tgw-attach-0f0927767cEXAMPLE \
  --peer-address 172.31.1.11 \
  --inside-cidr-blocks 169.254.6.0/29
```

Salida:

```
{
  "TransitGatewayConnectPeer": {
    "TransitGatewayAttachmentId": "tgw-attach-0f0927767cEXAMPLE",
```

```

"TransitGatewayConnectPeerId": "tgw-connect-peer-0666adbac4EXAMPLE",
"State": "pending",
"CreationTime": "2021-10-13T03:35:17.000Z",
"ConnectPeerConfiguration": {
  "TransitGatewayAddress": "10.0.0.234",
  "PeerAddress": "172.31.1.11",
  "InsideCidrBlocks": [
    "169.254.6.0/29"
  ],
  "Protocol": "gre",
  "BgpConfigurations": [
    {
      "TransitGatewayAsn": 64512,
      "PeerAsn": 64512,
      "TransitGatewayAddress": "169.254.6.2",
      "PeerAddress": "169.254.6.1",
      "BgpStatus": "down"
    },
    {
      "TransitGatewayAsn": 64512,
      "PeerAsn": 64512,
      "TransitGatewayAddress": "169.254.6.3",
      "PeerAddress": "169.254.6.1",
      "BgpStatus": "down"
    }
  ]
}
}
}
}

```

Para obtener más información, consulte los [archivos adjuntos de Transit Gateway Connect y los pares de Transit Gateway Connect](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [CreateTransitGatewayConnectPeer](#) la Referencia de AWS CLI comandos.

## create-transit-gateway-connect

En el siguiente ejemplo de código, se muestra cómo usar create-transit-gateway-connect.

### AWS CLI

Para crear un adjunto de Transit Gateway Connect

En el siguiente `create-transit-gateway-connect` ejemplo, se crea un adjunto Connect, con el protocolo «gre», para el adjunto especificado.

```
aws ec2 create-transit-gateway-connect \  
  --transport-transit-gateway-attachment-id tgw-attach-0a89069f57EXAMPLE \  
  --options "Protocol=gre"
```

Salida:

```
{  
  "TransitGatewayConnect": {  
    "TransitGatewayAttachmentId": "tgw-attach-037012e5dcEXAMPLE",  
    "TransportTransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",  
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",  
    "State": "pending",  
    "CreationTime": "2021-03-09T19:59:17+00:00",  
    "Options": {  
      "Protocol": "gre"  
    }  
  }  
}
```

Para obtener más información, consulte los [archivos adjuntos de Transit Gateway Connect y los pares de Transit Gateway Connect](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [CreateTransitGatewayConnect](#) la Referencia de AWS CLI comandos.

## **create-transit-gateway-multicast-domain**

En el siguiente ejemplo de código, se muestra cómo usar `create-transit-gateway-multicast-domain`.

### AWS CLI

Ejemplo 1: Para crear un dominio de IGMP multidifusión

El siguiente `create-transit-gateway-multicast-domain` ejemplo crea un dominio de multidifusión para la puerta de enlace de tránsito especificada. Con las fuentes estáticas deshabilitadas, cualquier instancia de las subredes asociadas al dominio de multidifusión puede



enviar tráfico de multidifusión. Si al menos un miembro usa el IGMP protocolo, debes habilitar la compatibilidad. IGMPv2

```
aws ec2 create-transit-gateway-multicast-domain \
  --transit-gateway-id tgw-0bf0bffefaEXAMPLE \
  --options StaticSourcesSupport=disable,Igmpv2Support=enable
```

Salida:

```
{
  "TransitGatewayMulticastDomain": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c9e29e2a7EXAMPLE",
    "TransitGatewayId": "tgw-0bf0bffefaEXAMPLE",
    "TransitGatewayMulticastDomainArn": "arn:aws:ec2:us-west-2:123456789012:transit-gateway-multicast-domain/tgw-mcast-domain-0c9e29e2a7EXAMPLE",
    "OwnerId": "123456789012",
    "Options": {
      "Igmpv2Support": "enable",
      "StaticSourcesSupport": "disable",
      "AutoAcceptSharedAssociations": "disable"
    },
    "State": "pending",
    "CreationTime": "2021-09-29T22:17:13.000Z"
  }
}
```

Ejemplo 2: Para crear un dominio de multidifusión estático

El siguiente `create-transit-gateway-multicast-domain` ejemplo crea un dominio de multidifusión para la puerta de enlace de tránsito especificada. Con las fuentes estáticas habilitadas, debe agregar fuentes de forma estática.

```
aws ec2 create-transit-gateway-multicast-domain \
  --transit-gateway-id tgw-0bf0bffefaEXAMPLE \
  --options StaticSourcesSupport=enable,Igmpv2Support=disable
```

Salida:

```
{
  "TransitGatewayMulticastDomain": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-000fb24d04EXAMPLE",
```

```

    "TransitGatewayId": "tgw-0bf0bfffefaEXAMPLE",
    "TransitGatewayMulticastDomainArn": "arn:aws:ec2:us-
west-2:123456789012:transit-gateway-multicast-domain/tgw-mcast-
domain-000fb24d04EXAMPLE",
    "OwnerId": "123456789012",
    "Options": {
      "Icmpv2Support": "disable",
      "StaticSourcesSupport": "enable",
      "AutoAcceptSharedAssociations": "disable"
    },
    "State": "pending",
    "CreationTime": "2021-09-29T22:20:19.000Z"
  }
}

```

Para obtener más información, consulte [Administrar dominios de multidifusión](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [CreateTransitGatewayMulticastDomain](#) de AWS CLI comandos.

## create-transit-gateway-peering-attachment

En el siguiente ejemplo de código, se muestra cómo usar `create-transit-gateway-peering-attachment`.

### AWS CLI

Para crear un adjunto de interconexión de Transit Gateway

En el siguiente `create-transit-gateway-peering-attachment` ejemplo, se crea una solicitud de adjunto de interconexión entre las dos pasarelas de tránsito especificadas.

```

aws ec2 create-transit-gateway-peering-attachment \
  --transit-gateway-id tgw-123abc05e04123abc \
  --peer-transit-gateway-id tgw-11223344aabbcc112 \
  --peer-account-id 123456789012 \
  --peer-region us-east-2

```

Salida:

```
{
```

```

"TransitGatewayPeeringAttachment": {
  "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",
  "RequesterTgwInfo": {
    "TransitGatewayId": "tgw-123abc05e04123abc",
    "OwnerId": "123456789012",
    "Region": "us-west-2"
  },
  "AcceptorTgwInfo": {
    "TransitGatewayId": "tgw-11223344aabbcc112",
    "OwnerId": "123456789012",
    "Region": "us-east-2"
  },
  "State": "initiatingRequest",
  "CreationTime": "2019-12-09T11:38:05.000Z"
}
}

```

Para obtener más información, consulte los [adjuntos de peering de Transit Gateway](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [CreateTransitGatewayPeeringAttachment](#) de AWS CLI comandos.

## create-transit-gateway-policy-table

En el siguiente ejemplo de código, se muestra cómo usar `create-transit-gateway-policy-table`.

### AWS CLI

Para crear una tabla de políticas de pasarelas de tránsito

El siguiente `create-transit-gateway-policy-table` ejemplo crea una tabla de políticas de la puerta de enlace de tránsito para la puerta de enlace de tránsito especificada.

```

aws ec2 create-transit-gateway-policy-table \
  --transit-gateway-id tgw-067f8505c18f0bd6e

```

Salida:

```

{
  "TransitGatewayPolicyTable": {

```

```

    "TransitGatewayPolicyTableId": "tgw-ptb-0a16f134b78668a81",
    "TransitGatewayId": "tgw-067f8505c18f0bd6e",
    "State": "pending",
    "CreationTime": "2023-11-28T16:36:43+00:00"
  }
}

```

Para obtener más información, consulte [las tablas de políticas de Transit Gateway](#) en la Guía del usuario de Transit Gateway.

- Para API obtener más información, consulte [CreateTransitGatewayPolicyTable](#) la Referencia de AWS CLI comandos.

## create-transit-gateway-prefix-list-reference

En el siguiente ejemplo de código, se muestra cómo usar `create-transit-gateway-prefix-list-reference`.

### AWS CLI

Para crear una referencia a una lista de prefijos

El siguiente `create-transit-gateway-prefix-list-reference` ejemplo crea una referencia a la lista de prefijos especificada en la tabla de rutas de la puerta de enlace de tránsito especificada.

```

aws ec2 create-transit-gateway-prefix-list-reference \
  --transit-gateway-route-table-id tgw-rtb-0123456789abcd123 \
  --prefix-list-id pl-1111112222222333 \
  --transit-gateway-attachment-id tgw-attach-aaaaaabbbbb11111

```

Salida:

```

{
  "TransitGatewayPrefixListReference": {
    "TransitGatewayRouteTableId": "tgw-rtb-0123456789abcd123",
    "PrefixListId": "pl-1111112222222333",
    "PrefixListOwnerId": "123456789012",
    "State": "pending",
    "Blackhole": false,
    "TransitGatewayAttachment": {
      "TransitGatewayAttachmentId": "tgw-attach-aaaaaabbbbb11111",

```

```

        "ResourceType": "vpc",
        "ResourceId": "vpc-112233445566aabbcc"
    }
}

```

Para obtener más información, consulte las [referencias a la lista de prefijos](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [CreateTransitGatewayPrefixListReference](#) de AWS CLI comandos.

## create-transit-gateway-route-table

En el siguiente ejemplo de código, se muestra cómo usar `create-transit-gateway-route-table`.

### AWS CLI

Para crear una tabla de rutas de Transit Gateway

El siguiente `create-transit-gateway-route-table` ejemplo crea una tabla de rutas para la puerta de enlace de tránsito especificada.

```

aws ec2 create-transit-gateway-route-table \
  --transit-gateway-id tgw-0262a0e521EXAMPLE

```

Salida:

```

{
  "TransitGatewayRouteTable": {
    "TransitGatewayRouteTableId": "tgw-rtb-0960981be7EXAMPLE",
    "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
    "State": "pending",
    "DefaultAssociationRouteTable": false,
    "DefaultPropagationRouteTable": false,
    "CreationTime": "2019-07-10T19:01:46.000Z"
  }
}

```

Para obtener más información, consulte la [tabla de rutas de una pasarela de tránsito](#) en la Guía de pasarelas de tránsito.

- Para API obtener más información, consulte [CreateTransitGatewayRouteTable](#) la Referencia de AWS CLI comandos.

## create-transit-gateway-route

En el siguiente ejemplo de código, se muestra cómo usar `create-transit-gateway-route`.

### AWS CLI

Para crear una ruta de entrada de tránsito

El siguiente `create-transit-gateway-route` ejemplo crea una ruta, con el destino especificado, para la tabla de rutas especificada.

```
aws ec2 create-transit-gateway-route \  
  --destination-cidr-block 10.0.2.0/24 \  
  --transit-gateway-route-table-id tgw-rtb-0b6f6aaa01EXAMPLE \  
  --transit-gateway-attachment-id tgw-attach-0b5968d3b6EXAMPLE
```

Salida:

```
{  
  "Route": {  
    "DestinationCidrBlock": "10.0.2.0/24",  
    "TransitGatewayAttachments": [  
      {  
        "ResourceId": "vpc-0065acced4EXAMPLE",  
        "TransitGatewayAttachmentId": "tgw-attach-0b5968d3b6EXAMPLE",  
        "ResourceType": "vpc"  
      }  
    ],  
    "Type": "static",  
    "State": "active"  
  }  
}
```

Para obtener más información, consulte [las tablas de rutas de Transit Gateways](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [CreateTransitGatewayRoute](#) la Referencia de AWS CLI comandos.

## create-transit-gateway-vpc-attachment

En el siguiente ejemplo de código, se muestra cómo usar `create-transit-gateway-vpc-attachment`.

### AWS CLI

Ejemplo 1: Para asociar una puerta de enlace de tránsito a un VPC

El siguiente `create-transit-gateway-vpc-attachment` ejemplo crea una pasarela de tránsito adjunta a la especificada VPC.

```
aws ec2 create-transit-gateway-vpc-attachment \
  --transit-gateway-id tgw-0262a0e521EXAMPLE \
  --vpc-id vpc-07e8ffd50f49335df \
  --subnet-id subnet-0752213d59EXAMPLE
```

Salida:

```
{
  "TransitGatewayVpcAttachment": {
    "TransitGatewayAttachmentId": "tgw-attach-0a34fe6b4fEXAMPLE",
    "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
    "VpcId": "vpc-07e8ffd50fEXAMPLE",
    "VpcOwnerId": "111122223333",
    "State": "pending",
    "SubnetIds": [
      "subnet-0752213d59EXAMPLE"
    ],
    "CreationTime": "2019-07-10T17:33:46.000Z",
    "Options": {
      "DnsSupport": "enable",
      "Ipv6Support": "disable"
    }
  }
}
```

Para obtener más información, consulte [Crear una pasarela de transporte adjunta a una VPC](#) en la Guía de pasarelas de tránsito.

Ejemplo 2: Para asociar una puerta de enlace de tránsito a varias subredes en un VPC

En el siguiente `create-transit-gateway-vpc-attachment` ejemplo, se crea una puerta de enlace de tránsito adjunta a las subredes VPC y especificadas.

```
aws ec2 create-transit-gateway-vpc-attachment \  
  --transit-gateway-id tgw-02f776b1a7EXAMPLE \  
  --vpc-id vpc-3EXAMPLE \  
  --subnet-ids "subnet-dEXAMPLE" "subnet-6EXAMPLE"
```

Salida:

```
{  
  "TransitGatewayVpcAttachment": {  
    "TransitGatewayAttachmentId": "tgw-attach-0e141e0bebEXAMPLE",  
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",  
    "VpcId": "vpc-3EXAMPLE",  
    "VpcOwnerId": "111122223333",  
    "State": "pending",  
    "SubnetIds": [  
      "subnet-6EXAMPLE",  
      "subnet-dEXAMPLE"  
    ],  
    "CreationTime": "2019-12-17T20:07:52.000Z",  
    "Options": {  
      "DnsSupport": "enable",  
      "Ipv6Support": "disable"  
    }  
  }  
}
```

Para obtener más información, consulte [Crear una pasarela de tránsito adjunta a una VPC](#) en la Guía de pasarelas de tránsito.

- Para API obtener más información, consulte [CreateTransitGatewayVpcAttachment](#) la Referencia de AWS CLI comandos.

## **create-transit-gateway**

En el siguiente ejemplo de código, se muestra cómo usar `create-transit-gateway`.

### AWS CLI

Para crear una puerta de enlace de tránsito



El siguiente `create-transit-gateway` ejemplo crea una puerta de enlace de tránsito.

```
aws ec2 create-transit-gateway \
  --description MyTGW \
  --
options AmazonSideAsn=64516,AutoAcceptSharedAttachments=enable,DefaultRouteTableAssociation=
```

Salida:

```
{
  "TransitGateway": {
    "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
    "TransitGatewayArn": "arn:aws:ec2:us-east-2:111122223333:transit-gateway/tgw-0262a0e521EXAMPLE",
    "State": "pending",
    "OwnerId": "111122223333",
    "Description": "MyTGW",
    "CreationTime": "2019-07-10T14:02:12.000Z",
    "Options": {
      "AmazonSideAsn": 64516,
      "AutoAcceptSharedAttachments": "enable",
      "DefaultRouteTableAssociation": "enable",
      "AssociationDefaultRouteTableId": "tgw-rtb-018774adf3EXAMPLE",
      "DefaultRouteTablePropagation": "enable",
      "PropagationDefaultRouteTableId": "tgw-rtb-018774adf3EXAMPLE",
      "VpnEcmpSupport": "enable",
      "DnsSupport": "enable"
    }
  }
}
```

Para obtener más información, consulte [Crear una puerta de enlace de tránsito](#) en la Guía de pasarelas de tránsito.

- Para API obtener más información, consulte [CreateTransitGateway](#) la Referencia de AWS CLI comandos.

## **create-verified-access-endpoint**

En el siguiente ejemplo de código, se muestra cómo usar `create-verified-access-endpoint`.

## AWS CLI

Para crear un punto final de acceso verificado

En el siguiente `create-verified-access-endpoint` ejemplo, se crea un punto final de acceso verificado para el grupo de acceso verificado especificado. La interfaz de red y el grupo de seguridad especificados deben pertenecer al mismo VPC grupo.

```
aws ec2 create-verified-access-endpoint \
  --verified-access-group-id vagr-0dbe967baf14b7235 \
  --endpoint-type network-interface \
  --attachment-type vpc \
  --domain-certificate-arn arn:aws:acm:us-east-2:123456789012:certificate/  
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE \
  --application-domain example.com \
  --endpoint-domain-prefix my-ava-app \
  --security-group-ids sg-004915970c4c8f13a \
  --network-interface-  
options NetworkInterfaceId=eni-0aec70418c8d87a0f,Protocol=https,Port=443 \
  --tag-specifications ResourceType=verified-access-  
endpoint,Tags=[{Key=Name,Value=my-va-endpoint}]
```

Salida:

```
{
  "VerifiedAccessEndpoint": {
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
    "VerifiedAccessEndpointId": "vae-066fac616d4d546f2",
    "ApplicationDomain": "example.com",
    "EndpointType": "network-interface",
    "AttachmentType": "vpc",
    "DomainCertificateArn": "arn:aws:acm:us-east-2:123456789012:certificate/
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE",
    "EndpointDomain": "my-ava-
app.edge-00c3372d53b1540bb.vai-0ce000c0b7643abea.prod.verified-access.us-
east-2.amazonaws.com",
    "SecurityGroupIds": [
      "sg-004915970c4c8f13a"
    ],
    "NetworkInterfaceOptions": {
      "NetworkInterfaceId": "eni-0aec70418c8d87a0f",
      "Protocol": "https",
```

```

        "Port": 443
      },
      "Status": {
        "Code": "pending"
      },
      "Description": "",
      "CreationTime": "2023-08-25T20:54:43",
      "LastUpdatedTime": "2023-08-25T20:54:43",
      "Tags": [
        {
          "Key": "Name",
          "Value": "my-va-endpoint"
        }
      ]
    }
  ]
}

```

Para obtener más información, consulte los [puntos finales de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [CreateVerifiedAccessEndpoint](#) la Referencia de AWS CLI comandos.

## create-verified-access-group

En el siguiente ejemplo de código, se muestra cómo usar `create-verified-access-group`.

### AWS CLI

Para crear un grupo de acceso verificado

En el siguiente `create-verified-access-group` ejemplo, se crea un grupo de acceso verificado para la instancia de acceso verificado especificada.

```

aws ec2 create-verified-access-group \
  --verified-access-instance-id vai-0ce000c0b7643abea \
  --tag-specifications ResourceType=verified-access-
group,Tags=[{Key=Name,Value=my-va-group}]

```

Salida:

```
{
```

```

"VerifiedAccessGroup": {
  "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
  "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
  "Description": "",
  "Owner": "123456789012",
  "VerifiedAccessGroupArn": "arn:aws:ec2:us-east-2:123456789012:verified-
access-group/vagr-0dbe967baf14b7235",
  "CreationTime": "2023-08-25T19:55:19",
  "LastUpdatedTime": "2023-08-25T19:55:19",
  "Tags": [
    {
      "Key": "Name",
      "Value": "my-va-group"
    }
  ]
}
}

```

Para obtener más información, consulte los [grupos de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [CreateVerifiedAccessGroup](#) la Referencia de AWS CLI comandos.

## create-verified-access-instance

En el siguiente ejemplo de código, se muestra cómo usar `create-verified-access-instance`.

### AWS CLI

Para crear una instancia de acceso verificado

En el siguiente `create-verified-access-instance` ejemplo, se crea una instancia de acceso verificado con una etiqueta de nombre.

```

aws ec2 create-verified-access-instance \
  --tag-specifications ResourceType=verified-access-
instance,Tags=[{Key=Name,Value=my-va-instance}]

```

Salida:

```
{
```

```
"VerifiedAccessInstance": {
  "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
  "Description": "",
  "VerifiedAccessTrustProviders": [],
  "CreationTime": "2023-08-25T18:27:56",
  "LastUpdatedTime": "2023-08-25T18:27:56",
  "Tags": [
    {
      "Key": "Name",
      "Value": "my-va-instance"
    }
  ]
}
```

Para obtener más información, consulte las [instancias de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [CreateVerifiedAccessInstance](#) la Referencia de AWS CLI comandos.

## create-verified-access-trust-provider

En el siguiente ejemplo de código, se muestra cómo usar `create-verified-access-trust-provider`.

### AWS CLI

Para crear un proveedor de confianza de acceso verificado

En el siguiente `create-verified-access-trust-provider` ejemplo, se configura un proveedor de confianza de acceso verificado mediante AWS Identity Center.

```
aws ec2 create-verified-access-trust-provider \
  --trust-provider-type user \
  --user-trust-provider-type iam-identity-center \
  --policy-reference-name idc \
  --tag-specifications ResourceType=verified-access-trust-
  provider,Tags=[{Key=Name,Value=my-va-trust-provider}]
```

Salida:

```
{
  "VerifiedAccessTrustProvider": {
    "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
    "Description": "",
    "TrustProviderType": "user",
    "UserTrustProviderType": "iam-identity-center",
    "PolicyReferenceName": "idc",
    "CreationTime": "2023-08-25T18:40:36",
    "LastUpdatedTime": "2023-08-25T18:40:36",
    "Tags": [
      {
        "Key": "Name",
        "Value": "my-va-trust-provider"
      }
    ]
  }
}
```

Para obtener más información, consulte [Proveedores de confianza para el acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [CreateVerifiedAccessTrustProvider](#) la Referencia de AWS CLI comandos.

## create-volume

En el siguiente ejemplo de código, se muestra cómo usar `create-volume`.

### AWS CLI

Para crear un volumen de uso general SSD (gp2) vacío

El siguiente `create-volume` ejemplo crea un volumen de uso general SSD (gp2) de 80 GiB en la zona de disponibilidad especificada. Tenga en cuenta que la región actual debe ser `us-east-1`, o bien puede añadir el `--region` parámetro para especificar la región del comando.

```
aws ec2 create-volume \
  --volume-type gp2 \
  --size 80 \
  --availability-zone us-east-1a
```

Salida:

```
{
  "AvailabilityZone": "us-east-1a",
  "Tags": [],
  "Encrypted": false,
  "VolumeType": "gp2",
  "VolumeId": "vol-1234567890abcdef0",
  "State": "creating",
  "Iops": 240,
  "SnapshotId": "",
  "CreateTime": "YYYY-MM-DDTHH:MM:SS.000Z",
  "Size": 80
}
```

Si no especifica un tipo de volumen, el tipo de volumen predeterminado es gp2.

```
aws ec2 create-volume \
  --size 80 \
  --availability-zone us-east-1a
```

Ejemplo 2: Para crear un volumen aprovisionado IOPS SSD (io1) a partir de una instantánea

En el siguiente create-volume ejemplo, se crea un volumen aprovisionado IOPS SSD (io1) con 1000 aprovisionados IOPS en la zona de disponibilidad especificada mediante la instantánea especificada.

```
aws ec2 create-volume \
  --volume-type io1 \
  --iops 1000 \
  --snapshot-id snap-066877671789bd71b \
  --availability-zone us-east-1a
```

Salida:

```
{
  "AvailabilityZone": "us-east-1a",
  "Tags": [],
  "Encrypted": false,
  "VolumeType": "io1",
  "VolumeId": "vol-1234567890abcdef0",
  "State": "creating",
  "Iops": 1000,
```

```
"SnapshotId": "snap-066877671789bd71b",
"CreateTime": "YYYY-MM-DDTHH:MM:SS.000Z",
"Size": 500
}
```

### Ejemplo 3: Para crear un volumen cifrado

En el siguiente `create-volume` ejemplo, se crea un volumen cifrado con CMK el EBS cifrado predeterminado. Si el cifrado está deshabilitado de forma predeterminada, debe especificar el `--encrypted` parámetro de la siguiente manera.

```
aws ec2 create-volume \
  --size 80 \
  --encrypted \
  --availability-zone us-east-1a
```

Salida:

```
{
  "AvailabilityZone": "us-east-1a",
  "Tags": [],
  "Encrypted": true,
  "VolumeType": "gp2",
  "VolumeId": "vol-1234567890abcdef0",
  "State": "creating",
  "Iops": 240,
  "SnapshotId": "",
  "CreateTime": "YYYY-MM-DDTHH:MM:SS.000Z",
  "Size": 80
}
```

Si el cifrado está activado de forma predeterminada, el siguiente comando de ejemplo crea un volumen cifrado, incluso sin el `--encrypted` parámetro.

```
aws ec2 create-volume \
  --size 80 \
  --availability-zone us-east-1a
```

Si utiliza el `--kms-key-id` parámetro para especificar un cliente gestionadoCMK, debe especificarlo incluso si el `--encrypted` cifrado está activado de forma predeterminada.



```
aws ec2 create-volume \  
  --volume-type gp2 \  
  --size 80 \  
  --encrypted \  
  --kms-key-id 0ea3fef3-80a7-4778-9d8c-1c0c6EXAMPLE \  
  --availability-zone us-east-1a
```

Ejemplo 4: Para crear un volumen con etiquetas

En el siguiente `create-volume` ejemplo, se crea un volumen y se añaden dos etiquetas.

```
aws ec2 create-volume \  
  --availability-zone us-east-1a \  
  --volume-type gp2 \  
  --size 80 \  
  --tag-specifications 'ResourceType=volume, Tags=[{Key=purpose, Value=production},  
{Key=cost-center, Value=cc123}]'
```

- Para API obtener más información, consulte [CreateVolume](#) la Referencia de AWS CLI comandos.

## **create-vpc-endpoint-connection-notification**

En el siguiente ejemplo de código, se muestra cómo usar `create-vpc-endpoint-connection-notification`.

### AWS CLI

Para crear una notificación de conexión de punto final

En este ejemplo, se crea una notificación para un servicio de punto final específico que le avisa cuando los puntos finales de la interfaz se han conectado a su servicio y cuando los puntos finales se han aceptado para su servicio.

Comando:

```
aws ec2 create-vpc-endpoint-connection-notification --connection-notification-  
arn arn:aws:sns:us-east-2:123456789012:VpceNotification --connection-  
events Connect Accept --service-id vpce-svc-1237881c0d25a3abc
```

Salida:

```
{
  "ConnectionNotification": {
    "ConnectionNotificationState": "Enabled",
    "ConnectionNotificationType": "Topic",
    "ServiceId": "vpce-svc-1237881c0d25a3abc",
    "ConnectionEvents": [
      "Accept",
      "Connect"
    ],
    "ConnectionNotificationId": "vpce-nfn-008776de7e03f5abc",
    "ConnectionNotificationArn": "arn:aws:sns:us-
east-2:123456789012:VpceNotification"
  }
}
```

- Para API obtener más información, consulte la Referencia [CreateVpcEndpointConnectionNotification](#) de AWS CLI comandos.

## create-vpc-endpoint-service-configuration

En el siguiente ejemplo de código, se muestra cómo usar `create-vpc-endpoint-service-configuration`.

### AWS CLI

Ejemplo 1: Para crear una configuración de servicio de punto final para un punto final de interfaz

En el siguiente `create-vpc-endpoint-service-configuration` ejemplo, se crea una configuración de servicio de VPC punto final mediante Network Load Balancer `nlb-vpce`. En este ejemplo también se especifica que se deben aceptar las solicitudes de conexión al servicio a través de un punto final de la interfaz.

```
aws ec2 create-vpc-endpoint-service-configuration \
  --network-load-balancer-arns arn:aws:elasticloadbalancing:us-
east-1:123456789012:loadbalancer/net/nlb-vpce/e94221227f1ba532 \
  --acceptance-required
```

Salida:

```
{
```

```

"ServiceConfiguration": {
  "ServiceType": [
    {
      "ServiceType": "Interface"
    }
  ],
  "NetworkLoadBalancerArns": [
    "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/net/
nlb-vpce/e94221227f1ba532"
  ],
  "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-03d5ebb7d9579a2b3",
  "ServiceState": "Available",
  "ServiceId": "vpce-svc-03d5ebb7d9579a2b3",
  "AcceptanceRequired": true,
  "AvailabilityZones": [
    "us-east-1d"
  ],
  "BaseEndpointDnsNames": [
    "vpce-svc-03d5ebb7d9579a2b3.us-east-1.vpce.amazonaws.com"
  ]
}
}

```

Ejemplo 2: Para crear una configuración de servicio de punto final para un punto final de Gateway Load Balancer

En el siguiente `create-vpc-endpoint-service-configuration` ejemplo, se crea una configuración de servicio de VPC punto final mediante el `Gateway Load Balancer GWLBService`. Las solicitudes para conectarse al servicio a través de un punto final de Gateway Load Balancer se aceptan automáticamente.

```

aws ec2 create-vpc-endpoint-service-configuration \
  --gateway-load-balancer-arns arn:aws:elasticloadbalancing:us-
east-1:123456789012:loadbalancer/gwy/GWLBService/123123123123abcc \
  --no-acceptance-required

```

Salida:

```

{
  "ServiceConfiguration": {
    "ServiceType": [
      {

```

```

        "ServiceType": "GatewayLoadBalancer"
    }
],
"ServiceId": "vpce-svc-123123a1c43abc123",
"ServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-123123a1c43abc123",
"ServiceState": "Available",
"AvailabilityZones": [
    "us-east-1d"
],
"AcceptanceRequired": false,
"ManagesVpcEndpoints": false,
"GatewayLoadBalancerArns": [
    "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/gwy/
    GWLBService/123123123123abcc"
]
}
}

```

Para obtener más información, consulta [los servicios de VPC punto final](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [CreateVpcEndpointServiceConfiguration](#) la Referencia de AWS CLI comandos.

## create-vpc-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `create-vpc-endpoint`.

### AWS CLI

Ejemplo 1: Para crear un punto final de puerta de enlace

El siguiente `create-vpc-endpoint` ejemplo crea un VPC punto de enlace entre Amazon S3 VPC `vpc-1a2b3c4d` y Amazon S3 de la `us-east-1` región y asocia la tabla `rtb-11aa22bb` de rutas al punto de enlace.

```

aws ec2 create-vpc-endpoint \
  --vpc-id vpc-1a2b3c4d \
  --service-name com.amazonaws.us-east-1.s3 \
  --route-table-ids rtb-11aa22bb

```

Salida:

```
{
  "VpcEndpoint": {
    "PolicyDocument": "{\"Version\":\"2008-10-17\",\"Statement\":[{\"Sid\":\"\", \"Effect\":\"Allow\", \"Principal\":\"*\", \"Action\":\"*\", \"Resource\":\"*\"}]}",
    "VpcId": "vpc-1a2b3c4d",
    "State": "available",
    "ServiceName": "com.amazonaws.us-east-1.s3",
    "RouteTableIds": [
      "rtb-11aa22bb"
    ],
    "VpcEndpointId": "vpc-1a2b3c4d",
    "CreationTimestamp": "2015-05-15T09:40:50Z"
  }
}
```

Para obtener más información, consulte [Creación de un punto final de puerta](#) de enlace en la AWS PrivateLink guía.

Ejemplo 2: Para crear un punto final de interfaz

En el siguiente `create-vpc-endpoint` ejemplo, se crea un VPC punto final de interfaz entre Amazon S3 VPC `vpc-1a2b3c4d` y Amazon S3 de la `us-east-1` región. El comando crea el punto final en una subred `subnet-1a2b3c4d`, lo asocia al grupo `sg-1a2b3c4d` de seguridad y agrega una etiqueta con una clave de «Servicio» y un valor de «S3».

```
aws ec2 create-vpc-endpoint \
  --vpc-id vpc-1a2b3c4d \
  --vpc-endpoint-type Interface \
  --service-name com.amazonaws.us-east-1.s3 \
  --subnet-ids subnet-7b16de0c \
  --security-group-id sg-1a2b3c4d \
  --tag-specifications ResourceType=vpc-endpoint,Tags=[{Key=service,Value=S3}]
```

Salida:

```
{
  "VpcEndpoint": {
    "VpcEndpointId": "vpce-1a2b3c4d5e6f1a2b3",
    "VpcEndpointType": "Interface",
    "VpcId": "vpc-1a2b3c4d",
    "ServiceName": "com.amazonaws.us-east-1.s3",
```

```
"State": "pending",
"RouteTableIds": [],
"SubnetIds": [
  "subnet-1a2b3c4d"
],
"Groups": [
  {
    "GroupId": "sg-1a2b3c4d",
    "GroupName": "default"
  }
],
"PrivateDnsEnabled": false,
"RequesterManaged": false,
"NetworkInterfaceIds": [
  "eni-0b16f0581c8ac6877"
],
"DnsEntries": [
  {
    "DnsName": "*.vpce-1a2b3c4d5e6f1a2b3-9hnenorg.s3.us-
east-1.vpce.amazonaws.com",
    "HostedZoneId": "Z7HUB22UULQXV"
  },
  {
    "DnsName": "*.vpce-1a2b3c4d5e6f1a2b3-9hnenorg-us-east-1c.s3.us-
east-1.vpce.amazonaws.com",
    "HostedZoneId": "Z7HUB22UULQXV"
  }
],
"CreationTimestamp": "2021-03-05T14:46:16.030000+00:00",
"Tags": [
  {
    "Key": "service",
    "Value": "S3"
  }
],
"OwnerId": "123456789012"
}
}
```

Para obtener más información, consulte [Creación de un punto final de interfaz](#) en la Guía del usuario de AWS PrivateLink

Ejemplo 3: Para crear un punto final de Gateway Load Balancer

En el siguiente `create-vpc-endpoint` ejemplo, se crea un punto final de Gateway Load Balancer entre VPC `vpc-111122223333aabb` y un servicio que se configura mediante un Gateway Load Balancer.

```
aws ec2 create-vpc-endpoint \  
  --service-name com.amazonaws.vpce.us-east-1.vpce-svc-123123a1c43abc123 \  
  --vpc-endpoint-type GatewayLoadBalancer \  
  --vpc-id vpc-111122223333aabb \  
  --subnet-ids subnet-0011aabbcc2233445
```

Salida:

```
{  
  "VpcEndpoint": {  
    "VpcEndpointId": "vpce-aabbaabbaabbaabba",  
    "VpcEndpointType": "GatewayLoadBalancer",  
    "VpcId": "vpc-111122223333aabb",  
    "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-123123a1c43abc123",  
    "State": "pending",  
    "SubnetIds": [  
      "subnet-0011aabbcc2233445"  
    ],  
    "RequesterManaged": false,  
    "NetworkInterfaceIds": [  
      "eni-01010120203030405"  
    ],  
    "CreationTimestamp": "2020-11-11T08:06:03.522Z",  
    "OwnerId": "123456789012"  
  }  
}
```

Para obtener más información, consulte los [puntos finales del Gateway Load Balancer](#) en la Guía del usuario de AWS PrivateLink

- Para API obtener más información, consulte la Referencia [CreateVpcEndpoint](#) de AWS CLI comandos.

## **create-vpc-peering-connection**

En el siguiente ejemplo de código, se muestra cómo usar `create-vpc-peering-connection`.

## AWS CLI

Para crear una conexión VPC de interconexión entre sus VPCs

En este ejemplo, se solicita una conexión de emparejamiento entre el vpc-1a2b3c4d y el VPCs vpc-11122233.

Comando:

```
aws ec2 create-vpc-peering-connection --vpc-id vpc-1a2b3c4d --peer-vpc-id vpc-11122233
```

Salida:

```
{
  "VpcPeeringConnection": {
    "Status": {
      "Message": "Initiating Request to 444455556666",
      "Code": "initiating-request"
    },
    "Tags": [],
    "RequesterVpcInfo": {
      "OwnerId": "444455556666",
      "VpcId": "vpc-1a2b3c4d",
      "CidrBlock": "10.0.0.0/28"
    },
    "VpcPeeringConnectionId": "pcx-111aaa111",
    "ExpirationTime": "2014-04-02T16:13:36.000Z",
    "AcceptorVpcInfo": {
      "OwnerId": "444455556666",
      "VpcId": "vpc-11122233"
    }
  }
}
```

Para VPC crear una conexión VPC de emparejamiento con una de otra cuenta

En este ejemplo, se solicita una conexión de emparejamiento entre tu VPC (vpc-1a2b3c4d) y una (vpc-11122233) que pertenece a la cuenta 123456789012. VPC AWS

Comando:



```
aws ec2 create-vpc-peering-connection --vpc-id vpc-1a2b3c4d --peer-vpc-id vpc-11122233 --peer-owner-id 123456789012
```

Para crear VPC una VPC conexión de emparejamiento con una de una región diferente

En este ejemplo, se solicita una conexión de emparejamiento entre tu VPC cuenta de la región actual (*vpc-1a2b3c4d*) y una VPC (*vpc-11122233*) de tu cuenta de la región. *us-west-2*

Comando:

```
aws ec2 create-vpc-peering-connection --vpc-id vpc-1a2b3c4d --peer-vpc-id vpc-11122233 --peer-region us-west-2
```

En este ejemplo, se solicita una conexión de emparejamiento entre tu región actual (*vpc-1a2b3c4d*) y una (*vpc-11122233*) que pertenezca a la cuenta *123456789012* que se encuentre VPC en la región. VPC AWS *us-west-2*

Comando:

```
aws ec2 create-vpc-peering-connection --vpc-id vpc-1a2b3c4d --peer-vpc-id vpc-11122233 --peer-owner-id 123456789012 --peer-region us-west-2
```

- [API `CreateVpcPeeringConnection`](#) Para AWS CLI obtener más información, consulta la Referencia de comandos.

## create-vpc

En el siguiente ejemplo de código, se muestra cómo usar `create-vpc`.

### AWS CLI

Ejemplo 1: Para crear un VPC

En el siguiente `create-vpc` ejemplo, se crea un VPC con el IPv4 CIDR bloque especificado y una etiqueta de nombre.

```
aws ec2 create-vpc \  
  --cidr-block 10.0.0.0/16 \  
  --tag-specifications ResourceType=vpc, Tags=[{Key=Name, Value=MyVpc}]
```

Salida:

```
{
  "Vpc": {
    "CidrBlock": "10.0.0.0/16",
    "DhcpOptionsId": "dopt-5EXAMPLE",
    "State": "pending",
    "VpcId": "vpc-0a60eb65b4EXAMPLE",
    "OwnerId": "123456789012",
    "InstanceTenancy": "default",
    "Ipv6CidrBlockAssociationSet": [],
    "CidrBlockAssociationSet": [
      {
        "AssociationId": "vpc-cidr-assoc-07501b79ecEXAMPLE",
        "CidrBlock": "10.0.0.0/16",
        "CidrBlockState": {
          "State": "associated"
        }
      }
    ],
    "IsDefault": false,
    "Tags": [
      {
        "Key": "Name",
        "Value": "MyVpc"
      }
    ]
  }
}
```

Ejemplo 2: Para crear una VPC con un arrendamiento dedicado

En el siguiente `create-vpc` ejemplo, se crea un VPC con el IPv4 CIDR bloque especificado y el arrendamiento dedicado.

```
aws ec2 create-vpc \
  --cidr-block 10.0.0.0/16 \
  --instance-tenancy dedicated
```

Salida:

```
{
  "Vpc": {
```

```

    "CidrBlock": "10.0.0.0/16",
    "DhcpOptionsId": "dopt-19edf471",
    "State": "pending",
    "VpcId": "vpc-0a53287fa4EXAMPLE",
    "OwnerId": "111122223333",
    "InstanceTenancy": "dedicated",
    "Ipv6CidrBlockAssociationSet": [],
    "CidrBlockAssociationSet": [
      {
        "AssociationId": "vpc-cidr-assoc-00b24cc1c2EXAMPLE",
        "CidrBlock": "10.0.0.0/16",
        "CidrBlockState": {
          "State": "associated"
        }
      }
    ],
    "IsDefault": false
  }
}

```

Ejemplo 3: Para crear un VPC con un bloque IPv6 CIDR

En el siguiente `create-vpc` ejemplo, se crea un VPC con un bloque proporcionado por Amazon IPv6CIDR.

```

aws ec2 create-vpc \
  --cidr-block 10.0.0.0/16 \
  --amazon-provided-ipv6-cidr-block

```

Salida:

```

{
  "Vpc": {
    "CidrBlock": "10.0.0.0/16",
    "DhcpOptionsId": "dopt-dEXAMPLE",
    "State": "pending",
    "VpcId": "vpc-0fc5e3406bEXAMPLE",
    "OwnerId": "123456789012",
    "InstanceTenancy": "default",
    "Ipv6CidrBlockAssociationSet": [
      {
        "AssociationId": "vpc-cidr-assoc-068432c60bEXAMPLE",
        "Ipv6CidrBlock": "",

```

```

        "Ipv6CidrBlockState": {
            "State": "associating"
        },
        "Ipv6Pool": "Amazon",
        "NetworkBorderGroup": "us-west-2"
    }
],
"CidrBlockAssociationSet": [
    {
        "AssociationId": "vpc-cidr-assoc-0669f8f9f5EXAMPLE",
        "CidrBlock": "10.0.0.0/16",
        "CidrBlockState": {
            "State": "associated"
        }
    }
],
"IsDefault": false
}
}

```

Ejemplo 4: Para crear un VPC con a CIDR partir de un grupo IPAM

El siguiente `create-vpc` ejemplo crea un VPC con CIDR a partir de un grupo de Amazon VPC IP Address Manager (IPAM).

Linux y macOS:

```

aws ec2 create-vpc \
  --ipv4-ipam-pool-id ipam-pool-0533048da7d823723 \
  --tag-specifications ResourceType=vpc,Tags='[{"Key=Environment,Value="Preprod"}, {"Key=Owner,Value="Build Team"}]'
```

Windows:

```

aws ec2 create-vpc ^
  --ipv4-ipam-pool-id ipam-pool-0533048da7d823723 ^
  --tag-specifications ResourceType=vpc,Tags=[{"Key=Environment,Value="Preprod"}, {"Key=Owner,Value="Build Team"}]
```

Salida:

```
{
```

```
"Vpc": {
  "CidrBlock": "10.0.1.0/24",
  "DhcpOptionsId": "dopt-2afccf50",
  "State": "pending",
  "VpcId": "vpc-010e1791024eb0af9",
  "OwnerId": "123456789012",
  "InstanceTenancy": "default",
  "Ipv6CidrBlockAssociationSet": [],
  "CidrBlockAssociationSet": [
    {
      "AssociationId": "vpc-cidr-assoc-0a77de1d803226d4b",
      "CidrBlock": "10.0.1.0/24",
      "CidrBlockState": {
        "State": "associated"
      }
    }
  ],
  "IsDefault": false,
  "Tags": [
    {
      "Key": "Environment",
      "Value": "Preprod"
    },
    {
      "Key": "Owner",
      "Value": "Build Team"
    }
  ]
}
}
```

Para obtener más información, consulta [Crear un grupo VPC que utilice un IPAM grupo CIDR](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [CreateVpc](#) la Referencia de AWS CLI comandos.

## **create-vpn-connection-route**

En el siguiente ejemplo de código, se muestra cómo usar `create-vpn-connection-route`.

### AWS CLI

Para crear una ruta estática para una VPN conexión

En este ejemplo, se crea una ruta estática para la VPN conexión especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 create-vpn-connection-route --vpn-connection-id vpn-40f41529 --destination-cidr-block 11.12.0.0/16
```

- Para API obtener más información, consulte [CreateVpnConnectionRoute](#) la Referencia de AWS CLI comandos.

## create-vpn-connection

En el siguiente ejemplo de código, se muestra cómo usar `create-vpn-connection`.

AWS CLI

Ejemplo 1: Para crear una VPN conexión con enrutamiento dinámico

El siguiente `create-vpn-connection` ejemplo crea una VPN conexión entre la puerta de enlace privada virtual especificada y la puerta de enlace del cliente especificada y aplica etiquetas a la VPN conexión. El resultado incluye la información de configuración del dispositivo de pasarela de cliente, en XML formato.

```
aws ec2 create-vpn-connection \  
  --type ipsec.1 \  
  --customer-gateway-id cgw-001122334455aabb \  
  --vpn-gateway-id vgw-1a1a1a1a1a1a2b2b2 \  
  --tag-specification 'ResourceType=vpn-connection,Tags=[{Key=Name,Value=BGP-VPN}]'
```

Salida:

```
{  
  "VpnConnection": {  
    "CustomerGatewayConfiguration": "...configuration information...",  
    "CustomerGatewayId": "cgw-001122334455aabb",  
    "Category": "VPN",  
    "State": "pending",  
    "VpnConnectionId": "vpn-123123123123abcab",  
    "VpnGatewayId": "vgw-1a1a1a1a1a1a2b2b2",  
    "Options": {
```

```

    "EnableAcceleration": false,
    "StaticRoutesOnly": false,
    "LocalIpv4NetworkCidr": "0.0.0.0/0",
    "RemoteIpv4NetworkCidr": "0.0.0.0/0",
    "TunnelInsideIpVersion": "ipv4",
    "TunnelOptions": [
      {},
      {}
    ]
  },
  "Routes": [],
  "Tags": [
    {
      "Key": "Name",
      "Value": "BGP-VPN"
    }
  ]
}
}

```

Para obtener más información, consulte [Cómo VPN funciona un AWS sitio a otro](#) en la Guía del usuario de AWS sitio a sitio. VPN

Ejemplo 2: Para crear una conexión con enrutamiento estático VPN

El siguiente `create-vpn-connection` ejemplo crea una VPN conexión entre la puerta de enlace privada virtual especificada y la puerta de enlace del cliente especificada. Las opciones especifican el enrutamiento estático. El resultado incluye la información de configuración del dispositivo de pasarela de cliente, en XML formato.

```

aws ec2 create-vpn-connection \
  --type ipsec.1 \
  --customer-gateway-id cgw-001122334455aabbc \
  --vpn-gateway-id vgw-1a1a1a1a1a1a2b2b2 \
  --options "{\"StaticRoutesOnly\":true}"

```

Salida:

```

{
  "VpnConnection": {
    "CustomerGatewayConfiguration": "..configuration information...",
    "CustomerGatewayId": "cgw-001122334455aabbc",

```

```

    "Category": "VPN",
    "State": "pending",
    "VpnConnectionId": "vpn-123123123123abcb",
    "VpnGatewayId": "vgw-1a1a1a1a1a1a2b2b2",
    "Options": {
      "EnableAcceleration": false,
      "StaticRoutesOnly": true,
      "LocalIpv4NetworkCidr": "0.0.0.0/0",
      "RemoteIpv4NetworkCidr": "0.0.0.0/0",
      "TunnelInsideIpVersion": "ipv4",
      "TunnelOptions": [
        {},
        {}
      ]
    },
    "Routes": [],
    "Tags": []
  }
}

```

Para obtener más información, consulte [Cómo VPN funciona un AWS sitio a otro](#) en la Guía del usuario de AWS sitio a sitio. VPN

Ejemplo 3: Para crear una VPN conexión y especificar su propia clave interna y previamente compartida CIDR

El siguiente `create-vpn-connection` ejemplo crea una VPN conexión y especifica el CIDR bloque de direcciones IP interno y una clave previamente compartida personalizada para cada túnel. Los valores especificados se devuelven en la `CustomerGatewayConfiguration` información.

```

aws ec2 create-vpn-connection \
  --type ipsec.1 \
  --customer-gateway-id cgw-001122334455aabb \
  --vpn-gateway-id vgw-1a1a1a1a1a1a2b2b2 \
  --options
  TunnelOptions='[{TunnelInsideCidr=169.254.12.0/30,PreSharedKey=ExamplePreSharedKey1},
  {TunnelInsideCidr=169.254.13.0/30,PreSharedKey=ExamplePreSharedKey2}]'

```

Salida:

```
{
```



```

"VpnConnection": {
  "CustomerGatewayConfiguration": "..configuration information...",
  "CustomerGatewayId": "cgw-001122334455aabbcc",
  "Category": "VPN",
  "State": "pending",
  "VpnConnectionId": "vpn-123123123123abcab",
  "VpnGatewayId": "vgw-1a1a1a1a1a1a2b2b2",
  "Options": {
    "EnableAcceleration": false,
    "StaticRoutesOnly": false,
    "LocalIpv4NetworkCidr": "0.0.0.0/0",
    "RemoteIpv4NetworkCidr": "0.0.0.0/0",
    "TunnelInsideIpVersion": "ipv4",
    "TunnelOptions": [
      {
        "OutsideIpAddress": "203.0.113.3",
        "TunnelInsideCidr": "169.254.12.0/30",
        "PreSharedKey": "ExamplePreSharedKey1"
      },
      {
        "OutsideIpAddress": "203.0.113.5",
        "TunnelInsideCidr": "169.254.13.0/30",
        "PreSharedKey": "ExamplePreSharedKey2"
      }
    ]
  },
  "Routes": [],
  "Tags": []
}
}

```

Para obtener más información, consulte [Cómo VPN funciona un AWS sitio a otro](#) en la Guía del usuario de AWS sitio a sitio. VPN

Ejemplo 4: Para crear una VPN conexión que admita el tráfico IPv6

El siguiente `create-vpn-connection` ejemplo crea una VPN conexión que admite el IPv6 tráfico entre la puerta de enlace de tránsito especificada y la puerta de enlace del cliente especificada. Las opciones de túnel para ambos túneles especifican AWS quién debe iniciar la IKE negociación.

```

aws ec2 create-vpn-connection \
  --type ipsec.1 \

```

```
--transit-gateway-id tgw-12312312312312312 \  
--customer-gateway-id cgw-001122334455aabbc \  
--options TunnelInsideIpVersion=ipv6,TunnelOptions=[{StartupAction=start},  
{StartupAction=start}]
```

Salida:

```
{  
  "VpnConnection": {  
    "CustomerGatewayConfiguration": "..configuration information..",  
    "CustomerGatewayId": "cgw-001122334455aabbc",  
    "Category": "VPN",  
    "State": "pending",  
    "VpnConnectionId": "vpn-111111111122222222",  
    "TransitGatewayId": "tgw-12312312312312312",  
    "Options": {  
      "EnableAcceleration": false,  
      "StaticRoutesOnly": false,  
      "LocalIpv6NetworkCidr": "::/0",  
      "RemoteIpv6NetworkCidr": "::/0",  
      "TunnelInsideIpVersion": "ipv6",  
      "TunnelOptions": [  
        {  
          "OutsideIpAddress": "203.0.113.3",  
          "StartupAction": "start"  
        },  
        {  
          "OutsideIpAddress": "203.0.113.5",  
          "StartupAction": "start"  
        }  
      ]  
    },  
    "Routes": [],  
    "Tags": []  
  }  
}
```

Para obtener más información, consulte [Cómo VPN funciona un AWS sitio a otro](#) en la Guía del usuario de AWS sitio a sitio. VPN

- Para obtener API más información, consulte la Referencia de comandos [CreateVpnConnection](#).AWS CLI

## create-vpn-gateway

En el siguiente ejemplo de código, se muestra cómo usar `create-vpn-gateway`.

### AWS CLI

Para crear una puerta de enlace privada virtual

En este ejemplo se crea una puerta de enlace privada virtual.

Comando:

```
aws ec2 create-vpn-gateway --type ipsec.1
```

Salida:

```
{
  "VpnGateway": {
    "AmazonSideAsn": 64512,
    "State": "available",
    "Type": "ipsec.1",
    "VpnGatewayId": "vgw-9a4cacf3",
    "VpcAttachments": []
  }
}
```

Para crear una puerta de enlace privada virtual con un lado específico de Amazon ASN

En este ejemplo, se crea una puerta de enlace privada virtual y se especifica el número de sistema autónomo (ASN) para la parte de Amazon de la BGP sesión.

Comando:

```
aws ec2 create-vpn-gateway --type ipsec.1 --amazon-side-asn 65001
```

Salida:

```
{
  "VpnGateway": {
    "AmazonSideAsn": 65001,
    "State": "available",
    "Type": "ipsec.1",
    "VpnGatewayId": "vgw-9a4cacf3",
  }
}
```

```
    "VpcAttachments": []
  }
}
```

- Para API obtener más información, consulte [CreateVpnGateway](#) la Referencia de AWS CLI comandos.

## delete-carrier-gateway

En el siguiente ejemplo de código, se muestra cómo usar `delete-carrier-gateway`.

### AWS CLI

Para eliminar la puerta de enlace de tu operador

En el siguiente `delete-carrier-gateway` ejemplo, se elimina la puerta de enlace del operador especificada.

```
aws ec2 delete-carrier-gateway \
  --carrier-gateway-id cagw-0465cdEXAMPLE1111
```

Salida:

```
{
  "CarrierGateway": {
    "CarrierGatewayId": "cagw-0465cdEXAMPLE1111",
    "VpcId": "vpc-0c529aEXAMPLE1111",
    "State": "deleting",
    "OwnerId": "123456789012"
  }
}
```

Para obtener más información, consulte [Carrier Gateways](#) en la Guía del usuario de Amazon Virtual Private Cloud.

- Para API obtener más información, consulte [DeleteCarrierGateway](#) la Referencia de AWS CLI comandos.

## delete-client-vpn-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `delete-client-vpn-endpoint`.

## AWS CLI

Para eliminar un VPN punto final del cliente

En el siguiente `delete-client-vpn-endpoint` ejemplo, se elimina el VPN punto final del cliente especificado.

```
aws ec2 delete-client-vpn-endpoint \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Salida:

```
{  
  "Status": {  
    "Code": "deleting"  
  }  
}
```

Para obtener más información, consulte los [VPNpuntos finales del cliente](#) en la Guía del VPNadministrador de AWS clientes.

- Para API obtener más información, consulte [DeleteClientVpnEndpoint](#) la Referencia de AWS CLI comandos.

## `delete-client-vpn-route`

En el siguiente ejemplo de código, se muestra cómo usar `delete-client-vpn-route`.

## AWS CLI

Para eliminar una ruta para un VPN punto final del cliente

En el siguiente `delete-client-vpn-route` ejemplo, se elimina la `0.0.0.0/0` ruta de la subred especificada de un punto final del clienteVPN.

```
aws ec2 delete-client-vpn-route \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \  
  --destination-cidr-block 0.0.0.0/0 \  
  --target-vpc-subnet-id subnet-0123456789abcabca
```

Salida:

```
{
  "Status": {
    "Code": "deleting"
  }
}
```

Para obtener más información, consulte [Rutas](#) en la Guía del VPNadministrador del AWS cliente.

- Para API obtener más información, consulte [DeleteClientVpnRoute](#) la Referencia de AWS CLI comandos.

## delete-coip-cidr

En el siguiente ejemplo de código, se muestra cómo usar `delete-coip-cidr`.

### AWS CLI

Para eliminar un rango de direcciones IP (CoIP) propiedad del cliente

En el siguiente `delete-coip-cidr` ejemplo, se elimina el rango especificado de direcciones de CoIP en el grupo de CoIP especificado.

```
aws ec2 delete-coip-cidr \
  --cidr 14.0.0.0/24 \
  --coip-pool-id ipv4pool-coip-1234567890abcdefg
```

Salida:

```
{
  "CoipCidr": {
    "Cidr": "14.0.0.0/24",
    "CoipPoolId": "ipv4pool-coip-1234567890abcdefg",
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890"
  }
}
```

Para obtener más información, consulte [Direcciones IP propiedad del cliente](#) en la Guía del usuario de AWS .

- Para API obtener más información, consulte la Referencia de [DeleteCoipCidr](#) comandos AWS CLI .

## delete-coip-pool

En el siguiente ejemplo de código, se muestra cómo usar `delete-coip-pool`.

### AWS CLI

Para eliminar un conjunto de direcciones IP (CoIP) propiedad del cliente

En el siguiente `delete-coip-pool` ejemplo, se elimina un conjunto de direcciones CoIP de CoIP.

```
aws ec2 delete-coip-pool \  
  --coip-pool-id ipv4pool-coip-1234567890abcdefg
```

Salida:

```
{  
  "CoipPool": {  
    "PoolId": "ipv4pool-coip-1234567890abcdefg",  
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",  
    "PoolArn": "arn:aws:ec2:us-west-2:123456789012:coip-pool/ipv4pool-  
coip-1234567890abcdefg"  
  }  
}
```

Para obtener más información, consulte [Direcciones IP propiedad del cliente](#) en la Guía del usuario de AWS .

- Para API obtener más información, consulte la Referencia de [DeleteCoipPool](#) comandos AWS CLI .

## delete-customer-gateway

En el siguiente ejemplo de código, se muestra cómo usar `delete-customer-gateway`.

### AWS CLI

Para eliminar una pasarela de clientes

En este ejemplo, se elimina la pasarela de clientes especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-customer-gateway --customer-gateway-id cgw-0e11f167
```

- Para API obtener más información, consulte [DeleteCustomerGateway](#) la Referencia de AWS CLI comandos.

## **delete-dhcp-options**

En el siguiente ejemplo de código, se muestra cómo usar `delete-dhcp-options`.

AWS CLI

Para eliminar un conjunto DHCP de opciones

En este ejemplo se elimina el conjunto de DHCP opciones especificado. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-dhcp-options --dhcp-options-id dopt-d9070ebb
```

- Para API obtener más información, consulte [DeleteDhcpOptions](#) la Referencia de AWS CLI comandos.

## **delete-egress-only-internet-gateway**

En el siguiente ejemplo de código, se muestra cómo usar `delete-egress-only-internet-gateway`.

AWS CLI

Para eliminar una puerta de enlace a Internet de solo salida

En este ejemplo, se elimina la puerta de enlace de Internet de solo salida especificada.

Comando:

```
aws ec2 delete-egress-only-internet-gateway --egress-only-internet-gateway-id eigw-01eadbd45ecd7943f
```



Salida:

```
{
  "ReturnCode": true
}
```

- Para obtener API más información, consulte la Referencia de comandos [DeleteEgressOnlyInternetGateway](#).AWS CLI

## delete-fleets

En el siguiente ejemplo de código, se muestra cómo usar delete-fleets.

AWS CLI

Ejemplo 1: Para eliminar una EC2 flota y cerrar las instancias asociadas

En el siguiente delete-fleets ejemplo, se elimina la EC2 flota especificada y se cancelan las instancias puntuales y bajo demanda asociadas.

```
aws ec2 delete-fleets \
  --fleet-ids fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE \
  --terminate-instances
```

Salida:

```
{
  "SuccessfulFleetDeletions": [
    {
      "CurrentFleetState": "deleted_terminating",
      "PreviousFleetState": "active",
      "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
    }
  ],
  "UnsuccessfulFleetDeletions": []
}
```

Para obtener más información, consulte [Eliminar una EC2 flota](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

Ejemplo 2: Eliminar una EC2 flota sin cerrar las instancias asociadas

En el siguiente `delete-fleets` ejemplo, se elimina la EC2 flota especificada sin cerrar las instancias puntuales y bajo demanda asociadas.

```
aws ec2 delete-fleets \  
  --fleet-ids fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE \  
  --no-terminate-instances
```

Salida:

```
{  
  "SuccessfulFleetDeletions": [  
    {  
      "CurrentFleetState": "deleted_running",  
      "PreviousFleetState": "active",  
      "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"  
    }  
  ],  
  "UnsuccessfulFleetDeletions": []  
}
```

Para obtener más información, consulte [Eliminar una EC2 flota](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [DeleteFleets](#) la Referencia de AWS CLI comandos.

## delete-flow-logs

En el siguiente ejemplo de código, se muestra cómo usar `delete-flow-logs`.

AWS CLI

Para eliminar un registro de flujo

En el siguiente `delete-flow-logs` ejemplo, se elimina el registro de flujo especificado.

```
aws ec2 delete-flow-logs --flow-log-id fl-11223344556677889
```

Salida:

```
{  
  "Unsuccessful": []  
}
```

```
}
```

- Para API obtener más información, consulte [DeleteFlowLogs](#) la Referencia de AWS CLI comandos.

## delete-fpga-image

En el siguiente ejemplo de código, se muestra cómo usar `delete-fpga-image`.

### AWS CLI

Para eliminar una FPGA imagen de Amazon

En este ejemplo se elimina lo especificado AFI.

Comando:

```
aws ec2 delete-fpga-image --fpga-image-id afi-06b12350a123fbabc
```

Salida:

```
{  
  "Return": true  
}
```

- Para API obtener más información, consulte [DeleteFpgaImage](#) la Referencia de AWS CLI comandos.

## delete-instance-connect-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `delete-instance-connect-endpoint`.

### AWS CLI

Para eliminar un punto final de EC2 Instance Connect

En el siguiente `delete-instance-connect-endpoint` ejemplo, se elimina el punto final EC2 Instance Connect especificado.

```
aws ec2 delete-instance-connect-endpoint \  

```

```
--instance-connect-endpoint-id eice-03f5e49b83924bbc7
```

Salida:

```
{
  "InstanceConnectEndpoint": {
    "OwnerId": "111111111111",
    "InstanceConnectEndpointId": "eice-0123456789example",
    "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-
connect-endpoint/eice-0123456789example",
    "State": "delete-in-progress",
    "StateMessage": "",
    "NetworkInterfaceIds": [],
    "VpcId": "vpc-0123abcd",
    "AvailabilityZone": "us-east-1d",
    "CreatedAt": "2023-02-07T12:05:37+00:00",
    "SubnetId": "subnet-0123abcd"
  }
}
```

Para obtener más información, consulte [Eliminar EC2 Instance Connect Endpoint](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DeleteInstanceConnectEndpoint](#) la Referencia de AWS CLI comandos.

## **delete-instance-event-window**

En el siguiente ejemplo de código, se muestra cómo usar `delete-instance-event-window`.

AWS CLI

Ejemplo 1: Para eliminar una ventana de eventos

En el siguiente `delete-instance-event-window` ejemplo, se elimina una ventana de eventos.

```
aws ec2 delete-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890
```

Salida:

```
{
  "InstanceEventWindowState": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "State": "deleting"
  }
}
```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para forzar la eliminación de una ventana de eventos

En el siguiente `delete-instance-event-window` ejemplo, se fuerza la eliminación de una ventana de eventos si la ventana de eventos está asociada actualmente a los objetivos.

```
aws ec2 delete-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --force-delete
```

Salida:

```
{
  "InstanceEventWindowState": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "State": "deleting"
  }
}
```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DeleteInstanceEventWindow](#) la Referencia de AWS CLI comandos.

## **delete-internet-gateway**

En el siguiente ejemplo de código, se muestra cómo usar `delete-internet-gateway`.

### AWS CLI

Para eliminar una puerta de enlace a Internet

En el siguiente `delete-internet-gateway` ejemplo, se elimina la puerta de enlace de Internet especificada.

```
aws ec2 delete-internet-gateway \  
  --internet-gateway-id igw-0d0fb496b3EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [las pasarelas de Internet](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DeleteInternetGateway](#) la Referencia de AWS CLI comandos.

## `delete-ipam-pool`

En el siguiente ejemplo de código, se muestra cómo usar `delete-ipam-pool`.

### AWS CLI

Para eliminar un IPAM grupo

En este ejemplo, usted es un administrador IPAM delegado que desea eliminar un IPAM grupo que ya no necesita, pero el grupo tiene uno CIDR aprovisionado. No puedes eliminar un grupo si se ha CIDRs aprovisionado para él, a menos que utilices la `--cascade` opción, por lo que utilizarás. `--cascade`

Para completar esta solicitud:

Necesitarás el identificador del IPAM grupo con el que puedes obtener [describe-ipam-pools](#). `--region` Debe ser la región de IPAM origen.

En el siguiente `delete-ipam-pool` ejemplo, se elimina un IPAM grupo de tu AWS cuenta.

```
aws ec2 delete-ipam-pool \  
  --ipam-pool-id ipam-pool-050c886a3ca41cd5b \  
  --cascade \  
  --region us-east-1
```

Salida:

```
{
```

```
"IpamPool": {
  "OwnerId": "320805250157",
  "IpamPoolId": "ipam-pool-050c886a3ca41cd5b",
  "IpamPoolArn": "arn:aws:ec2::320805250157:ipam-pool/ipam-
pool-050c886a3ca41cd5b",
  "IpamScopeArn": "arn:aws:ec2::320805250157:ipam-scope/ipam-
scope-0a158dde35c51107b",
  "IpamScopeType": "private",
  "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",
  "IpamRegion": "us-east-1",
  "Locale": "None",
  "PoolDepth": 1,
  "State": "delete-in-progress",
  "Description": "example",
  "AutoImport": false,
  "AddressFamily": "ipv4",
  "AllocationMinNetmaskLength": 0,
  "AllocationMaxNetmaskLength": 32
}
}
```

Para obtener más información, consulta [Eliminar un grupo](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [DeleteIpamPool](#) la Referencia de AWS CLI comandos.

## **delete-ipam-resource-discovery**

En el siguiente ejemplo de código, se muestra cómo usar `delete-ipam-resource-discovery`.

### AWS CLI

Para eliminar un descubrimiento de recursos

En este ejemplo, usted es un administrador IPAM delegado que desea eliminar una detección de recursos no predeterminada que creó para compartirla con otro IPAM administrador durante el proceso de integración IPAM con cuentas ajenas a su organización.

Para completar esta solicitud:

`--region` Debe ser la región en la que creó la detección de recursos. No puede eliminar una detección de recursos predeterminada si. `"IsDefault": true` Una detección de recursos

predeterminada es aquella que se crea automáticamente en la cuenta que crea una. IPAM Para eliminar una detección de recursos predeterminada, debe eliminar la IPAM.

En el siguiente `delete-ipam-resource-discovery` ejemplo, se elimina una detección de recursos.

```
aws ec2 delete-ipam-resource-discovery \
  --ipam-resource-discovery-id ipam-res-disco-0e39761475298ee0f \
  --region us-east-1
```

Salida:

```
{
  "IpamResourceDiscovery": {
    "OwnerId": "149977607591",
    "IpamResourceDiscoveryId": "ipam-res-disco-0e39761475298ee0f",
    "IpamResourceDiscoveryArn": "arn:aws:ec2::149977607591:ipam-resource-
discovery/ipam-res-disco-0e39761475298ee0f",
    "IpamResourceDiscoveryRegion": "us-east-1",
    "OperatingRegions": [
      {
        "RegionName": "us-east-1"
      }
    ],
    "IsDefault": false,
    "State": "delete-in-progress"
  }
}
```

Para obtener más información sobre los descubrimientos de recursos, consulte [Trabajar con descubrimientos de recursos](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [DeleteIpamResourceDiscovery](#) la Referencia de AWS CLI comandos.

## **delete-ipam-scope**

En el siguiente ejemplo de código, se muestra cómo usar `delete-ipam-scope`.

AWS CLI

Para eliminar un IPAM ámbito



En el siguiente `delete-ipam-scope` ejemplo, se elimina un IPAM.

```
aws ec2 delete-ipam-scope \  
  --ipam-scope-id ipam-scope-01c1ebab2b63bd7e4
```

Salida:

```
{  
  "IpamScope": {  
    "OwnerId": "123456789012",  
    "IpamScopeId": "ipam-scope-01c1ebab2b63bd7e4",  
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-  
scope-01c1ebab2b63bd7e4",  
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",  
    "IpamRegion": "us-east-1",  
    "IpamScopeType": "private",  
    "IsDefault": false,  
    "Description": "Example description",  
    "PoolCount": 0,  
    "State": "delete-in-progress"  
  }  
}
```

Para obtener más información, consulta [Eliminar un ámbito](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [DeleteIpamScope](#) la Referencia de AWS CLI comandos.

## **delete-ipam**

En el siguiente ejemplo de código, se muestra cómo usar `delete-ipam`.

### AWS CLI

Para eliminar un IPAM

En el `delete-ipam` ejemplo siguiente se elimina un IPAM.

```
aws ec2 delete-ipam \  
  --ipam-id ipam-036486dfa6af58ee0
```

Salida:

```
{
  "Ipam": {
    "OwnerId": "123456789012",
    "IpamId": "ipam-036486dfa6af58ee0",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-036486dfa6af58ee0",
    "IpamRegion": "us-east-1",
    "PublicDefaultScopeId": "ipam-scope-071b8042b0195c183",
    "PrivateDefaultScopeId": "ipam-scope-0807405dece705a30",
    "ScopeCount": 2,
    "OperatingRegions": [
      {
        "RegionName": "us-east-1"
      },
      {
        "RegionName": "us-east-2"
      },
      {
        "RegionName": "us-west-1"
      }
    ],
    "State": "delete-in-progress"
  }
}
```

Para obtener más información, consulta [Eliminar un IPAM](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [DeleteIpam](#) la Referencia de AWS CLI comandos.

## delete-key-pair

En el siguiente ejemplo de código, se muestra cómo usar `delete-key-pair`.

AWS CLI

Eliminar un par de claves

En el siguiente `delete-key-pair` ejemplo, se elimina el key pair especificado.

```
aws ec2 delete-key-pair \
```

```
--key-name my-key-pair
```

Salida:

```
{
  "Return": true,
  "KeyPairId": "key-03c8d3aceb53b507"
}
```

Para obtener más información, consulte [Crear y eliminar pares de claves](#) en la Guía del usuario de la interfaz de línea de AWS comandos.

- Para API obtener más información, consulte [DeleteKeyPair](#) la Referencia de AWS CLI comandos.

## delete-launch-template-versions

En el siguiente ejemplo de código, se muestra cómo usar `delete-launch-template-versions`.

AWS CLI

Para eliminar una versión de una plantilla de lanzamiento

En este ejemplo, se elimina la versión de la plantilla de lanzamiento especificada.

Comando:

```
aws ec2 delete-launch-template-versions --launch-template-id lt-0abcd290751193123 --versions 1
```

Salida:

```
{
  "UnsuccessfullyDeletedLaunchTemplateVersions": [],
  "SuccessfullyDeletedLaunchTemplateVersions": [
    {
      "LaunchTemplateName": "TestVersion",
      "VersionNumber": 1,
      "LaunchTemplateId": "lt-0abcd290751193123"
    }
  ]
}
```

```
]
}
```

- Para API obtener más información, consulte [DeleteLaunchTemplateVersions](#) la Referencia de AWS CLI comandos.

## delete-launch-template

En el siguiente ejemplo de código, se muestra cómo usar `delete-launch-template`.

### AWS CLI

Eliminar una plantilla de lanzamiento

En este ejemplo, se elimina la plantilla de lanzamiento especificada.

Comando:

```
aws ec2 delete-launch-template --launch-template-id lt-0abcd290751193123
```

Salida:

```
{
  "LaunchTemplate": {
    "LatestVersionNumber": 2,
    "LaunchTemplateId": "lt-0abcd290751193123",
    "LaunchTemplateName": "TestTemplate",
    "DefaultVersionNumber": 2,
    "CreatedBy": "arn:aws:iam::123456789012:root",
    "CreateTime": "2017-11-23T16:46:25.000Z"
  }
}
```

- Para API obtener más información, consulte [DeleteLaunchTemplate](#) la Referencia de AWS CLI comandos.

## delete-local-gateway-route-table-virtual-interface-group-association

En el siguiente ejemplo de código, se muestra cómo usar `delete-local-gateway-route-table-virtual-interface-group-association`.

## AWS CLI

Para desasociar una tabla de enrutamiento de una puerta de enlace local de un grupo de interfaces virtuales (VIFs)

El siguiente `delete-local-gateway-route-table-virtual-interface-group-association` ejemplo elimina la asociación entre la tabla de rutas de la puerta de enlace local especificada y VIF el grupo.

```
aws ec2 delete-local-gateway-route-table-virtual-interface-group-association \
  --local-gateway-route-table-virtual-interface-group-association-id lgw-vif-grp-
  assoc-exampleid12345678
```

Salida:

```
{
  "LocalGatewayRouteTableVirtualInterfaceGroupAssociation": {
    "LocalGatewayRouteTableVirtualInterfaceGroupAssociationId": "lgw-vif-grp-
    assoc-exampleid12345678",
    "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-exampleid0123abcd",
    "LocalGatewayId": "lgw-exampleid11223344",
    "LocalGatewayRouteTableId": "lgw-rtb-exampleidabcd1234",
    "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:111122223333:local-
    gateway-route-table/lgw-rtb-exampleidabcd1234",
    "OwnerId": "111122223333",
    "State": "disassociating",
    "Tags": []
  }
}
```

Para obtener más información, consulta [las asociaciones de VIF grupos](#) en la Guía del usuario de AWS Outposts.

- Para API obtener más información, consulte [DeleteLocalGatewayRouteTableVirtualInterfaceGroupAssociation](#) la Referencia de AWS CLI comandos.

## **delete-local-gateway-route-table-vpc-association**

En el siguiente ejemplo de código, se muestra cómo usar `delete-local-gateway-route-table-vpc-association`.

## AWS CLI

Para desasociar una tabla de enrutamiento de una puerta de enlace local de una VPC

El siguiente `delete-local-gateway-route-table-vpc-association` ejemplo elimina la asociación entre la tabla de enrutamiento de la puerta de enlace local especificada y VPC

```
aws ec2 delete-local-gateway-route-table-vpc-association \  
--local-gateway-route-table-vpc-association-id vpc-example0123456789
```

Salida:

```
{  
  "LocalGatewayRouteTableVpcAssociation": {  
    "LocalGatewayRouteTableVpcAssociationId": "lgw-vpc-assoc-abcd1234wxyz56789",  
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",  
    "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:555555555555:local-  
gateway-route-table/lgw-rtb-abcdefg1234567890",  
    "LocalGatewayId": "lgw-exampleid01234567",  
    "VpcId": "vpc-example0123456789",  
    "OwnerId": "555555555555",  
    "State": "disassociating"  
  }  
}
```

Para obtener más información, consulta [VPCs y asociaciones](#) en la Guía del usuario de AWS Outposts.

- Para API obtener más información, consulte [DeleteLocalGatewayRouteTableVpcAssociation](#) la Referencia de AWS CLI comandos.

## **delete-local-gateway-route-table**

En el siguiente ejemplo de código, se muestra cómo usar `delete-local-gateway-route-table`.

## AWS CLI

Para eliminar una tabla de enrutamiento de una puerta de enlace local

El siguiente `delete-local-gateway-route-table` ejemplo crea una tabla de enrutamiento de puerta de enlace local con el modo de VPC enrutamiento directo.

```
aws ec2 delete-local-gateway-route-table \  
--local-gateway-route-table-id lgw-rtb-abcdefg1234567890
```

Salida:

```
{  
  "LocalGatewayRouteTable": {  
    "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",  
    "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:111122223333:local-  
gateway-route-table/lgw-rtb-abcdefg1234567890",  
    "LocalGatewayId": "lgw-1a2b3c4d5e6f7g8h9",  
    "OutpostArn": "arn:aws:outposts:us-west-2:111122223333:outpost/  
op-021345abcdef67890",  
    "OwnerId": "111122223333",  
    "State": "deleting",  
    "Tags": [],  
    "Mode": "direct-vpc-routing"  
  }  
}
```

Para obtener más información, consulte [Tablas de enrutamiento de puerta de enlace locales](#) en la Guía del usuario de AWS Outposts.

- Para API obtener más información, consulte [DeleteLocalGatewayRouteTable](#) la Referencia de AWS CLI comandos.

## delete-local-gateway-route

En el siguiente ejemplo de código, se muestra cómo usar `delete-local-gateway-route`.

### AWS CLI

Para eliminar una ruta de una tabla de rutas de una puerta de enlace local

El siguiente `delete-local-gateway-route` ejemplo elimina la ruta especificada de la tabla de rutas de la puerta de enlace local especificada.

```
aws ec2 delete-local-gateway-route \  
--destination-cidr-block 0.0.0.0/0 \  
--local-gateway-route-table-id lgw-rtb-059615ef7dEXAMPLE
```

Salida:

```
{
  "Route": {
    "DestinationCidrBlock": "0.0.0.0/0",
    "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
    "Type": "static",
    "State": "deleted",
    "LocalGatewayRouteTableId": "lgw-rtb-059615ef7EXAMPLE"
  }
}
```

- Para API obtener más información, consulte [DeleteLocalGatewayRoute](#) la Referencia de AWS CLI comandos.

## delete-managed-prefix-list

En el siguiente ejemplo de código, se muestra cómo usar `delete-managed-prefix-list`.

AWS CLI

Para eliminar una lista de prefijos

En el siguiente `delete-managed-prefix-list` ejemplo, se elimina la lista de prefijos especificada.

```
aws ec2 delete-managed-prefix-list \
  --prefix-list-id pl-0123456abcabcabc1
```

Salida:

```
{
  "PrefixList": {
    "PrefixListId": "pl-0123456abcabcabc1",
    "AddressFamily": "IPv4",
    "State": "delete-in-progress",
    "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-0123456abcabcabc1",
    "PrefixListName": "test",
    "MaxEntries": 10,
    "Version": 1,
  }
}
```



```
    "OwnerId": "123456789012"  
  }  
}
```

Para obtener más información, consulta [Listas de prefijos gestionadas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DeleteManagedPrefixList](#) la Referencia de AWS CLI comandos.

## delete-nat-gateway

En el siguiente ejemplo de código, se muestra cómo usar `delete-nat-gateway`.

### AWS CLI

Para eliminar una NAT puerta de enlace

En este ejemplo se elimina la NAT puerta de enlace `nat-04ae55e711cec5680`.

Comando:

```
aws ec2 delete-nat-gateway --nat-gateway-id nat-04ae55e711cec5680
```

Salida:

```
{  
  "NatGatewayId": "nat-04ae55e711cec5680"  
}
```

- Para API obtener más información, consulte [DeleteNatGateway](#) la Referencia de AWS CLI comandos.

## delete-network-acl-entry

En el siguiente ejemplo de código, se muestra cómo usar `delete-network-acl-entry`.

### AWS CLI

Para eliminar una ACL entrada de red

En este ejemplo, se elimina la regla de entrada número 100 de la red especificada. ACL Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-network-acl-entry --network-acl-id acl-5fb85d36 --ingress --rule-number 100
```

- Para API obtener más información, consulte la Referencia [DeleteNetworkAclEntry](#) de AWS CLI comandos.

## **delete-network-acl**

En el siguiente ejemplo de código, se muestra cómo usar `delete-network-acl`.

AWS CLI

Para eliminar una red ACL

En este ejemplo, se elimina la red ACL especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-network-acl --network-acl-id acl-5fb85d36
```

- Para API obtener más información, consulte [DeleteNetworkAcl](#) la Referencia de AWS CLI comandos.

## **delete-network-insights-access-scope-analysis**

En el siguiente ejemplo de código, se muestra cómo usar `delete-network-insights-access-scope-analysis`.

AWS CLI

Para eliminar un análisis del alcance de acceso a la red

En el siguiente `delete-network-insights-access-scope-analysis` ejemplo, se elimina el análisis del alcance de acceso a la red especificado.

```
aws ec2 delete-network-insights-access-scope-analysis \  
--network-insights-access-scope-analysis-id nisa-01234567891abcdef
```

Salida:

```
{  
  "NetworkInsightsAccessScopeAnalysisId": "nisa-01234567891abcdef"  
}
```

Para obtener más información, consulte [Introducción al analizador de acceso a la red mediante la guía del analizador AWS CLI](#) de acceso a la red.

- Para API obtener más información, consulte la Referencia [DeleteNetworkInsightsAccessScopeAnalysis](#) de AWS CLI comandos.

## **delete-network-insights-access-scope**

En el siguiente ejemplo de código, se muestra cómo usar `delete-network-insights-access-scope`.

AWS CLI

Para eliminar un ámbito de acceso a la red

En el siguiente `delete-network-insights-access-scope` ejemplo, se elimina el ámbito de acceso a la red especificado.

```
aws ec2 delete-network-insights-access-scope \  
--network-insights-access-scope-id nis-123456789abc01234
```

Salida:

```
{  
  "NetworkInsightsAccessScopeId": "nis-123456789abc01234"  
}
```

Para obtener más información, consulte [Introducción al analizador de acceso a la red mediante la Guía del analizador AWS CLI](#) de acceso a la red.

- Para API obtener más información, consulte la Referencia [DeleteNetworkInsightsAccessScope](#) de AWS CLI comandos.

## delete-network-insights-analysis

En el siguiente ejemplo de código, se muestra cómo usar `delete-network-insights-analysis`.

### AWS CLI

Para eliminar un análisis de ruta

En el siguiente `delete-network-insights-analysis` ejemplo, se elimina el análisis especificado.

```
aws ec2 delete-network-insights-analysis \  
  --network-insights-analysis-id nia-02207aa13eb480c7a
```

Salida:

```
{  
  "NetworkInsightsAnalysisId": "nia-02207aa13eb480c7a"  
}
```

Para obtener más información, consulte [Cómo empezar a utilizar la Guía del AWS CLI Reachability Analyzer](#).

- Para API obtener más información, consulte la Referencia de [DeleteNetworkInsightsAnalysis](#) comandos AWS CLI .

## delete-network-insights-path

En el siguiente ejemplo de código, se muestra cómo usar `delete-network-insights-path`.

### AWS CLI

Para eliminar una ruta

En el siguiente `delete-network-insights-path` ejemplo, se elimina la ruta especificada. Antes de poder eliminar una ruta, debe eliminar todos sus análisis mediante el `delete-network-insights-analysis` comando.

```
aws ec2 delete-network-insights-path \  
  --network-insights-path-id nip-0b26f224f1d131fa8
```

Salida:

```
{
  "NetworkInsightsPathId": "nip-0b26f224f1d131fa8"
}
```

Para obtener más información, consulte [Cómo empezar a utilizar la Guía del AWS CLI Reachability Analyzer](#).

- Para API obtener más información, consulte la Referencia de [DeleteNetworkInsightsPath](#) comandos AWS CLI .

## **delete-network-interface-permission**

En el siguiente ejemplo de código, se muestra cómo usar `delete-network-interface-permission`.

AWS CLI

Para eliminar un permiso de interfaz de red

En este ejemplo se elimina el permiso de interfaz de red especificado.

Comando:

```
aws ec2 delete-network-interface-permission --network-interface-permission-id eni-  
perm-06fd19020ede149ea
```

Salida:

```
{
  "Return": true
}
```

- Para API obtener más información, consulte [DeleteNetworkInterfacePermission](#) la Referencia de AWS CLI comandos.

## **delete-network-interface**

En el siguiente ejemplo de código, se muestra cómo usar `delete-network-interface`.

## AWS CLI

Para eliminar una interfaz de red

En este ejemplo se elimina la interfaz de red especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-network-interface --network-interface-id eni-e5aa89a3
```

- Para API obtener más información, consulte [DeleteNetworkInterface](#) la Referencia de AWS CLI comandos.

## **delete-placement-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-placement-group`.

## AWS CLI

Para eliminar un grupo de ubicaciones

Este comando de ejemplo elimina el grupo de ubicación especificado.

Comando:

```
aws ec2 delete-placement-group --group-name my-cluster
```

- Para API obtener más información, consulte [DeletePlacementGroup](#) la Referencia de AWS CLI comandos.

## **delete-queued-reserved-instances**

En el siguiente ejemplo de código, se muestra cómo usar `delete-queued-reserved-instances`.

## AWS CLI

Para eliminar una compra en cola

En el siguiente `delete-queued-reserved-instances` ejemplo, se elimina la instancia reservada especificada, que estaba en cola para su compra.

```
aws ec2 delete-queued-reserved-instances \  
--reserved-instances-ids af9f760e-6f91-4559-85f7-4980example
```

Salida:

```
{  
  "SuccessfulQueuedPurchaseDeletions": [  
    {  
      "ReservedInstancesId": "af9f760e-6f91-4559-85f7-4980example"  
    }  
  ],  
  "FailedQueuedPurchaseDeletions": []  
}
```

- Para API obtener más información, consulte la Referencia [DeleteQueuedReservedInstances](#) de AWS CLI comandos.

## **delete-route-table**

En el siguiente ejemplo de código, se muestra cómo usar `delete-route-table`.

AWS CLI

Para eliminar una tabla de rutas

En este ejemplo, se elimina la tabla de rutas especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-route-table --route-table-id rtb-22574640
```

- Para API obtener más información, consulte [DeleteRouteTable](#) la Referencia de AWS CLI comandos.

## **delete-route**

En el siguiente ejemplo de código, se muestra cómo usar `delete-route`.

## AWS CLI

Para eliminar una ruta

En este ejemplo, se elimina la ruta especificada de la tabla de rutas especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-route --route-table-id rtb-22574640 --destination-cidr-  
block 0.0.0.0/0
```

- Para API obtener más información, consulte [DeleteRoute](#) la Referencia de AWS CLI comandos.

## delete-security-group

En el siguiente ejemplo de código, se muestra cómo usar delete-security-group.

### AWS CLI

[EC2-Classical] Para eliminar un grupo de seguridad

En este ejemplo, se elimina el grupo de seguridad denominado MySecurityGroup. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-security-group --group-name MySecurityGroup
```

[EC2-VPC] Para eliminar un grupo de seguridad

En este ejemplo, se elimina el grupo de seguridad con el ID sg-903004f8. Tenga en cuenta que no puede hacer referencia a un grupo de seguridad para EC2... VPC por su nombre. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-security-group --group-id sg-903004f8
```

Para obtener más información, consulte [Uso de los grupos de seguridad](#) en la Guía del usuario de la Interfaz de la línea de comandos de AWS .



- Para API obtener más información, consulte [DeleteSecurityGroup](#) la Referencia de AWS CLI comandos.

## **delete-snapshot**

En el siguiente ejemplo de código, se muestra cómo usar `delete-snapshot`.

### AWS CLI

Eliminar una instantánea

Este comando de ejemplo elimina la instantánea con el ID de instantánea de `snap-1234567890abcdef0`. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-snapshot --snapshot-id snap-1234567890abcdef0
```

- Para API obtener más información, consulte [DeleteSnapshot](#) la Referencia de AWS CLI comandos.

## **delete-spot-datafeed-subscription**

En el siguiente ejemplo de código, se muestra cómo usar `delete-spot-datafeed-subscription`.

### AWS CLI

Para cancelar una suscripción a una fuente de datos de Spot Instance

Este comando de ejemplo elimina una suscripción a una fuente de datos de Spot de la cuenta. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-spot-datafeed-subscription
```

- Para API obtener más información, consulte [DeleteSpotDatafeedSubscription](#) la Referencia de AWS CLI comandos.

## delete-subnet-cidr-reservation

En el siguiente ejemplo de código, se muestra cómo usar `delete-subnet-cidr-reservation`.

### AWS CLI

Para eliminar una reserva de subred CIDR

En el siguiente `delete-subnet-cidr-reservation` ejemplo, se elimina la reserva de CIDR subred especificada.

```
aws ec2 delete-subnet-cidr-reservation \  
--subnet-cidr-reservation-id scr-044f977c4eEXAMPLE
```

Salida:

```
{  
  "DeletedSubnetCidrReservation": {  
    "SubnetCidrReservationId": "scr-044f977c4eEXAMPLE",  
    "SubnetId": "subnet-03c51e2e6cEXAMPLE",  
    "Cidr": "10.1.0.16/28",  
    "ReservationType": "prefix",  
    "OwnerId": "123456789012"  
  }  
}
```

Para obtener más información, consulta [CIDRReservas de subred](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DeleteSubnetCidrReservation](#) la Referencia de AWS CLI comandos.

## delete-subnet

En el siguiente ejemplo de código, se muestra cómo usar `delete-subnet`.

### AWS CLI

Para eliminar una subred

En este ejemplo, se elimina la subred especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-subnet --subnet-id subnet-9d4a7b6c
```

- Para API obtener más información, consulte la Referencia [DeleteSubnet](#) de AWS CLI comandos.

## delete-tags

En el siguiente ejemplo de código, se muestra cómo usar `delete-tags`.

AWS CLI

Ejemplo 1: Para eliminar una etiqueta de un recurso

En el siguiente `delete-tags` ejemplo, se elimina la etiqueta `Stack=Test` de la imagen especificada. Al especificar un valor y un nombre de clave, la etiqueta se elimina solo si el valor de la etiqueta coincide con el valor especificado.

```
aws ec2 delete-tags \  
  --resources ami-1234567890abcdef0 \  
  --tags Key=Stack,Value=Test
```

Es opcional especificar el valor de una etiqueta. En el siguiente `delete-tags` ejemplo, se elimina la etiqueta con el nombre `purpose` de clave de la instancia especificada, independientemente del valor de la etiqueta.

```
aws ec2 delete-tags \  
  --resources i-1234567890abcdef0 \  
  --tags Key=purpose
```

Si especifica la cadena vacía como el valor de la etiqueta, la etiqueta se eliminará solo si el valor de la etiqueta es la cadena vacía. En el siguiente `delete-tags` ejemplo, se especifica la cadena vacía como valor de etiqueta de la etiqueta que se va a eliminar.

```
aws ec2 delete-tags \  
  --resources i-1234567890abcdef0 \  
  --tags Key=Name,Value=
```

Ejemplo 2: Para eliminar una etiqueta de varios recursos

En el siguiente `delete-tags` ejemplo, se elimina la etiqueta ```Purpose=Test``` tanto de una instancia como de una AMI. Como se muestra en el ejemplo anterior, puedes omitir el valor de la etiqueta en el comando.

```
aws ec2 delete-tags \  
  --resources i-1234567890abcdef0 ami-1234567890abcdef0 \  
  --tags Key=Purpose
```

- Para API obtener más información, consulte [DeleteTags](#) la Referencia de AWS CLI comandos.

## **delete-traffic-mirror-filter-rule**

En el siguiente ejemplo de código, se muestra cómo usar `delete-traffic-mirror-filter-rule`.

### AWS CLI

Para eliminar una regla de filtrado de réplica de tráfico

En el siguiente `delete-traffic-mirror-filter-rule` ejemplo, se elimina la regla de filtro de reflejo de tráfico especificada.

```
aws ec2 delete-traffic-mirror-filter-rule \  
  --traffic-mirror-filter-rule-id tmfr-081f71283bEXAMPLE
```

Salida:

```
{  
  "TrafficMirrorFilterRuleId": "tmfr-081f71283bEXAMPLE"  
}
```

Para obtener más información, consulte [Modificar las reglas del filtro de espejo de tráfico](#) en la Guía de duplicación de AWS tráfico.

- Para API obtener más información, consulte la Referencia [DeleteTrafficMirrorFilterRule](#) de AWS CLI comandos.

## **delete-traffic-mirror-filter**

En el siguiente ejemplo de código, se muestra cómo usar `delete-traffic-mirror-filter`.

## AWS CLI

Para eliminar un filtro espejo de tráfico

En el siguiente `delete-traffic-mirror-filter` ejemplo, se elimina el filtro de reflejo de tráfico especificado.

```
aws ec2 delete-traffic-mirror-filter \  
  --traffic-mirror-filter-id tmf-0be0b25fcdEXAMPLE
```

Salida:

```
{  
  "TrafficMirrorFilterId": "tmf-0be0b25fcdEXAMPLE"  
}
```

Para obtener más información, consulte [Eliminar un filtro de espejo de tráfico](#) en la Guía de duplicación de AWS tráfico.

- Para API obtener más información, consulte la Referencia [DeleteTrafficMirrorFilter](#) de AWS CLI comandos.

## `delete-traffic-mirror-session`

En el siguiente ejemplo de código, se muestra cómo usar `delete-traffic-mirror-session`.

## AWS CLI

Para eliminar una sesión duplicada de tráfico

En el siguiente `delete-traffic-mirror-session` ejemplo, se elimina la sesión reflejada de tráfico especificada.

```
aws ec2 delete-traffic-mirror-session \  
  --traffic-mirror-session-id tms-0af3141ce5EXAMPLE
```

Salida:

```
{
```

```
"TrafficMirrorSessionId": "tms-0af3141ce5EXAMPLE"  
}
```

Para obtener más información, consulte [Eliminar una sesión de duplicación de tráfico en la AWS Guía de duplicación de tráfico](#).

- Para API obtener más información, consulte la Referencia [DeleteTrafficMirrorSession](#) de AWS CLI comandos.

## **delete-traffic-mirror-target**

En el siguiente ejemplo de código, se muestra cómo usar `delete-traffic-mirror-target`.

### AWS CLI

Para eliminar un objetivo reflejado de tráfico

En el siguiente `delete-traffic-mirror-target` ejemplo, se elimina el objetivo de reflejo de tráfico especificado.

```
aws ec2 delete-traffic-mirror-target \  
  --traffic-mirror-target-id tmt-060f48ce9EXAMPLE
```

Salida:

```
{  
  "TrafficMirrorTargetId": "tmt-060f48ce9EXAMPLE"  
}
```

Para obtener más información, consulte [Eliminar un objetivo de espejo de tráfico](#) en la Guía de duplicación de AWS tráfico.

- Para API obtener más información, consulte la Referencia [DeleteTrafficMirrorTarget](#) de AWS CLI comandos.

## **delete-transit-gateway-connect-peer**

En el siguiente ejemplo de código, se muestra cómo usar `delete-transit-gateway-connect-peer`.

## AWS CLI

Para eliminar un par de Transit Gateway Connect

En el siguiente `delete-transit-gateway-connect-peer` ejemplo, se elimina el par Connect especificado.

```
aws ec2 delete-transit-gateway-connect-peer \
  --transit-gateway-connect-peer-id tgw-connect-peer-0666adbac4EXAMPLE
```

Salida:

```
{
  "TransitGatewayConnectPeer": {
    "TransitGatewayAttachmentId": "tgw-attach-0f0927767cEXAMPLE",
    "TransitGatewayConnectPeerId": "tgw-connect-peer-0666adbac4EXAMPLE",
    "State": "deleting",
    "CreationTime": "2021-10-13T03:35:17.000Z",
    "ConnectPeerConfiguration": {
      "TransitGatewayAddress": "10.0.0.234",
      "PeerAddress": "172.31.1.11",
      "InsideCidrBlocks": [
        "169.254.6.0/29"
      ],
      "Protocol": "gre",
      "BgpConfigurations": [
        {
          "TransitGatewayAsn": 64512,
          "PeerAsn": 64512,
          "TransitGatewayAddress": "169.254.6.2",
          "PeerAddress": "169.254.6.1",
          "BgpStatus": "down"
        },
        {
          "TransitGatewayAsn": 64512,
          "PeerAsn": 64512,
          "TransitGatewayAddress": "169.254.6.3",
          "PeerAddress": "169.254.6.1",
          "BgpStatus": "down"
        }
      ]
    }
  }
}
```

```
}
```

Para obtener más información, consulte los [archivos adjuntos de Transit Gateway Connect y los pares de Transit Gateway Connect](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [DeleteTransitGatewayConnectPeer](#) la Referencia de AWS CLI comandos.

## delete-transit-gateway-connect

En el siguiente ejemplo de código, se muestra cómo usar delete-transit-gateway-connect.

### AWS CLI

Para eliminar un archivo adjunto de Transit Gateway Connect

En el siguiente delete-transit-gateway-connect ejemplo, se elimina el adjunto Connect especificado.

```
aws ec2 delete-transit-gateway-connect \
  --transit-gateway-attachment-id tgw-attach-037012e5dcEXAMPLE
```

Salida:

```
{
  "TransitGatewayConnect": {
    "TransitGatewayAttachmentId": "tgw-attach-037012e5dcEXAMPLE",
    "TransportTransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
    "State": "deleting",
    "CreationTime": "2021-03-09T19:59:17+00:00",
    "Options": {
      "Protocol": "gre"
    }
  }
}
```

Para obtener más información, consulte los [archivos adjuntos de Transit Gateway Connect y los pares de Transit Gateway Connect](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [DeleteTransitGatewayConnect](#) la Referencia de AWS CLI comandos.



## delete-transit-gateway-multicast-domain

En el siguiente ejemplo de código, se muestra cómo usar `delete-transit-gateway-multicast-domain`.

### AWS CLI

Para eliminar un dominio de multidifusión de Transit Gateway

El siguiente `delete-transit-gateway-multicast-domain` ejemplo elimina el dominio de multidifusión especificado.

```
aws ec2 delete-transit-gateway-multicast-domain \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE
```

Salida:

```
{
  "TransitGatewayMulticastDomain": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-02bb79002bEXAMPLE",
    "TransitGatewayId": "tgw-0d88d2d0d5EXAMPLE",
    "State": "deleting",
    "CreationTime": "2019-11-20T22:02:03.000Z"
  }
}
```

Para obtener más información, consulte [Administración de dominios de multidifusión](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [DeleteTransitGatewayMulticastDomain](#) de AWS CLI comandos.

## delete-transit-gateway-peering-attachment

En el siguiente ejemplo de código, se muestra cómo usar `delete-transit-gateway-peering-attachment`.

### AWS CLI

Para eliminar un adjunto de peering de Transit Gateway

En el siguiente `delete-transit-gateway-peering-attachment` ejemplo, se elimina el adjunto de interconexión de la pasarela de tránsito especificado.

```
aws ec2 delete-transit-gateway-peering-attachment \  
--transit-gateway-attachment-id tgw-attach-4455667788aabbccd
```

Salida:

```
{  
  "TransitGatewayPeeringAttachment": {  
    "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",  
    "RequesterTgwInfo": {  
      "TransitGatewayId": "tgw-123abc05e04123abc",  
      "OwnerId": "123456789012",  
      "Region": "us-west-2"  
    },  
    "AcceptorTgwInfo": {  
      "TransitGatewayId": "tgw-11223344aabbcc112",  
      "OwnerId": "123456789012",  
      "Region": "us-east-2"  
    },  
    "State": "deleting",  
    "CreationTime": "2019-12-09T11:38:31.000Z"  
  }  
}
```

Para obtener más información, consulte los [adjuntos de peering de Transit Gateway](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [DeleteTransitGatewayPeeringAttachment](#) de AWS CLI comandos.

## **delete-transit-gateway-policy-table**

En el siguiente ejemplo de código, se muestra cómo usar `delete-transit-gateway-policy-table`.

AWS CLI

Para eliminar una tabla de políticas de una pasarela de tránsito

El siguiente `delete-transit-gateway-policy-table` ejemplo elimina la tabla de políticas de la pasarela de tránsito especificada.

```
aws ec2 delete-transit-gateway-policy-table \  
  --transit-gateway-policy-table-id tgw-ptb-0a16f134b78668a81
```

Salida:

```
{  
  "TransitGatewayPolicyTables": [  
    {  
      "TransitGatewayPolicyTableId": "tgw-ptb-0a16f134b78668a81",  
      "TransitGatewayId": "tgw-067f8505c18f0bd6e",  
      "State": "deleting",  
      "CreationTime": "2023-11-28T16:36:43+00:00",  
      "Tags": []  
    }  
  ]  
}
```

Para obtener más información, consulte [las tablas de políticas de Transit Gateway](#) en la Guía del usuario de Transit Gateway.

- Para API obtener más información, consulte [DeleteTransitGatewayPolicyTable](#) la Referencia de AWS CLI comandos.

## **delete-transit-gateway-prefix-list-reference**

En el siguiente ejemplo de código, se muestra cómo usar `delete-transit-gateway-prefix-list-reference`.

AWS CLI

Para eliminar una referencia a una lista de prefijos

En el siguiente `delete-transit-gateway-prefix-list-reference` ejemplo, se elimina la referencia de la lista de prefijos especificada.

```
aws ec2 delete-transit-gateway-prefix-list-reference \  
  --transit-gateway-route-table-id tgw-rtb-0123456789abcd123 \  
  --prefix-list-id pl-11111122222222333
```

Salida:

```
{
  "TransitGatewayPrefixListReference": {
    "TransitGatewayRouteTableId": "tgw-rtb-0123456789abcd123",
    "PrefixListId": "pl-1111112222222333",
    "PrefixListOwnerId": "123456789012",
    "State": "deleting",
    "Blackhole": false,
    "TransitGatewayAttachment": {
      "TransitGatewayAttachmentId": "tgw-attach-aabbccddaabbccaab",
      "ResourceType": "vpc",
      "ResourceId": "vpc-112233445566aabbcc"
    }
  }
}
```

Para obtener más información, consulte las [referencias a las listas de prefijos](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [DeleteTransitGatewayPrefixListReference](#) de AWS CLI comandos.

## delete-transit-gateway-route-table

En el siguiente ejemplo de código, se muestra cómo usar `delete-transit-gateway-route-table`.

AWS CLI

Para eliminar una tabla de rutas de una puerta de enlace de tránsito

El siguiente `delete-transit-gateway-route-table` ejemplo elimina la tabla de rutas de la puerta de enlace de tránsito especificada.

```
aws ec2 delete-transit-gateway-route-table \
  --transit-gateway-route-table-id tgw-rtb-0b6f6aaa01EXAMPLE
```

Salida:

```
{
  "TransitGatewayRouteTable": {
```

```

    "TransitGatewayRouteTableId": "tgw-rtb-0b6f6aaa01EXAMPLE",
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
    "State": "deleting",
    "DefaultAssociationRouteTable": false,
    "DefaultPropagationRouteTable": false,
    "CreationTime": "2019-07-17T20:27:26.000Z"
  }
}

```

Para obtener más información, consulte [Eliminar una tabla de rutas de una pasarela de tránsito](#) en la Guía de pasarelas de tránsito.

- Para API obtener más información, consulte [DeleteTransitGatewayRouteTable](#) la Referencia de AWS CLI comandos.

## delete-transit-gateway-route

En el siguiente ejemplo de código, se muestra cómo usar `delete-transit-gateway-route`.

### AWS CLI

Para eliminar un CIDR bloque de una tabla de enrutamiento

El siguiente `delete-transit-gateway-route` ejemplo elimina el CIDR bloque de la tabla de rutas de la puerta de enlace de tránsito especificada.

```

aws ec2 delete-transit-gateway-route \
  --transit-gateway-route-table-id tgw-rtb-0b6f6aaa01EXAMPLE \
  --destination-cidr-block 10.0.2.0/24

```

Salida:

```

{
  "Route": {
    "DestinationCidrBlock": "10.0.2.0/24",
    "TransitGatewayAttachments": [
      {
        "ResourceId": "vpc-0065acced4EXAMPLE",
        "TransitGatewayAttachmentId": "tgw-attach-0b5968d3b6EXAMPLE",
        "ResourceType": "vpc"
      }
    ]
  },
}

```

```
    "Type": "static",
    "State": "deleted"
  }
}
```

Para obtener más información, consulte [Eliminar una ruta estática](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [DeleteTransitGatewayRoute](#) la Referencia de AWS CLI comandos.

## delete-transit-gateway-vpc-attachment

En el siguiente ejemplo de código, se muestra cómo usar `delete-transit-gateway-vpc-attachment`.

### AWS CLI

Para eliminar un VPC archivo adjunto a una pasarela de tránsito

En el siguiente `delete-transit-gateway-vpc-attachment` ejemplo, se elimina el VPC adjunto especificado.

```
aws ec2 delete-transit-gateway-vpc-attachment \
  --transit-gateway-attachment-id tgw-attach-0d2c54bdbEXAMPLE
```

Salida:

```
{
  "TransitGatewayVpcAttachment": {
    "TransitGatewayAttachmentId": "tgw-attach-0d2c54bdb3EXAMPLE",
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
    "VpcId": "vpc-0065acced4f61c651",
    "VpcOwnerId": "111122223333",
    "State": "deleting",
    "CreationTime": "2019-07-17T16:04:27.000Z"
  }
}
```

Para obtener más información, consulte [Eliminar un VPC archivo adjunto](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [DeleteTransitGatewayVpcAttachment](#) la Referencia de AWS CLI comandos.

## delete-transit-gateway

En el siguiente ejemplo de código, se muestra cómo usar `delete-transit-gateway`.

### AWS CLI

Para eliminar una puerta de enlace de tránsito

El siguiente `delete-transit-gateway` ejemplo elimina la puerta de enlace de tránsito especificada.

```
aws ec2 delete-transit-gateway \  
--transit-gateway-id tgw-01f04542b2EXAMPLE
```

Salida:

```
{  
  "TransitGateway": {  
    "TransitGatewayId": "tgw-01f04542b2EXAMPLE",  
    "State": "deleting",  
    "OwnerId": "123456789012",  
    "Description": "Example Transit Gateway",  
    "CreationTime": "2019-08-27T15:04:35.000Z",  
    "Options": {  
      "AmazonSideAsn": 64515,  
      "AutoAcceptSharedAttachments": "disable",  
      "DefaultRouteTableAssociation": "enable",  
      "AssociationDefaultRouteTableId": "tgw-rtb-0ce7a6948fEXAMPLE",  
      "DefaultRouteTablePropagation": "enable",  
      "PropagationDefaultRouteTableId": "tgw-rtb-0ce7a6948fEXAMPLE",  
      "VpnEcmpSupport": "enable",  
      "DnsSupport": "enable"  
    }  
  }  
}
```

Para obtener más información, consulte [Eliminar una pasarela de tránsito](#) en la Guía de pasarelas de tránsito.

- Para API obtener más información, consulte [Delete Transit Gateway](#) la Referencia de AWS CLI comandos.

## delete-verified-access-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `delete-verified-access-endpoint`.

AWS CLI

Para eliminar un punto final de Verified Access

En el siguiente `delete-verified-access-endpoint` ejemplo, se elimina el punto final de Verified Access especificado.

```
aws ec2 delete-verified-access-endpoint \
  --verified-access-endpoint-id vae-066fac616d4d546f2
```

Salida:

```
{
  "VerifiedAccessEndpoint": {
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
    "VerifiedAccessEndpointId": "vae-066fac616d4d546f2",
    "ApplicationDomain": "example.com",
    "EndpointType": "network-interface",
    "AttachmentType": "vpc",
    "DomainCertificateArn": "arn:aws:acm:us-east-2:123456789012:certificate/
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE",
    "EndpointDomain": "my-ava-
app.edge-00c3372d53b1540bb.vai-0ce000c0b7643abea.prod.verified-access.us-
east-2.amazonaws.com",
    "SecurityGroupIds": [
      "sg-004915970c4c8f13a"
    ],
    "NetworkInterfaceOptions": {
      "NetworkInterfaceId": "eni-0aec70418c8d87a0f",
      "Protocol": "https",
      "Port": 443
    },
    "Status": {
      "Code": "deleting"
    }
  }
}
```



```
    },
    "Description": "Testing Verified Access",
    "CreationTime": "2023-08-25T20:54:43",
    "LastUpdatedTime": "2023-08-25T22:46:32"
  }
}
```

Para obtener más información, consulte los [puntos finales de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [DeleteVerifiedAccessEndpoint](#) la Referencia de AWS CLI comandos.

## delete-verified-access-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-verified-access-group`.

### AWS CLI

Para eliminar un grupo de acceso verificado

En el siguiente `delete-verified-access-group` ejemplo, se elimina el grupo de acceso verificado especificado.

```
aws ec2 delete-verified-access-group \
  --verified-access-group-id vagr-0dbe967baf14b7235
```

Salida:

```
{
  "VerifiedAccessGroup": {
    "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "Description": "Testing Verified Access",
    "Owner": "123456789012",
    "VerifiedAccessGroupArn": "arn:aws:ec2:us-east-2:123456789012:verified-
access-group/vagr-0dbe967baf14b7235",
    "CreationTime": "2023-08-25T19:55:19",
    "LastUpdatedTime": "2023-08-25T22:49:03",
    "DeletionTime": "2023-08-26T00:58:31"
  }
}
```

Para obtener más información, consulte los [grupos de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [DeleteVerifiedAccessGroup](#) la Referencia de AWS CLI comandos.

## **delete-verified-access-instance**

En el siguiente ejemplo de código, se muestra cómo usar `delete-verified-access-instance`.

### AWS CLI

Para eliminar una instancia de Verified Access

En el siguiente `delete-verified-access-instance` ejemplo, se elimina la instancia de acceso verificado especificada.

```
aws ec2 delete-verified-access-instance \  
  --verified-access-instance-id vai-0ce000c0b7643abea
```

Salida:

```
{  
  "VerifiedAccessInstance": {  
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",  
    "Description": "Testing Verified Access",  
    "VerifiedAccessTrustProviders": [],  
    "CreationTime": "2023-08-25T18:27:56",  
    "LastUpdatedTime": "2023-08-26T01:00:18"  
  }  
}
```

Para obtener más información, consulte las [instancias de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [DeleteVerifiedAccessInstance](#) la Referencia de AWS CLI comandos.

## **delete-verified-access-trust-provider**

En el siguiente ejemplo de código, se muestra cómo usar `delete-verified-access-trust-provider`.

## AWS CLI

Para eliminar un proveedor de confianza de Verified Access

En el siguiente `delete-verified-access-trust-provider` ejemplo, se elimina el proveedor de confianza de Verified Access especificado.

```
aws ec2 delete-verified-access-trust-provider \  
--verified-access-trust-provider-id vatp-0bb32de759a3e19e7
```

Salida:

```
{  
  "VerifiedAccessTrustProvider": {  
    "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",  
    "Description": "Testing Verified Access",  
    "TrustProviderType": "user",  
    "UserTrustProviderType": "iam-identity-center",  
    "PolicyReferenceName": "idc",  
    "CreationTime": "2023-08-25T18:40:36",  
    "LastUpdatedTime": "2023-08-25T18:40:36"  
  }  
}
```

Para obtener más información, consulte [Proveedores de confianza para el acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [DeleteVerifiedAccessTrustProvider](#) la Referencia de AWS CLI comandos.

## **delete-volume**

En el siguiente ejemplo de código, se muestra cómo usar `delete-volume`.

## AWS CLI

Para eliminar un volumen

Este comando de ejemplo elimina un volumen disponible con el identificador de volumen `devol-049df61146c4d7901`. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-volume --volume-id vol-049df61146c4d7901
```

- Para API obtener más información, consulte [DeleteVolume](#) la Referencia de AWS CLI comandos.

## **delete-vpc-endpoint-connection-notifications**

En el siguiente ejemplo de código, se muestra cómo usar `delete-vpc-endpoint-connection-notifications`.

AWS CLI

Para eliminar una notificación de conexión de punto final

En este ejemplo, se elimina la notificación de conexión de punto final especificada.

Comando:

```
aws ec2 delete-vpc-endpoint-connection-notifications --connection-notification-ids vpce-nfn-008776de7e03f5abc
```

Salida:

```
{
  "Unsuccessful": []
}
```

- Para API obtener más información, consulte [DeleteVpcEndpointConnectionNotifications](#) la Referencia de AWS CLI comandos.

## **delete-vpc-endpoint-service-configurations**

En el siguiente ejemplo de código, se muestra cómo usar `delete-vpc-endpoint-service-configurations`.

AWS CLI

Para eliminar la configuración de un servicio de punto final

En este ejemplo, se elimina la configuración del servicio de punto final especificada.

Comando:

```
aws ec2 delete-vpc-endpoint-service-configurations --service-ids vpce-  
svc-03d5ebb7d9579a2b3
```

Salida:

```
{  
  "Unsuccessful": []  
}
```

- Para API obtener más información, consulte [DeleteVpcEndpointServiceConfigurations](#) la Referencia de AWS CLI comandos.

## delete-vpc-endpoints

En el siguiente ejemplo de código, se muestra cómo usar `delete-vpc-endpoints`.

AWS CLI

Para eliminar un punto final

En este ejemplo, se eliminan los puntos finales `vpce-aa22bb33` y `vpce-1a2b3c4d`. Si el comando se ejecuta correctamente parcialmente o no se ejecuta correctamente, se devuelve una lista de los elementos fallidos. Si el comando se ejecuta correctamente, la lista devuelta está vacía.

Comando:

```
aws ec2 delete-vpc-endpoints --vpc-endpoint-ids vpce-aa22bb33 vpce-1a2b3c4d
```

Salida:

```
{  
  "Unsuccessful": []  
}
```

- Para API obtener más información, consulte [DeleteVpcEndpoints](#) la Referencia de AWS CLI comandos.

## delete-vpc-peering-connection

En el siguiente ejemplo de código, se muestra cómo usar `delete-vpc-peering-connection`.

### AWS CLI

Para eliminar una conexión VPC entre pares

En este ejemplo, se elimina la conexión de VPC emparejamiento especificada.

Comando:

```
aws ec2 delete-vpc-peering-connection --vpc-peering-connection-id pcx-1a2b3c4d
```

Salida:

```
{
  "Return": true
}
```

- Para API obtener más información, consulte la Referencia [DeleteVpcPeeringConnection](#) de AWS CLI comandos.

## delete-vpc

En el siguiente ejemplo de código, se muestra cómo usar `delete-vpc`.

### AWS CLI

Para eliminar un VPC

En este ejemplo se elimina lo especificado VPC. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-vpc --vpc-id vpc-a01106c2
```

- Para API obtener más información, consulte [DeleteVpc](#) la Referencia de AWS CLI comandos.

## delete-vpn-connection-route

En el siguiente ejemplo de código, se muestra cómo usar `delete-vpn-connection-route`.

### AWS CLI

Para eliminar una ruta estática de una VPN conexión

En este ejemplo, se elimina la ruta estática especificada de la VPN conexión especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-vpn-connection-route --vpn-connection-id vpn-40f41529 --destination-cidr-block 11.12.0.0/16
```

- Para API obtener más información, consulte [DeleteVpnConnectionRoute](#) la Referencia de AWS CLI comandos.

## delete-vpn-connection

En el siguiente ejemplo de código, se muestra cómo usar `delete-vpn-connection`.

### AWS CLI

Para eliminar una VPN conexión

En este ejemplo, se elimina la VPN conexión especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-vpn-connection --vpn-connection-id vpn-40f41529
```

- Para API obtener más información, consulte [DeleteVpnConnection](#) la Referencia de AWS CLI comandos.

## delete-vpn-gateway

En el siguiente ejemplo de código, se muestra cómo usar `delete-vpn-gateway`.

## AWS CLI

Para eliminar una puerta de enlace privada virtual

En este ejemplo, se elimina la puerta de enlace privada virtual especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 delete-vpn-gateway --vpn-gateway-id vgw-9a4caf3
```

- Para API obtener más información, consulte [DeleteVpnGateway](#) la Referencia de AWS CLI comandos.

## deprovision-byoip-cidr

En el siguiente ejemplo de código, se muestra cómo usar `deprovision-byoip-cidr`.

## AWS CLI

Para eliminar un rango de direcciones IP del uso

En el siguiente ejemplo, se quita el rango de direcciones especificado para que no se utilice con AWS.

```
aws ec2 deprovision-byoip-cidr \  
  --cidr 203.0.113.25/24
```

Salida:

```
{  
  "ByoipCidr": {  
    "Cidr": "203.0.113.25/24",  
    "State": "pending-deprovision"  
  }  
}
```

- Para API obtener más información, consulte [DeprovisionByoipCidr](#) la Referencia de AWS CLI comandos.



## deprovision-ipam-pool-cidr

En el siguiente ejemplo de código, se muestra cómo usar `deprovision-ipam-pool-cidr`.

### AWS CLI

Para desaproveccionar un grupo IPAM CIDR

En el siguiente `deprovision-ipam-pool-cidr` ejemplo, se desaprovecciona un CIDR aprovisionamiento en un grupo. IPAM

(Linux):

```
aws ec2 deprovision-ipam-pool-cidr \  
  --ipam-pool-id ipam-pool-02ec043a19bbe5d08 \  
  --cidr 11.0.0.0/16
```

(Windows):

```
aws ec2 deprovision-ipam-pool-cidr ^  
  --ipam-pool-id ipam-pool-02ec043a19bbe5d08 ^  
  --cidr 11.0.0.0/16
```

Salida:

```
{  
  "IpamPoolCidr": {  
    "Cidr": "11.0.0.0/16",  
    "State": "pending-deprovision"  
  }  
}
```

Para obtener más información, consulta [Desaproveccionar un grupo CIDRs](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [DeprovisionIpamPoolCidr](#) la Referencia de AWS CLI comandos.

## deregister-image

En el siguiente ejemplo de código, se muestra cómo usar `deregister-image`.

## AWS CLI

Para anular el registro de un AMI

En este ejemplo se anula el registro de lo especificado. AMI Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 deregister-image --image-id ami-4fa54026
```

- Para API obtener más información, consulte la Referencia [DeregisterImage](#) de AWS CLI comandos.

## deregister-instance-event-notification-attributes

En el siguiente ejemplo de código, se muestra cómo usar `deregister-instance-event-notification-attributes`.

### AWS CLI

Ejemplo 1: Para eliminar todas las etiquetas de las notificaciones de eventos

En el siguiente `deregister-instance-event-notification-attributes` ejemplo, se elimina `IncludeAllTagsOfInstance=true`, lo que tiene el efecto de `IncludeAllTagsOfInstance` establecer en `false`.

```
aws ec2 deregister-instance-event-notification-attributes \  
--instance-tag-attribute IncludeAllTagsOfInstance=true
```

Salida:

```
{  
  "InstanceTagAttribute": {  
    "InstanceTagKeys": [],  
    "IncludeAllTagsOfInstance": true  
  }  
}
```

Para obtener más información, consulte [Eventos programados para sus instancias](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

## Ejemplo 2: Para eliminar etiquetas específicas de las notificaciones de eventos

El siguiente `deregister-instance-event-notification-attributes` ejemplo elimina la etiqueta especificada de las etiquetas incluidas en las notificaciones de eventos. Para describir el resto de etiquetas incluidas en las notificaciones de eventos, utilice `describe-instance-event-notification-attributes`.

```
aws ec2 deregister-instance-event-notification-attributes \  
  --instance-tag-attribute InstanceTagKeys="tag-key2"
```

Salida:

```
{  
  "InstanceTagAttribute": {  
    "InstanceTagKeys": [  
      "tag-key2"  
    ],  
    "IncludeAllTagsOfInstance": false  
  }  
}
```

Para obtener más información, consulte [Eventos programados para sus instancias](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [DeregisterInstanceEventNotificationAttributes](#) la Referencia de AWS CLI comandos.

## **deregister-transit-gateway-multicast-group-members**

En el siguiente ejemplo de código, se muestra cómo usar `deregister-transit-gateway-multicast-group-members`.

AWS CLI

Para anular el registro de los miembros de un grupo de multidifusión

En este ejemplo se anula el registro del miembro del grupo de interfaz de red especificado del grupo de multidifusión de Transit Gateway.

```
aws ec2 deregister-transit-gateway-multicast-group-members \  
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE \  
  --group-ip-address 224.0.1.0 \  
  --
```

```
--network-interface-ids eni-0e246d3269EXAMPLE
```

Salida:

```
{
  "DeregisteredMulticastGroupMembers": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef7EXAMPLE",
    "RegisteredNetworkInterfaceIds": [
      "eni-0e246d3269EXAMPLE"
    ],
    "GroupIpAddress": "224.0.1.0"
  }
}
```

Para obtener más información, consulte [Anular el registro de miembros de un grupo de multidifusión en la Guía del](#) usuario de Transit Gateways.AWS

- Para obtener API más información, consulte la Referencia de comandos [DeregisterTransitGatewayMulticastGroupMembers.AWS CLI](#)

## deregister-transit-gateway-multicast-group-source

En el siguiente ejemplo de código, se muestra cómo usar `deregister-transit-gateway-multicast-group-source`.

AWS CLI

Para anular el registro de una fuente del grupo de multidifusión de Transit Gateway

En este ejemplo se anula el registro de la fuente del grupo de interfaz de red especificada del grupo de multidifusión.

```
aws ec2 register-transit-gateway-multicast-group-sources \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
  --group-ip-address 224.0.1.0 \
  --network-interface-ids eni-07f290fc3c090cbae
```

Salida:

```
{
  "DeregisteredMulticastGroupSources": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
```

```
    "DeregisteredNetworkInterfaceIds": [  
        "eni-07f290fc3c090cbae"  
    ],  
    "GroupIpAddress": "224.0.1.0"  
  }  
}
```

Para obtener más información, consulte [Anular el registro de fuentes de un grupo de multidifusión en la Guía del](#) usuario de Transit Gateways.AWS

- Para obtener API más información, consulte la Referencia de comandos [DeregisterTransitGatewayMulticastGroupSource](#).AWS CLI

## describe-account-attributes

En el siguiente ejemplo de código, se muestra cómo usar `describe-account-attributes`.

### AWS CLI

Para describir todos los atributos de su AWS cuenta

En este ejemplo se describen los atributos de su AWS cuenta.

Comando:

```
aws ec2 describe-account-attributes
```

Salida:

```
{  
  "AccountAttributes": [  
    {  
      "AttributeName": "vpc-max-security-groups-per-interface",  
      "AttributeValues": [  
        {  
          "AttributeValue": "5"  
        }  
      ]  
    },  
    {  
      "AttributeName": "max-instances",  
      "AttributeValues": [  
        {
```

```
        "AttributeValue": "20"
      }
    ]
  },
  {
    "AttributeName": "supported-platforms",
    "AttributeValues": [
      {
        "AttributeValue": "EC2"
      },
      {
        "AttributeValue": "VPC"
      }
    ]
  },
  {
    "AttributeName": "default-vpc",
    "AttributeValues": [
      {
        "AttributeValue": "none"
      }
    ]
  },
  {
    "AttributeName": "max-elastic-ips",
    "AttributeValues": [
      {
        "AttributeValue": "5"
      }
    ]
  },
  {
    "AttributeName": "vpc-max-elastic-ips",
    "AttributeValues": [
      {
        "AttributeValue": "5"
      }
    ]
  }
]
}
```

Para describir un único atributo de su AWS cuenta

En este ejemplo se describe el `supported-platforms` atributo de tu AWS cuenta.

Comando:

```
aws ec2 describe-account-attributes --attribute-names supported-platforms
```

Salida:

```
{
  "AccountAttributes": [
    {
      "AttributeName": "supported-platforms",
      "AttributeValues": [
        {
          "AttributeValue": "EC2"
        },
        {
          "AttributeValue": "VPC"
        }
      ]
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeAccountAttributes](#) la Referencia de AWS CLI comandos.

## **describe-address-transfers**

En el siguiente ejemplo de código, se muestra cómo usar `describe-address-transfers`.

AWS CLI

Para describir una transferencia de direcciones IP elásticas

El siguiente `describe-address-transfers` ejemplo describe la transferencia de direcciones IP elásticas para la dirección IP elástica especificada.

```
aws ec2 describe-address-transfers \
  --allocation-ids eipalloc-09ad461b0d03f6aaf
```

Salida:

```
{
  "AddressTransfers": [
    {
      "PublicIp": "100.21.184.216",
      "AllocationId": "eipalloc-09ad461b0d03f6aaf",
      "TransferAccountId": "123456789012",
      "TransferOfferExpirationTimestamp": "2023-02-22T22:51:01.000Z",
      "AddressTransferStatus": "pending"
    }
  ]
}
```

Para obtener más información, consulta [Transferir direcciones IP elásticas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DescribeAddressTransfers](#) la Referencia de AWS CLI comandos.

## describe-addresses-attribute

En el siguiente ejemplo de código, se muestra cómo usar `describe-addresses-attribute`.

### AWS CLI

Para ver los atributos del nombre de dominio asociado a una dirección IP elástica

`describe-addresses-attribute` Los ejemplos siguientes devuelven los atributos del nombre de dominio asociado a la dirección IP elástica.

Linux:

```
aws ec2 describe-addresses-attribute \
  --allocation-ids eipalloc-abcdef01234567890 \
  --attribute domain-name
```

Windows:

```
aws ec2 describe-addresses-attribute ^
  --allocation-ids eipalloc-abcdef01234567890 ^
```



```
--attribute domain-name
```

Salida:

```
{
  "Addresses": [
    {
      "PublicIp": "192.0.2.0",
      "AllocationId": "eipalloc-abcdef01234567890",
      "PtrRecord": "example.com."
    }
  ]
}
```

Para ver los atributos de una dirección IP elástica, primero debe haber asociado un nombre de dominio a la dirección IP elástica. Para obtener más información, consulte [Uso inverso DNS para aplicaciones de correo electrónico](#) en la Guía del EC2 usuario de Amazon o [modify-address-attribute](#) en la Referencia de AWS CLI comandos.

- Para API obtener más información, consulte [DescribeAddressesAttribute](#) la Referencia de AWS CLI comandos.

## describe-addresses

En el siguiente ejemplo de código, se muestra cómo usar `describe-addresses`.

AWS CLI

Ejemplo 1: Recuperar detalles sobre todas las direcciones IP elásticas

En el siguiente ejemplo de `describe addresses`, se muestran los detalles de las direcciones IP elásticas.

```
aws ec2 describe-addresses
```

Salida:

```
{
  "Addresses": [
    {
```

```

    "InstanceId": "i-1234567890abcdef0",
    "PublicIp": "198.51.100.0",
    "PublicIpv4Pool": "amazon",
    "Domain": "standard"
  },
  {
    "Domain": "vpc",
    "PublicIpv4Pool": "amazon",
    "InstanceId": "i-1234567890abcdef0",
    "NetworkInterfaceId": "eni-12345678",
    "AssociationId": "eipassoc-12345678",
    "NetworkInterfaceOwnerId": "123456789012",
    "PublicIp": "203.0.113.0",
    "AllocationId": "eipalloc-12345678",
    "PrivateIpAddress": "10.0.1.241"
  }
]
}

```

Ejemplo 2: Para recuperar los detalles de sus direcciones IP elásticas para EC2 - VPC

En el siguiente describe-addresses ejemplo, se muestran detalles sobre las direcciones IP elásticas para utilizarlas con las instancias de unVPC.

```

aws ec2 describe-addresses \
  --filters "Name=domain,Values=vpc"

```

Salida:

```

{
  "Addresses": [
    {
      "Domain": "vpc",
      "PublicIpv4Pool": "amazon",
      "InstanceId": "i-1234567890abcdef0",
      "NetworkInterfaceId": "eni-12345678",
      "AssociationId": "eipassoc-12345678",
      "NetworkInterfaceOwnerId": "123456789012",
      "PublicIp": "203.0.113.0",
      "AllocationId": "eipalloc-12345678",
      "PrivateIpAddress": "10.0.1.241"
    }
  ]
}

```

```
]
}
```

Ejemplo 3: Recuperar detalles sobre una dirección IP elástica especificada por el ID de asignación

En el siguiente `describe-addresses` ejemplo, se muestran detalles sobre la dirección IP elástica con el ID de asignación especificado, que está asociado a una instancia en EC2 -VPC.

```
aws ec2 describe-addresses \
  --allocation-ids eipalloc-282d9641
```

Salida:

```
{
  "Addresses": [
    {
      "Domain": "vpc",
      "PublicIpv4Pool": "amazon",
      "InstanceId": "i-1234567890abcdef0",
      "NetworkInterfaceId": "eni-1a2b3c4d",
      "AssociationId": "eipassoc-123abc12",
      "NetworkInterfaceOwnerId": "1234567891012",
      "PublicIp": "203.0.113.25",
      "AllocationId": "eipalloc-282d9641",
      "PrivateIpAddress": "10.251.50.12"
    }
  ]
}
```

Ejemplo 4: Para recuperar detalles sobre una dirección IP elástica especificada por su dirección IP VPC privada

El siguiente `describe-addresses` ejemplo muestra detalles sobre la dirección IP elástica asociada a una dirección IP privada concreta en EC2 -VPC.

```
aws ec2 describe-addresses \
  --filters "Name=private-ip-address,Values=10.251.50.12"
```

Ejemplo 5: Para recuperar detalles sobre las direcciones IP elásticas en EC2 -Classic

El siguiente `describe-addresses` ejemplo muestra detalles sobre las direcciones IP elásticas para su uso en EC2 -Classic.

```
aws ec2 describe-addresses \  
  --filters "Name=domain,Values=standard"
```

Salida:

```
{  
  "Addresses": [  
    {  
      "InstanceId": "i-1234567890abcdef0",  
      "PublicIp": "203.0.110.25",  
      "PublicIpv4Pool": "amazon",  
      "Domain": "standard"  
    }  
  ]  
}
```

Ejemplo 6: Recuperar detalles sobre una dirección IP elástica especificada por la dirección IP pública

En el siguiente `describe-addresses` ejemplo, se muestran detalles sobre la dirección IP elástica con el valor `203.0.110.25`, que está asociado a una instancia en EC2 -Classic.

```
aws ec2 describe-addresses \  
  --public-ips 203.0.110.25
```

Salida:

```
{  
  "Addresses": [  
    {  
      "InstanceId": "i-1234567890abcdef0",  
      "PublicIp": "203.0.110.25",  
      "PublicIpv4Pool": "amazon",  
      "Domain": "standard"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [DescribeAddresses](#) la Referencia de AWS CLI comandos.

## **describe-aggregate-id-format**

En el siguiente ejemplo de código, se muestra cómo usar `describe-aggregate-id-format`.

### AWS CLI

Para describir la configuración del formato de ID más extenso para todos los tipos de recursos de una región

En el siguiente `describe-aggregate-id-format` ejemplo, se describe el estado general del formato de ID largo de la región actual. El `Deadline` valor indica que han expirado los plazos para que estos recursos cambien permanentemente del formato de ID corto al formato de ID largo. El `UseLongIdsAggregated` valor indica que todos los IAM usuarios y IAM roles están configurados para usar el formato de ID largo para todos los tipos de recursos.

```
aws ec2 describe-aggregate-id-format
```

Salida:

```
{
  "UseLongIdsAggregated": true,
  "Statuses": [
    {
      "Deadline": "2018-08-13T02:00:00.000Z",
      "Resource": "network-interface-attachment",
      "UseLongIds": true
    },
    {
      "Deadline": "2016-12-13T02:00:00.000Z",
      "Resource": "instance",
      "UseLongIds": true
    },
    {
      "Deadline": "2018-08-13T02:00:00.000Z",
      "Resource": "elastic-ip-association",
      "UseLongIds": true
    },
    ...
  ]
}
```

```
]
}
```

- Para API obtener más información, consulte [DescribeAggregatIdFormat](#) la Referencia de AWS CLI comandos.

## describe-availability-zones

En el siguiente ejemplo de código, se muestra cómo usar `describe-availability-zones`.

### AWS CLI

Describir las zonas de disponibilidad

En el siguiente ejemplo de `describe-availability-zones`, se muestran los detalles de las zonas de disponibilidad que están disponibles para usted. La respuesta incluye las zonas de disponibilidad solo para la región actual. En este ejemplo, se usa la región predeterminada del perfil `us-west-2` (Oregón).

```
aws ec2 describe-availability-zones
```

Salida:

```
{
  "AvailabilityZones": [
    {
      "State": "available",
      "OptInStatus": "opt-in-not-required",
      "Messages": [],
      "RegionName": "us-west-2",
      "ZoneName": "us-west-2a",
      "ZoneId": "usw2-az1",
      "GroupName": "us-west-2",
      "NetworkBorderGroup": "us-west-2"
    },
    {
      "State": "available",
      "OptInStatus": "opt-in-not-required",
      "Messages": [],
      "RegionName": "us-west-2",
      "ZoneName": "us-west-2b",
      "ZoneId": "usw2-az2",

```

```

    "GroupName": "us-west-2",
    "NetworkBorderGroup": "us-west-2"
  },
  {
    "State": "available",
    "OptInStatus": "opt-in-not-required",
    "Messages": [],
    "RegionName": "us-west-2",
    "ZoneName": "us-west-2c",
    "ZoneId": "usw2-az3",
    "GroupName": "us-west-2",
    "NetworkBorderGroup": "us-west-2"
  },
  {
    "State": "available",
    "OptInStatus": "opt-in-not-required",
    "Messages": [],
    "RegionName": "us-west-2",
    "ZoneName": "us-west-2d",
    "ZoneId": "usw2-az4",
    "GroupName": "us-west-2",
    "NetworkBorderGroup": "us-west-2"
  },
  {
    "State": "available",
    "OptInStatus": "opted-in",
    "Messages": [],
    "RegionName": "us-west-2",
    "ZoneName": "us-west-2-lax-1a",
    "ZoneId": "usw2-lax1-az1",
    "GroupName": "us-west-2-lax-1",
    "NetworkBorderGroup": "us-west-2-lax-1"
  }
]
}

```

- Para API obtener más información, consulte [DescribeAvailabilityZones](#) la Referencia de AWS CLI comandos.

## **describe-aws-network-performance-metric-subscription**

En el siguiente ejemplo de código, se muestra cómo usar `describe-aws-network-performance-metric-subscription`.

## AWS CLI

Para describir sus suscripciones a métricas

En el siguiente `describe-aws-network-performance-metric-subscriptions` ejemplo, se describen tus suscripciones a métricas.

```
aws ec2 describe-aws-network-performance-metric-subscriptions
```

Salida:

```
{
  "Subscriptions": [
    {
      "Source": "us-east-1",
      "Destination": "eu-west-1",
      "Metric": "aggregate-latency",
      "Statistic": "p50",
      "Period": "five-minutes"
    }
  ]
}
```

Para obtener más información, consulte [Administrar suscripciones](#) en la Guía del usuario de Infrastructure Performance.

- Para API obtener más información, consulte [DescribeAwsNetworkPerformanceMetricSubscription](#) la Referencia de AWS CLI comandos.

## `describe-aws-network-performance-metric-subscriptions`

En el siguiente ejemplo de código, se muestra cómo usar `describe-aws-network-performance-metric-subscriptions`.

## AWS CLI

Para describir sus suscripciones a métricas

En el siguiente `describe-aws-network-performance-metric-subscriptions` ejemplo, se describen tus suscripciones a métricas.



```
aws ec2 describe-aws-network-performance-metric-subscriptions
```

Salida:

```
{
  "Subscriptions": [
    {
      "Source": "us-east-1",
      "Destination": "eu-west-1",
      "Metric": "aggregate-latency",
      "Statistic": "p50",
      "Period": "five-minutes"
    }
  ]
}
```

Para obtener más información, consulte [Administrar suscripciones](#) en la Guía del usuario de Infrastructure Performance.

- Para API obtener más información, consulte [DescribeAwsNetworkPerformanceMetricSubscriptions](#) la Referencia de AWS CLI comandos.

## describe-bundle-tasks

En el siguiente ejemplo de código, se muestra cómo usar `describe-bundle-tasks`.

AWS CLI

Para describir las tareas del paquete

En este ejemplo se describen todas las tareas del paquete.

Comando:

```
aws ec2 describe-bundle-tasks
```

Salida:

```
{
  "BundleTasks": [
    {
```

```
"UpdateTime": "2015-09-15T13:26:54.000Z",
"InstanceId": "i-1234567890abcdef0",
"Storage": {
  "S3": {
    "Prefix": "winami",
    "Bucket": "bundletasks"
  }
},
"State": "bundling",
"StartTime": "2015-09-15T13:24:35.000Z",
"Progress": "3%",
"BundleId": "bun-2a4e041c"
}
]
}
```

- Para API obtener más información, consulte [DescribeBundleTasks](#) la Referencia de AWS CLI comandos.

## describe-byoip-cidrs

En el siguiente ejemplo de código, se muestra cómo usar `describe-byoip-cidrs`.

### AWS CLI

Para describir los rangos de direcciones aprovisionados

En el siguiente `describe-byoip-cidrs` ejemplo, se muestran detalles sobre los rangos de IPv4 direcciones públicas que ha aprovisionado para su uso. AWS

```
aws ec2 describe-byoip-cidrs
```

Salida:

```
{
  "ByoipCidrs": [
    {
      "Cidr": "203.0.113.25/24",
      "StatusMessage": "ipv4pool-ec2-1234567890abcdef0",
      "State": "provisioned"
    }
  ]
}
```

```
}
```

- Para API obtener más información, consulte [DescribeByoipCidrs](#) la Referencia de AWS CLI comandos.

## describe-capacity-reservation-fleets

En el siguiente ejemplo de código, se muestra cómo usar `describe-capacity-reservation-fleets`.

### AWS CLI

Para ver una flota de reserva de capacidad

En el siguiente `describe-capacity-reservation-fleets` ejemplo, se muestra la información de configuración y capacidad de la flota de reserva de capacidad especificada. También incluye detalles sobre las reservas de capacidad individuales que se encuentran dentro de la flota. :

```
aws ec2 describe-capacity-reservation-fleets \  
  --capacity-reservation-fleet-ids crf-abcdef01234567890
```

Salida:

```
{  
  "CapacityReservationFleets": [  
    {  
      "Status": "active",  
      "EndDate": "2022-12-31T23:59:59.000Z",  
      "InstanceMatchCriteria": "open",  
      "Tags": [],  
      "CapacityReservationFleetId": "crf-abcdef01234567890",  
      "Tenancy": "default",  
      "InstanceTypeSpecifications": [  
        {  
          "CapacityReservationId": "cr-1234567890abcdef0",  
          "AvailabilityZone": "us-east-1a",  
          "FulfilledCapacity": 5.0,  
          "Weight": 1.0,  
          "CreateDate": "2022-07-02T08:34:33.398Z",  
          "InstancePlatform": "Linux/UNIX",
```

```
        "TotalInstanceCount": 5,  
        "Priority": 1,  
        "EbsOptimized": true,  
        "InstanceType": "m5.xlarge"  
    }  
],  
"TotalTargetCapacity": 5,  
"TotalFulfilledCapacity": 5.0,  
"CreateTime": "2022-07-02T08:34:33.397Z",  
"AllocationStrategy": "prioritized"  
}  
]  
}
```

Para obtener más información sobre las flotas de reserva de capacidad, consulta [Flotas de reserva de capacidad](#) en la Guía EC2 del usuario de Amazon.

- Para API obtener más información, consulte la Referencia [DescribeCapacityReservationFleets](#) de AWS CLI comandos.

## describe-capacity-reservations

En el siguiente ejemplo de código, se muestra cómo usar `describe-capacity-reservations`.

### AWS CLI

Ejemplo 1: Para describir una o más de sus reservas de capacidad

En el siguiente `describe-capacity-reservations` ejemplo, se muestran detalles sobre todas sus reservas de capacidad en la AWS región actual.

```
aws ec2 describe-capacity-reservations
```

Salida:

```
{  
  "CapacityReservations": [  
    {  
      "CapacityReservationId": "cr-1234abcd56EXAMPLE ",  
      "EndDateType": "unlimited",  
      "AvailabilityZone": "eu-west-1a",  
      "InstanceMatchCriteria": "open",  
    }  
  ]  
}
```

```

    "Tags": [],
    "EphemeralStorage": false,
    "CreateDate": "2019-08-16T09:03:18.000Z",
    "AvailableInstanceCount": 1,
    "InstancePlatform": "Linux/UNIX",
    "TotalInstanceCount": 1,
    "State": "active",
    "Tenancy": "default",
    "EbsOptimized": true,
    "InstanceType": "a1.medium"
  },
  {
    "CapacityReservationId": "cr-abcdEXAMPLE9876ef ",
    "EndDateType": "unlimited",
    "AvailabilityZone": "eu-west-1a",
    "InstanceMatchCriteria": "open",
    "Tags": [],
    "EphemeralStorage": false,
    "CreateDate": "2019-08-07T11:34:19.000Z",
    "AvailableInstanceCount": 3,
    "InstancePlatform": "Linux/UNIX",
    "TotalInstanceCount": 3,
    "State": "cancelled",
    "Tenancy": "default",
    "EbsOptimized": true,
    "InstanceType": "m5.large"
  }
]
}

```

Ejemplo 2: Para describir una o más de sus reservas de capacidad

En el siguiente describe-capacity-reservations ejemplo, se muestran detalles sobre la reserva de capacidad especificada.

```

aws ec2 describe-capacity-reservations \
  --capacity-reservation-ids cr-1234abcd56EXAMPLE

```

Salida:

```

{
  "CapacityReservations": [
    {

```

```
    "CapacityReservationId": "cr-1234abcd56EXAMPLE",
    "EndDateType": "unlimited",
    "AvailabilityZone": "eu-west-1a",
    "InstanceMatchCriteria": "open",
    "Tags": [],
    "EphemeralStorage": false,
    "CreateDate": "2019-08-16T09:03:18.000Z",
    "AvailableInstanceCount": 1,
    "InstancePlatform": "Linux/UNIX",
    "TotalInstanceCount": 1,
    "State": "active",
    "Tenancy": "default",
    "EbsOptimized": true,
    "InstanceType": "a1.medium"
  }
]
```

Para obtener más información, consulte [Visualización de una reserva de capacidad](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [DescribeCapacityReservations](#) la Referencia de AWS CLI comandos.

## describe-carrier-gateways

En el siguiente ejemplo de código, se muestra cómo usar `describe-carrier-gateways`.

### AWS CLI

Para describir todas las pasarelas portadoras

En el siguiente `describe-carrier-gateways` ejemplo, se enumeran todas las pasarelas de sus operadores.

```
aws ec2 describe-carrier-gateways
```

Salida:

```
{
  "CarrierGateways": [
    {
```

```
    "CarrierGatewayId": "cagw-0465cdEXAMPLE1111",
    "VpcId": "vpc-0c529aEXAMPLE",
    "State": "available",
    "OwnerId": "123456789012",
    "Tags": [
      {
        "Key": "example",
        "Value": "tag"
      }
    ]
  }
]
```

Para obtener más información, consulte [Carrier gateways](https://docs.aws.amazon.com/vpc/latest/UserGuide/carrier_Gateway.html)< [https://docs.aws.amazon.com/vpc/latest/UserGuide/carrier\\_Gateway.html](https://docs.aws.amazon.com/vpc/latest/UserGuide/carrier_Gateway.html)> en la Guía del usuario de Amazon Virtual Private Cloud.

- Para obtener API más información, [DescribeCarrierGateways](#) consulte AWS CLI la Referencia de comandos.

## describe-classic-link-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-classic-link-instances`.

### AWS CLI

Para describir las instancias EC2 clásicas enlazadas

En este ejemplo, se muestran todas las instancias de EC2 -Classic vinculadas.

Comando:

```
aws ec2 describe-classic-link-instances
```

Salida:

```
{
  "Instances": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "VpcId": "vpc-88888888",
      "Groups": [
```

```

        {
            "GroupId": "sg-11122233"
        }
    ],
    "Tags": [
        {
            "Value": "ClassicInstance",
            "Key": "Name"
        }
    ]
},
{
    "InstanceId": "i-0598c7d356eba48d7",
    "VpcId": "vpc-12312312",
    "Groups": [
        {
            "GroupId": "sg-aabbccdd"
        }
    ],
    "Tags": [
        {
            "Value": "ClassicInstance2",
            "Key": "Name"
        }
    ]
}
]
}

```

En este ejemplo, se muestran todas las instancias de EC2 -Classic vinculadas y se filtra la respuesta para incluir solo las instancias que están vinculadas a VPC vpc-88888888.

Comando:

```
aws ec2 describe-classic-link-instances --filter "Name=vpc-id,Values=vpc-88888888"
```

Salida:

```

{
    "Instances": [
        {
            "InstanceId": "i-1234567890abcdef0",
            "VpcId": "vpc-88888888",

```



```

        "Groups": [
            {
                "GroupId": "sg-11122233"
            }
        ],
        "Tags": [
            {
                "Value": "ClassicInstance",
                "Key": "Name"
            }
        ]
    }
}

```

- Para API obtener más información, consulta la Referencia de [DescribeClassicLinkInstances](#) comandos AWS CLI .

## describe-client-vpn-authorization-rules

En el siguiente ejemplo de código, se muestra cómo usar `describe-client-vpn-authorization-rules`.

### AWS CLI

Para describir las reglas de autorización de un VPN punto final de cliente

En el siguiente `describe-client-vpn-authorization-rules` ejemplo, se muestran detalles sobre las reglas de autorización del VPN punto final del cliente especificado.

```

aws ec2 describe-client-vpn-authorization-rules \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde

```

Salida:

```

{
  "AuthorizationRules": [
    {
      "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
      "GroupId": "",
      "AccessAll": true,
      "DestinationCidr": "0.0.0.0/0",
    }
  ]
}

```

```

        "Status": {
            "Code": "active"
        }
    ]
}

```

Para obtener más información, consulte [las reglas de autorización](#) en la Guía del VPN administrador del AWS cliente.

- Para API obtener más información, consulte [DescribeClientVpnAuthorizationRules](#) la Referencia de AWS CLI comandos.

## describe-client-vpn-connections

En el siguiente ejemplo de código, se muestra cómo usar `describe-client-vpn-connections`.

### AWS CLI

Para describir las conexiones a un VPN punto final del cliente

El siguiente `describe-client-vpn-connections` ejemplo muestra detalles sobre las conexiones del cliente al VPN punto final del cliente especificado.

```

aws ec2 describe-client-vpn-connections \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde

```

Salida:

```

{
  "Connections": [
    {
      "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
      "Timestamp": "2019-08-12 07:58:34",
      "ConnectionId": "cvpn-connection-0e03eb24267165acd",
      "ConnectionEstablishedTime": "2019-08-12 07:57:14",
      "IngressBytes": "32302",
      "EgressBytes": "5696",
      "IngressPackets": "332",
      "EgressPackets": "67",
      "ClientIp": "172.31.0.225",
      "CommonName": "client1.domain.tld",
    }
  ]
}

```

```

        "Status": {
            "Code": "terminated"
        },
        "ConnectionEndTime": "2019-08-12 07:58:34"
    },
    {
        "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
        "Timestamp": "2019-08-12 08:02:54",
        "ConnectionId": "cvpn-connection-00668867a40f18253",
        "ConnectionEstablishedTime": "2019-08-12 08:02:53",
        "IngressBytes": "2951",
        "EgressBytes": "2611",
        "IngressPackets": "9",
        "EgressPackets": "6",
        "ClientIp": "172.31.0.226",
        "CommonName": "client1.domain.tld",
        "Status": {
            "Code": "active"
        },
        "ConnectionEndTime": "-"
    }
]
}

```

Para obtener más información, consulte [Conexiones de clientes](#) en la Guía VPN del administrador de AWS clientes.

- Para API obtener más información, consulte [DescribeClientVpnConnections](#) la Referencia de AWS CLI comandos.

## describe-client-vpn-endpoints

En el siguiente ejemplo de código, se muestra cómo usar `describe-client-vpn-endpoints`.

### AWS CLI

Para describir los puntos VPN finales de su cliente

En el siguiente `describe-client-vpn-endpoints` ejemplo, se muestran detalles sobre todos los puntos VPN finales de su cliente.

```
aws ec2 describe-client-vpn-endpoints
```

**Salida:**

```
{
  "ClientVpnEndpoints": [
    {
      "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
      "Description": "Endpoint for Admin access",
      "Status": {
        "Code": "available"
      },
      "CreationTime": "2020-11-13T11:37:27",
      "DnsName": "*.cvpn-endpoint-123456789123abcde.prod.clientvpn.ap-
south-1.amazonaws.com",
      "ClientCidrBlock": "172.31.0.0/16",
      "DnsServers": [
        "8.8.8.8"
      ],
      "SplitTunnel": false,
      "VpnProtocol": "openvpn",
      "TransportProtocol": "udp",
      "VpnPort": 443,
      "ServerCertificateArn": "arn:aws:acm:ap-
south-1:123456789012:certificate/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "AuthenticationOptions": [
        {
          "Type": "certificate-authentication",
          "MutualAuthentication": {
            "ClientRootCertificateChain": "arn:aws:acm:ap-
south-1:123456789012:certificate/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE"
          }
        }
      ],
      "ConnectionLogOptions": {
        "Enabled": true,
        "CloudwatchLogGroup": "Client-vpn-connection-logs",
        "CloudwatchLogStream": "cvpn-endpoint-123456789123abcde-ap-
south-1-2020/11/13-FCD8HEMVAccw"
      },
      "Tags": [
        {
          "Key": "Name",
          "Value": "Client VPN"
        }
      ],
    }
  ],
}
```

```

        "SecurityGroupIds": [
            "sg-aabbcc11223344567"
        ],
        "VpcId": "vpc-a87f92c1",
        "SelfServicePortalUrl": "https://self-service.clientvpn.amazonaws.com/
endpoints/cvpn-endpoint-123456789123abcde",
        "ClientConnectOptions": {
            "Enabled": false
        }
    }
]
}

```

Para obtener más información, consulte los [VPN puntos finales de los clientes](#) en la Guía del VPN administrador de AWS clientes.

- Para API obtener más información, consulte [DescribeClientVpnEndpoints](#) la Referencia de AWS CLI comandos.

## describe-client-vpn-routes

En el siguiente ejemplo de código, se muestra cómo usar `describe-client-vpn-routes`.

### AWS CLI

Para describir las rutas de un VPN punto final del cliente

En el siguiente `describe-client-vpn-routes` ejemplo, se muestran detalles sobre las rutas del VPN punto final del cliente especificado.

```

aws ec2 describe-client-vpn-routes \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde

```

Salida:

```

{
  "Routes": [
    {
      "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
      "DestinationCidr": "10.0.0.0/16",
      "TargetSubnet": "subnet-0123456789abcabca",
      "Type": "Nat",
    }
  ]
}

```

```

    "Origin": "associate",
    "Status": {
      "Code": "active"
    },
    "Description": "Default Route"
  },
  {
    "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
    "DestinationCidr": "0.0.0.0/0",
    "TargetSubnet": "subnet-0123456789abcabca",
    "Type": "Nat",
    "Origin": "add-route",
    "Status": {
      "Code": "active"
    }
  }
]
}

```

Para obtener más información, consulte [Rutas](#) en la Guía del VPN administrador del AWS cliente.

- Para API obtener más información, consulte [DescribeClientVpnRoutes](#) la Referencia de AWS CLI comandos.

## describe-client-vpn-target-networks

En el siguiente ejemplo de código, se muestra cómo usar `describe-client-vpn-target-networks`.

### AWS CLI

Para describir las redes de destino de un VPN punto final de cliente

El siguiente `describe-client-vpn-target-networks` ejemplo muestra detalles sobre las redes de destino del VPN punto final del cliente especificado.

```

aws ec2 describe-client-vpn-target-networks \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde

```

Salida:

```
{
```

```

"ClientVpnTargetNetworks": [
  {
    "AssociationId": "cvpn-assoc-012e837060753dc3d",
    "VpcId": "vpc-11111222222333333",
    "TargetNetworkId": "subnet-0123456789abcabca",
    "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
    "Status": {
      "Code": "associating"
    },
    "SecurityGroups": [
      "sg-012345678910abcab"
    ]
  }
]
}

```

Para obtener más información, consulte [Target Networks](#) en la Guía del VPN administrador de AWS clientes.

- Para API obtener más información, consulte [DescribeClientVpnTargetNetworks](#) la Referencia de AWS CLI comandos.

## describe-coip-pools

En el siguiente ejemplo de código, se muestra cómo usar `describe-coip-pools`.

### AWS CLI

Para describir los grupos de direcciones IP propiedad del cliente

En el siguiente `describe-coip-pools` ejemplo, se describen los grupos de direcciones IP propiedad del cliente en su cuenta. AWS

```
aws ec2 describe-coip-pools
```

Salida:

```

{
  "CoipPools": [
    {
      "PoolId": "ipv4pool-coip-123a45678bEXAMPLE",
      "PoolCidrs": [

```

```
        "0.0.0.0/0"
      ],
      "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE",
      "PoolArn": "arn:aws:ec2:us-west-2:123456789012:coip-pool/ipv4pool-
coip-123a45678bEXAMPLE"
    }
  ]
}
```

Para obtener más información, consulte [Direcciones IP propiedad del cliente](#) en la Guía del usuario de AWS .

- Para API obtener más información, consulte la Referencia [DescribeCoipPools](#) de AWS CLI comandos.

## describe-conversion-tasks

En el siguiente ejemplo de código, se muestra cómo usar `describe-conversion-tasks`.

### AWS CLI

Para ver el estado de una tarea de conversión

En este ejemplo, se devuelve el estado de una tarea de conversión con el identificador `import-i-ffvko9js`.

Comando:

```
aws ec2 describe-conversion-tasks --conversion-task-ids import-i-ffvko9js
```

Salida:

```
{
  "ConversionTasks": [
    {
      "ConversionTaskId": "import-i-ffvko9js",
      "ImportInstance": {
        "InstanceId": "i-1234567890abcdef0",
        "Volumes": [
          {
            "Volume": {
              "Id": "vol-049df61146c4d7901",
```



```

        "Size": 16
      },
      "Status": "completed",
      "Image": {
        "Size": 1300687360,
        "ImportManifestUrl": "https://s3.amazonaws.com/
myimportbucket/411443cd-d620-4f1c-9d66-13144EXAMPLE/RHEL5.vmdkmanifest.xml?
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Expires=140EXAMPLE&Signature=XYNhznHNgCqsjDxL9wRL
%2FJvEXAMPLE",
        "Format": "VMDK"
      },
      "BytesConverted": 1300682960,
      "AvailabilityZone": "us-east-1d"
    }
  ]
},
"ExpirationTime": "2014-05-14T22:06:23Z",
"State": "completed"
}
]
}

```

- Para API obtener más información, consulte [DescribeConversionTasks](#) la Referencia de AWS CLI comandos.

## describe-customer-gateways

En el siguiente ejemplo de código, se muestra cómo usar `describe-customer-gateways`.

### AWS CLI

Para describir las pasarelas de sus clientes

En este ejemplo se describen las pasarelas de sus clientes.

Comando:

```
aws ec2 describe-customer-gateways
```

Salida:

```
{
```

```
"CustomerGateways": [  
  {  
    "CustomerGatewayId": "cgw-b4dc3961",  
    "IpAddress": "203.0.113.12",  
    "State": "available",  
    "Type": "ipsec.1",  
    "BgpAsn": "65000"  
  },  
  {  
    "CustomerGatewayId": "cgw-0e11f167",  
    "IpAddress": "12.1.2.3",  
    "State": "available",  
    "Type": "ipsec.1",  
    "BgpAsn": "65534"  
  }  
]  
}
```

Para describir una pasarela de clientes específica

En este ejemplo se describe la pasarela de clientes especificada.

Comando:

```
aws ec2 describe-customer-gateways --customer-gateway-ids cgw-0e11f167
```

Salida:

```
{  
  "CustomerGateways": [  
    {  
      "CustomerGatewayId": "cgw-0e11f167",  
      "IpAddress": "12.1.2.3",  
      "State": "available",  
      "Type": "ipsec.1",  
      "BgpAsn": "65534"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [DescribeCustomerGateways](#) la Referencia de AWS CLI comandos.

## describe-dhcp-options

En el siguiente ejemplo de código, se muestra cómo usar `describe-dhcp-options`.

### AWS CLI

Ejemplo 1: Para describir sus DHCP opciones

En el siguiente `describe-dhcp-options` ejemplo, se recuperan los detalles sobre sus DHCP opciones.

```
aws ec2 describe-dhcp-options
```

Salida:

```
{
  "DhcpOptions": [
    {
      "DhcpConfigurations": [
        {
          "Key": "domain-name",
          "Values": [
            {
              "Value": "us-east-2.compute.internal"
            }
          ]
        },
        {
          "Key": "domain-name-servers",
          "Values": [
            {
              "Value": "AmazonProvidedDNS"
            }
          ]
        }
      ],
      "DhcpOptionsId": "dopt-19edf471",
      "OwnerId": "111122223333"
    },
    {
      "DhcpConfigurations": [
        {
          "Key": "domain-name",
```

```

        "Values": [
            {
                "Value": "us-east-2.compute.internal"
            }
        ],
    },
    {
        "Key": "domain-name-servers",
        "Values": [
            {
                "Value": "AmazonProvidedDNS"
            }
        ]
    }
],
"DhcpOptionsId": "dopt-fEXAMPLE",
"OwnerId": "111122223333"
}
]
}

```

Para obtener más información, consulte [Trabajo con conjuntos de DHCP opciones](#) en la Guía del AWS VPC usuario.

Ejemplo 2: Para describir DHCP las opciones y filtrar el resultado

En el siguiente `describe-dhcp-options` ejemplo se describen DHCP las opciones y se utiliza un filtro para mostrar solo DHCP las opciones disponibles `example.com` para el servidor de nombres de dominio. El ejemplo usa el `--query` parámetro para mostrar solo la información de configuración y el ID en la salida.

```

aws ec2 describe-dhcp-options \
  --filters Name=key,Values=domain-name-servers Name=value,Values=example.com \
  --query "DhcpOptions[*].[DhcpConfigurations,DhcpOptionsId]"

```

Salida:

```

[
  [
    [
      {
        "Key": "domain-name",

```

```
    "Values": [  
      {  
        "Value": "example.com"  
      }  
    ],  
    {  
      "Key": "domain-name-servers",  
      "Values": [  
        {  
          "Value": "172.16.16.16"  
        }  
      ]  
    }  
  ],  
  "dopt-001122334455667ab"  
]
```

Para obtener más información, consulte [Trabajo con conjuntos de DHCP opciones](#) en la Guía del AWS VPC usuario.

- Para API obtener más información, consulte [DescribeDhcpOptions](#) la Referencia de AWS CLI comandos.

## **describe-egress-only-internet-gateways**

En el siguiente ejemplo de código, se muestra cómo usar `describe-egress-only-internet-gateways`.

AWS CLI

Para describir las pasarelas de Internet de solo salida

En este ejemplo se describen las pasarelas de Internet de solo salida.

Comando:

```
aws ec2 describe-egress-only-internet-gateways
```

Salida:

```
{
  "EgressOnlyInternetGateways": [
    {
      "EgressOnlyInternetGatewayId": "eigw-015e0e244e24dfe8a",
      "Attachments": [
        {
          "State": "attached",
          "VpcId": "vpc-0c62a468"
        }
      ]
    }
  ]
}
```

- Para obtener API más información, consulte la Referencia de comandos [DescribeEgressOnlyInternetGateways](#).AWS CLI

## describe-elastic-gpus

En el siguiente ejemplo de código, se muestra cómo usar `describe-elastic-gpus`.

AWS CLI

Para describir un Elastic GPU

Comando:

```
aws ec2 describe-elastic-gpus --elastic-gpu-ids egpu-12345678901234567890abcdefghijkl
```

- Para API obtener más información, consulte [DescribeElasticGpus](#) la Referencia de AWS CLI comandos.

## describe-export-image-tasks

En el siguiente ejemplo de código, se muestra cómo usar `describe-export-image-tasks`.

AWS CLI

Para supervisar una tarea de exportación de imágenes

El siguiente `describe-export-image-tasks` ejemplo comprueba el estado de la tarea de exportación de imágenes especificada. El archivo de imagen resultante en Amazon S3 es `my-export-bucket/exports/export-ami-1234567890abcdef0.vmdk`.

```
aws ec2 describe-export-image-tasks \  
  --export-image-task-ids export-ami-1234567890abcdef0
```

Resultado de una tarea de exportación de imágenes en curso.

```
{  
  "ExportImageTasks": [  
    {  
      "ExportImageTaskId": "export-ami-1234567890abcdef0"  
      "Progress": "21",  
      "S3ExportLocation": {  
        "S3Bucket": "my-export-bucket",  
        "S3Prefix": "exports/"  
      },  
      "Status": "active",  
      "StatusMessage": "updating"  
    }  
  ]  
}
```

Resultado de una tarea de exportación de imágenes que se ha completado.

```
{  
  "ExportImageTasks": [  
    {  
      "ExportImageTaskId": "export-ami-1234567890abcdef0"  
      "S3ExportLocation": {  
        "S3Bucket": "my-export-bucket",  
        "S3Prefix": "exports/"  
      },  
      "Status": "completed"  
    }  
  ]  
}
```

Para obtener más información, consulte [Exportar una máquina virtual desde una AMI](#) en la Guía del usuario de VM Import/Export.

- Para API obtener más información, consulte la Referencia [DescribeExportImageTasks](#) de AWS CLI comandos.

## describe-export-tasks

En el siguiente ejemplo de código, se muestra cómo usar `describe-export-tasks`.

### AWS CLI

Para enumerar los detalles sobre una tarea de exportación de instancias

En este ejemplo, se describe la tarea de exportación con el ID `export-i-fh8sjjsq`.

Comando:

```
aws ec2 describe-export-tasks --export-task-ids export-i-fh8sjjsq
```

Salida:

```
{
  "ExportTasks": [
    {
      "State": "active",
      "InstanceExportDetails": {
        "InstanceId": "i-1234567890abcdef0",
        "TargetEnvironment": "vmware"
      },
      "ExportToS3Task": {
        "S3Bucket": "myexportbucket",
        "S3Key": "RHEL5export-i-fh8sjjsq.ova",
        "DiskImageFormat": "vmdk",
        "ContainerFormat": "ova"
      },
      "Description": "RHEL5 instance",
      "ExportTaskId": "export-i-fh8sjjsq"
    }
  ]
}
```

- Para API obtener más información, consulte la Referencia de [DescribeExportTasks](#) comandos AWS CLI .



## describe-fast-launch-images

En el siguiente ejemplo de código, se muestra cómo usar `describe-fast-launch-images`.

### AWS CLI

Para describir los detalles de Windows AMIs que están configurados para un inicio más rápido

El siguiente `describe-fast-launch-images` ejemplo describe los detalles de cada uno de los AMIs componentes de su cuenta que están configurados para un inicio más rápido, incluidos el tipo de recurso, la configuración de las instantáneas, los detalles de la plantilla de lanzamiento, el número máximo de lanzamientos paralelos, el ID de AMI propietario, el estado de la configuración de inicio rápido, el motivo por el que se cambió el estado y la hora en que se produjo el cambio de estado.

```
aws ec2 describe-fast-launch-images
```

Salida:

```
{
  "FastLaunchImages": [
    {
      "ImageId": "ami-01234567890abcdef",
      "ResourceType": "snapshot",
      "SnapshotConfiguration": {},
      "LaunchTemplate": {
        "LaunchTemplateId": "lt-01234567890abcdef",
        "LaunchTemplateName": "EC2FastLaunchDefaultResourceCreation-
a8c6215d-94e6-441b-9272-dbd1f87b07e2",
        "Version": "1"
      },
      "MaxParallelLaunches": 6,
      "OwnerId": "0123456789123",
      "State": "enabled",
      "StateTransitionReason": "Client.UserInitiated",
      "StateTransitionTime": "2022-01-27T22:20:06.552000+00:00"
    }
  ]
}
```

Para obtener más información sobre cómo configurar un Windows AMI para un inicio más rápido, consulte [Configurar su sistema AMI para un inicio más rápido](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeFastLaunchImages](#) la Referencia de AWS CLI comandos.

## describe-fast-snapshot-restores

En el siguiente ejemplo de código, se muestra cómo usar `describe-fast-snapshot-restores`.

### AWS CLI

Para describir las restauraciones rápidas de instantáneas

El siguiente `describe-fast-snapshot-restores` ejemplo muestra los detalles de todas las restauraciones rápidas de instantáneas con un estado `disabled`.

```
aws ec2 describe-fast-snapshot-restores \  
  --filters Name=state,Values=disabled
```

Salida:

```
{  
  "FastSnapshotRestores": [  
    {  
      "SnapshotId": "snap-1234567890abcdef0",  
      "AvailabilityZone": "us-west-2c",  
      "State": "disabled",  
      "StateTransitionReason": "Client.UserInitiated - Lifecycle state  
transition",  
      "OwnerId": "123456789012",  
      "EnablingTime": "2020-01-25T23:57:49.596Z",  
      "OptimizingTime": "2020-01-25T23:58:25.573Z",  
      "EnabledTime": "2020-01-25T23:59:29.852Z",  
      "DisablingTime": "2020-01-26T00:40:56.069Z",  
      "DisabledTime": "2020-01-26T00:41:27.390Z"  
    }  
  ]  
}
```

El siguiente `describe-fast-snapshot-restores` ejemplo describe todas las restauraciones rápidas de instantáneas.

```
aws ec2 describe-fast-snapshot-restores
```

- Para API obtener más información, consulte [DescribeFastSnapshotRestores](#) la Referencia de AWS CLI comandos.

## `describe-fleet-history`

En el siguiente ejemplo de código, se muestra cómo usar `describe-fleet-history`.

### AWS CLI

Para describir la historia EC2 de la flota

En el siguiente `describe-fleet-history` ejemplo, se devuelve el historial de la EC2 flota especificada a partir de la hora especificada. El resultado es para una EC2 flota con dos instancias en ejecución.

```
aws ec2 describe-fleet-history \  
  --fleet-id fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE \  
  --start-time 2020-09-01T00:00:00Z
```

Salida:

```
{  
  "HistoryRecords": [  
    {  
      "EventInformation": {  
        "EventSubType": "submitted"  
      },  
      "EventType": "fleetRequestChange",  
      "Timestamp": "2020-09-01T18:26:05.000Z"  
    },  
    {  
      "EventInformation": {  
        "EventSubType": "active"  
      },  
      "EventType": "fleetRequestChange",
```

```

    "Timestamp": "2020-09-01T18:26:15.000Z"
  },
  {
    "EventInformation": {
      "EventDescription": "t2.small, ami-07c8bc5c1ce9598c3, ...",
      "EventSubType": "progress"
    },
    "EventType": "fleetRequestChange",
    "Timestamp": "2020-09-01T18:26:17.000Z"
  },
  {
    "EventInformation": {
      "EventDescription": "{\"instanceType\": \"t2.small\", ...}",
      "EventSubType": "launched",
      "InstanceId": "i-083a1c446e66085d2"
    },
    "EventType": "instanceChange",
    "Timestamp": "2020-09-01T18:26:17.000Z"
  },
  {
    "EventInformation": {
      "EventDescription": "{\"instanceType\": \"t2.small\", ...}",
      "EventSubType": "launched",
      "InstanceId": "i-090db02406cc3c2d6"
    },
    "EventType": "instanceChange",
    "Timestamp": "2020-09-01T18:26:17.000Z"
  }
],
"LastEvaluatedTime": "2020-09-01T19:10:19.000Z",
"FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
"StartTime": "2020-08-31T23:53:20.000Z"
}

```

Para obtener más información, consulte [Administración de una EC2 flota](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [DescribeFleetHistory](#) la Referencia de AWS CLI comandos.

## describe-fleet-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-fleet-instances`.

## AWS CLI

Para describir las instancias en ejecución de una EC2 flota

En el siguiente `describe-fleet-instances` ejemplo, se describen las instancias en ejecución de la EC2 flota especificada.

```
aws ec2 describe-fleet-instances \  
  --fleet-id 12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE
```

Salida:

```
{  
  "ActiveInstances": [  
    {  
      "InstanceId": "i-090db02406cc3c2d6",  
      "InstanceType": "t2.small",  
      "SpotInstanceRequestId": "sir-a43gtpfk",  
      "InstanceHealth": "healthy"  
    },  
    {  
      "InstanceId": "i-083a1c446e66085d2",  
      "InstanceType": "t2.small",  
      "SpotInstanceRequestId": "sir-iwcit2nj",  
      "InstanceHealth": "healthy"  
    }  
  ],  
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"  
}
```

Para obtener más información, consulte [Administración de una EC2 flota](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [DescribeFleetInstances](#) la Referencia de AWS CLI comandos.

## **describe-fleets**

En el siguiente ejemplo de código, se muestra cómo usar `describe-fleets`.

## AWS CLI

Para describir una EC2 flota

El siguiente describe-fleets ejemplo describe la EC2 flota especificada.

```
aws ec2 describe-fleets \  
  --fleet-ids fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE
```

Salida:

```
{  
  "Fleets": [  
    {  
      "ActivityStatus": "pending_fulfillment",  
      "CreateTime": "2020-09-01T18:26:05.000Z",  
      "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",  
      "FleetState": "active",  
      "ExcessCapacityTerminationPolicy": "termination",  
      "FulfilledCapacity": 0.0,  
      "FulfilledOnDemandCapacity": 0.0,  
      "LaunchTemplateConfigs": [  
        {  
          "LaunchTemplateSpecification": {  
            "LaunchTemplateId": "lt-0e632f2855a979cd5",  
            "Version": "1"  
          }  
        }  
      ],  
      "TargetCapacitySpecification": {  
        "TotalTargetCapacity": 2,  
        "OnDemandTargetCapacity": 0,  
        "SpotTargetCapacity": 2,  
        "DefaultTargetCapacityType": "spot"  
      },  
      "TerminateInstancesWithExpiration": false,  
      "Type": "maintain",  
      "ReplaceUnhealthyInstances": false,  
      "SpotOptions": {  
        "AllocationStrategy": "lowestPrice",  
        "InstanceInterruptionBehavior": "terminate",  
        "InstancePoolsToUseCount": 1  
      }  
    }  
  ],  
}
```

```
        "OnDemandOptions": {
            "AllocationStrategy": "lowestPrice"
        }
    ]
}
```

Para obtener más información, consulte [Administración de una EC2 flota](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [DescribeFleets](#) la Referencia de AWS CLI comandos.

## describe-flow-logs

En el siguiente ejemplo de código, se muestra cómo usar `describe-flow-logs`.

### AWS CLI

Ejemplo 1: Para describir todos los registros de flujo

En el siguiente `describe-flow-logs` ejemplo, se muestran los detalles de todos los registros de flujo.

```
aws ec2 describe-flow-logs
```

Salida:

```
{
  "FlowLogs": [
    {
      "CreationTime": "2018-02-21T13:22:12.644Z",
      "DeliverLogsPermissionArn": "arn:aws:iam::123456789012:role/flow-logs-
role",
      "DeliverLogsStatus": "SUCCESS",
      "FlowLogId": "fl-aabbccdd112233445",
      "MaxAggregationInterval": 600,
      "FlowLogStatus": "ACTIVE",
      "LogGroupName": "FlowLogGroup",
      "ResourceId": "subnet-12345678901234567",
      "TrafficType": "ALL",
    }
  ]
}
```

```

        "LogDestinationType": "cloud-watch-logs",
        "LogFormat": "${version} ${account-id} ${interface-id} ${srcaddr}
${dstaddr} ${srcport} ${dstport} ${protocol} ${packets} ${bytes} ${start} ${end}
${action} ${log-status}"
    },
    {
        "CreationTime": "2020-02-04T15:22:29.986Z",
        "DeliverLogsStatus": "SUCCESS",
        "FlowLogId": "fl-01234567890123456",
        "MaxAggregationInterval": 60,
        "FlowLogStatus": "ACTIVE",
        "ResourceId": "vpc-00112233445566778",
        "TrafficType": "ACCEPT",
        "LogDestinationType": "s3",
        "LogDestination": "arn:aws:s3:::my-flow-log-bucket/custom",
        "LogFormat": "${version} ${vpc-id} ${subnet-id} ${instance-id}
${interface-id} ${account-id} ${type} ${srcaddr} ${dstaddr} ${srcport} ${dstport}
${pkt-srcaddr} ${pkt-dstaddr} ${protocol} ${bytes} ${packets} ${start} ${end}
${action} ${tcp-flags} ${log-status}"
    }
]
}

```

Ejemplo 2: Para describir un subconjunto de sus registros de flujo

En el siguiente `describe-flow-logs` ejemplo, se utiliza un filtro para mostrar los detalles únicamente de los registros de flujo que se encuentran en el grupo de CloudWatch registros especificado en Amazon Logs.

```

aws ec2 describe-flow-logs \
  --filter "Name=log-group-name,Values=MyFlowLogs"

```

- Para API obtener más información, consulte [DescribeFlowLogs](#) la Referencia de AWS CLI comandos.

## describe-fpga-image-attribute

En el siguiente ejemplo de código, se muestra cómo usar `describe-fpga-image-attribute`.

### AWS CLI

Para describir los atributos de una FPGA imagen de Amazon



En este ejemplo se describen los permisos de carga de la información especificada AFI.

Comando:

```
aws ec2 describe-fpga-image-attribute --fpga-image-id afi-0d123e123bfc85abc --  
attribute LoadPermission
```

Salida:

```
{  
  "FpgaImageAttribute": {  
    "FpgaImageId": "afi-0d123e123bfc85abc",  
    "LoadPermissions": [  
      {  
        "UserId": "123456789012"  
      }  
    ]  
  }  
}
```

- Para API obtener más información, consulte [DescribeFpgaImageAttribute](#) la Referencia de AWS CLI comandos.

## describe-fpga-images

En el siguiente ejemplo de código, se muestra cómo usar describe-fpga-images.

AWS CLI

Para describir las FPGA imágenes de Amazon

En este ejemplo se describen las AFIs que son propiedad de una cuenta123456789012.

Comando:

```
aws ec2 describe-fpga-images --filters Name=owner-id,Values=123456789012
```

Salida:

```
{
```

```
"FpgaImages": [  
  {  
    "UpdateTime": "2017-12-22T12:09:14.000Z",  
    "Name": "my-afi",  
    "PciId": {  
      "SubsystemVendorId": "0xfedd",  
      "VendorId": "0x1d0f",  
      "DeviceId": "0xf000",  
      "SubsystemId": "0x1d51"  
    },  
    "FpgaImageGlobalId": "agfi-123cb27b5e84a0abc",  
    "Public": false,  
    "State": {  
      "Code": "available"  
    },  
    "ShellVersion": "0x071417d3",  
    "OwnerId": "123456789012",  
    "FpgaImageId": "afi-0d123e123bfc85abc",  
    "CreateTime": "2017-12-22T11:43:33.000Z",  
    "Description": "my-afi"  
  }  
]
```

- Para API obtener más información, consulte [DescribeFpgaImages](#) la Referencia de AWS CLI comandos.

## describe-host-reservation-offerings

En el siguiente ejemplo de código, se muestra cómo usar `describe-host-reservation-offerings`.

### AWS CLI

Para describir las ofertas de reservas para anfitriones dedicados

En este ejemplo, se describen las reservas de hosts dedicados para la familia de instancias M4 que están disponibles para su compra.

Comando:

```
aws ec2 describe-host-reservation-offerings --filter Name=instance-family,Values=m4
```

## Salida:

```
{
  "OfferingSet": [
    {
      "HourlyPrice": "1.499",
      "OfferingId": "hro-03f707bf363b6b324",
      "InstanceFamily": "m4",
      "PaymentOption": "NoUpfront",
      "UpfrontPrice": "0.000",
      "Duration": 31536000
    },
    {
      "HourlyPrice": "1.045",
      "OfferingId": "hro-0ef9181cabdef7a02",
      "InstanceFamily": "m4",
      "PaymentOption": "NoUpfront",
      "UpfrontPrice": "0.000",
      "Duration": 94608000
    },
    {
      "HourlyPrice": "0.714",
      "OfferingId": "hro-04567a15500b92a51",
      "InstanceFamily": "m4",
      "PaymentOption": "PartialUpfront",
      "UpfrontPrice": "6254.000",
      "Duration": 31536000
    },
    {
      "HourlyPrice": "0.484",
      "OfferingId": "hro-0d5d7a9d23ed7fbfe",
      "InstanceFamily": "m4",
      "PaymentOption": "PartialUpfront",
      "UpfrontPrice": "12720.000",
      "Duration": 94608000
    },
    {
      "HourlyPrice": "0.000",
      "OfferingId": "hro-05da4108ca998c2e5",
      "InstanceFamily": "m4",
      "PaymentOption": "AllUpfront",
      "UpfrontPrice": "23913.000",
      "Duration": 94608000
    }
  ]
}
```

```
{
  "HourlyPrice": "0.000",
  "OfferingId": "hro-0a9f9be3b95a3dc8f",
  "InstanceFamily": "m4",
  "PaymentOption": "AllUpfront",
  "UpfrontPrice": "12257.000",
  "Duration": 31536000
}
]
```

- Para API obtener más información, consulte [DescribeHostReservationOfferings](#) la Referencia de AWS CLI comandos.

## describe-host-reservations

En el siguiente ejemplo de código, se muestra cómo usar `describe-host-reservations`.

### AWS CLI

Para describir las reservas para anfitriones dedicados en tu cuenta

En este ejemplo, se describen las reservas para anfitriones dedicados de tu cuenta.

Comando:

```
aws ec2 describe-host-reservations
```

Salida:

```
{
  "HostReservationSet": [
    {
      "Count": 1,
      "End": "2019-01-10T12:14:09Z",
      "HourlyPrice": "1.499",
      "InstanceFamily": "m4",
      "OfferingId": "hro-03f707bf363b6b324",
      "PaymentOption": "NoUpfront",
      "State": "active",
      "HostIdSet": [
        "h-013abcd2a00cbd123"
      ]
    }
  ]
}
```

```
    ],
    "Start": "2018-01-10T12:14:09Z",
    "HostReservationId": "hr-0d418a3a4ffc669ae",
    "UpfrontPrice": "0.000",
    "Duration": 31536000
  }
]
}
```

- Para API obtener más información, consulte [DescribeHostReservations](#) la Referencia de AWS CLI comandos.

## describe-hosts

En el siguiente ejemplo de código, se muestra cómo usar describe-hosts.

### AWS CLI

Para ver detalles sobre los hosts dedicados

En el siguiente describe-hosts ejemplo, se muestran los detalles de los hosts available dedicados de su AWS cuenta.

```
aws ec2 describe-hosts --filter "Name=state,Values=available"
```

Salida:

```
{
  "Hosts": [
    {
      "HostId": "h-07879acf49EXAMPLE",
      "Tags": [
        {
          "Value": "production",
          "Key": "purpose"
        }
      ],
      "HostProperties": {
        "Cores": 48,
        "TotalVCpus": 96,
        "InstanceType": "m5.large",
```

```
        "Sockets": 2
      },
      "Instances": [],
      "State": "available",
      "AvailabilityZone": "eu-west-1a",
      "AvailableCapacity": {
        "AvailableInstanceCapacity": [
          {
            "AvailableCapacity": 48,
            "InstanceType": "m5.large",
            "TotalCapacity": 48
          }
        ],
        "AvailableVCpus": 96
      },
      "HostRecovery": "on",
      "AllocationTime": "2019-08-19T08:57:44.000Z",
      "AutoPlacement": "off"
    }
  ]
}
```

Para obtener más información, consulte [Visualización de hosts dedicados](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [DescribeHosts](#) la Referencia de AWS CLI comandos.

## **describe-iam-instance-profile-associations**

En el siguiente ejemplo de código, se muestra cómo usar `describe-iam-instance-profile-associations`.

### AWS CLI

Para describir las asociaciones de perfiles de IAM instancia

En este ejemplo, se describen todas las asociaciones de perfiles de IAM instancia.

Comando:

```
aws ec2 describe-iam-instance-profile-associations
```

**Salida:**

```
{
  "IamInstanceProfileAssociations": [
    {
      "InstanceId": "i-09eb09efa73ec1dee",
      "State": "associated",
      "AssociationId": "iip-assoc-0db249b1f25fa24b8",
      "IamInstanceProfile": {
        "Id": "AIPAJVQN4F5WVLGCJDRGM",
        "Arn": "arn:aws:iam::123456789012:instance-profile/admin-role"
      }
    },
    {
      "InstanceId": "i-0402909a2f4dfffd14",
      "State": "associating",
      "AssociationId": "iip-assoc-0d1ec06278d29f44a",
      "IamInstanceProfile": {
        "Id": "AGJAJVQN4F5WVLGCJABCM",
        "Arn": "arn:aws:iam::123456789012:instance-profile/user1-role"
      }
    }
  ]
}
```

- Para API obtener más información, consulte [DescribelamInstanceProfileAssociations](#) la Referencia de AWS CLI comandos.

**describe-id-format**

En el siguiente ejemplo de código, se muestra cómo usar `describe-id-format`.

**AWS CLI**

Ejemplo 1: Para describir el formato de ID de un recurso

El siguiente `describe-id-format` ejemplo describe el formato de ID de los grupos de seguridad.

```
aws ec2 describe-id-format \
  --resource security-group
```

En el siguiente resultado de ejemplo, el `Deadline` valor indica que la fecha límite para que este tipo de recurso cambiara permanentemente del formato de ID corto al formato de ID largo expiró a las 00:00 del 15 de UTC agosto de 2018.

```
{
  "Statuses": [
    {
      "Deadline": "2018-08-15T00:00:00.000Z",
      "Resource": "security-group",
      "UseLongIds": true
    }
  ]
}
```

Ejemplo 2: Para describir el formato de ID de todos los recursos

En el siguiente `describe-id-format` ejemplo, se describe el formato de ID de todos los tipos de recursos. Todos los tipos de recursos que admitían el formato de ID corto se cambiaron para usar el formato de ID largo.

```
aws ec2 describe-id-format
```

- Para API obtener más información, consulte [DescribeIdFormat](#) la Referencia de AWS CLI comandos.

## **describe-identity-id-format**

En el siguiente ejemplo de código, se muestra cómo usar `describe-identity-id-format`.

AWS CLI

Para describir el formato de ID de un IAM rol

En el siguiente `describe-identity-id-format` ejemplo, se describe el formato de ID que reciben las instancias creadas por el IAM rol `EC2Role` en tu AWS cuenta.

```
aws ec2 describe-identity-id-format \
  --principal-arn arn:aws:iam::123456789012:role/my-iam-role \
  --resource instance
```



El siguiente resultado indica que las instancias creadas por este rol se reciben IDs en un formato de ID largo.

```
{
  "Statuses": [
    {
      "Deadline": "2016-12-15T00:00:00Z",
      "Resource": "instance",
      "UseLongIds": true
    }
  ]
}
```

Para describir el formato de ID de un IAM usuario

En el siguiente `describe-identity-id-format` ejemplo, se describe el formato de ID que reciben las instantáneas creadas por el IAM usuario `AdminUser` en su AWS cuenta.

```
aws ec2 describe-identity-id-format \
  --principal-arn arn:aws:iam::123456789012:user/AdminUser \
  --resource snapshot
```

El resultado indica que las instantáneas creadas por este usuario se reciben IDs en formato de ID largo.

```
{
  "Statuses": [
    {
      "Deadline": "2016-12-15T00:00:00Z",
      "Resource": "snapshot",
      "UseLongIds": true
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeIdentityIdFormat](#) la Referencia de AWS CLI comandos.

## **describe-image-attribute**

En el siguiente ejemplo de código, se muestra cómo usar `describe-image-attribute`.

## AWS CLI

Para describir los permisos de lanzamiento de un AMI

En este ejemplo, se describen los permisos de lanzamiento del especificadoAMI.

Comando:

```
aws ec2 describe-image-attribute --image-id ami-5731123e --  
attribute LaunchPermission
```

Salida:

```
{  
  "LaunchPermissions": [  
    {  
      "UserId": "123456789012"  
    }  
  ],  
  "ImageId": "ami-5731123e",  
}
```

Para describir los códigos de producto de un AMI

En este ejemplo se describen los códigos de producto del producto especificadoAMI. Ten en cuenta que no AMI tiene códigos de producto.

Comando:

```
aws ec2 describe-image-attribute --image-id ami-5731123e --attribute productCodes
```

Salida:

```
{  
  "ProductCodes": [],  
  "ImageId": "ami-5731123e",  
}
```

- Para API obtener más información, consulte [DescribeImageAttribute](#) la Referencia de AWS CLI comandos.

## describe-images

En el siguiente ejemplo de código, se muestra cómo usar `describe-images`.

### AWS CLI

Ejemplo 1: Para describir un AMI

El siguiente `describe-images` ejemplo describe lo especificado AMI en la región especificada.

```
aws ec2 describe-images \  
  --region us-east-1 \  
  --image-ids ami-1234567890EXAMPLE
```

Salida:

```
{  
  "Images": [  
    {  
      "VirtualizationType": "hvm",  
      "Description": "Provided by Red Hat, Inc.",  
      "PlatformDetails": "Red Hat Enterprise Linux",  
      "EnaSupport": true,  
      "Hypervisor": "xen",  
      "State": "available",  
      "SriovNetSupport": "simple",  
      "ImageId": "ami-1234567890EXAMPLE",  
      "UsageOperation": "RunInstances:0010",  
      "BlockDeviceMappings": [  
        {  
          "DeviceName": "/dev/sda1",  
          "Ebs": {  
            "SnapshotId": "snap-111222333444aaabb",  
            "DeleteOnTermination": true,  
            "VolumeType": "gp2",  
            "VolumeSize": 10,  
            "Encrypted": false  
          }  
        }  
      ],  
      "Architecture": "x86_64",  
      "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-  
GP2",
```

```

        "RootDeviceType": "ebs",
        "OwnerId": "123456789012",
        "RootDeviceName": "/dev/sda1",
        "CreationDate": "2019-05-10T13:17:12.000Z",
        "Public": true,
        "ImageType": "machine",
        "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"
    }
]
}

```

Para obtener más información, consulte [Amazon Machine Images \(AMI\)](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para describir AMIs en función de filtros

El siguiente describe-images ejemplo describe los Windows AMIs proporcionados por Amazon y respaldados por AmazonEBS.

```

aws ec2 describe-images \
  --owners amazon \
  --filters "Name=platform,Values=windows" "Name=root-device-type,Values=ebs"

```

Para ver un ejemplo del resultado de describe-images, consulte el ejemplo 1.

Para ver ejemplos adicionales de uso de filtros, consulta [Cómo publicar y filtrar tus recursos](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 3: Describir AMIs en función de las etiquetas

El siguiente describe-images ejemplo describe todos los AMIs que tienen la etiquetaType=Custom. En el ejemplo, se utiliza el --query parámetro para mostrar solo el AMIIDs.

```

aws ec2 describe-images \
  --filters "Name=tag:Type,Values=Custom" \
  --query 'Images[*].[ImageId]' \
  --output text

```

Salida:

```
ami-1234567890EXAMPLE
```

```
ami-0abcdef1234567890
```

Para ver más ejemplos de uso de filtros de etiquetas, consulta [Cómo trabajar con etiquetas](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeImages](#) la Referencia de AWS CLI comandos.

## describe-import-image-tasks

En el siguiente ejemplo de código, se muestra cómo usar `describe-import-image-tasks`.

### AWS CLI

Para supervisar una tarea de importación de imágenes

El siguiente `describe-import-image-tasks` ejemplo comprueba el estado de la tarea de importación de imágenes especificada.

```
aws ec2 describe-import-image-tasks \
  --import-task-ids import-ami-1234567890abcdef0
```

Resultado de una tarea de importación de imágenes en curso.

```
{
  "ImportImageTasks": [
    {
      "ImportTaskId": "import-ami-1234567890abcdef0",
      "Progress": "28",
      "SnapshotDetails": [
        {
          "DiskImageSize": 705638400.0,
          "Format": "ova",
          "Status": "completed",
          "UserBucket": {
            "S3Bucket": "my-import-bucket",
            "S3Key": "vms/my-server-vm.ova"
          }
        }
      ],
      "Status": "active",
      "StatusMessage": "converting"
    }
  ]
}
```

```

    }
  ]
}

```

Resultado de una tarea de importación de imágenes que se ha completado. El identificador del resultado AMI lo proporciona `ImageId`.

```

{
  "ImportImageTasks": [
    {
      "ImportTaskId": "import-ami-1234567890abcdef0",
      "ImageId": "ami-1234567890abcdef0",
      "SnapshotDetails": [
        {
          "DiskImageSize": 705638400.0,
          "Format": "ova",
          "SnapshotId": "snap-1234567890abcdef0"
          "Status": "completed",
          "UserBucket": {
            "S3Bucket": "my-import-bucket",
            "S3Key": "vms/my-server-vm.ova"
          }
        }
      ],
      "Status": "completed"
    }
  ]
}

```

- Para API obtener más información, consulte [DescribeImportImageTasks](#) la Referencia de AWS CLI comandos.

## describe-import-snapshot-tasks

En el siguiente ejemplo de código, se muestra cómo usar `describe-import-snapshot-tasks`.

### AWS CLI

Para supervisar una tarea de importación de instantáneas

El siguiente `describe-import-snapshot-tasks` ejemplo comprueba el estado de la tarea de importación de instantáneas especificada.

```
aws ec2 describe-import-snapshot-tasks \
  --import-task-ids import-snap-1234567890abcdef0
```

Resultado de una tarea de importación de instantáneas en curso:

```
{
  "ImportSnapshotTasks": [
    {
      "Description": "My server VMDK",
      "ImportTaskId": "import-snap-1234567890abcdef0",
      "SnapshotTaskDetail": {
        "Description": "My server VMDK",
        "DiskImageSize": "705638400.0",
        "Format": "VMDK",
        "Progress": "42",
        "Status": "active",
        "StatusMessage": "downloading/convertng",
        "UserBucket": {
          "S3Bucket": "my-import-bucket",
          "S3Key": "vms/my-server-vm.vmdk"
        }
      }
    }
  ]
}
```

Resultado de una tarea de importación de instantáneas que se ha completado. El ID de la instantánea resultante lo proporciona `SnapshotId`.

```
{
  "ImportSnapshotTasks": [
    {
      "Description": "My server VMDK",
      "ImportTaskId": "import-snap-1234567890abcdef0",
      "SnapshotTaskDetail": {
        "Description": "My server VMDK",
        "DiskImageSize": "705638400.0",
        "Format": "VMDK",
        "SnapshotId": "snap-1234567890abcdef0",
        "Status": "completed",
        "UserBucket": {
          "S3Bucket": "my-import-bucket",

```

```
    "S3Key": "vms/my-server-vm.vmdk"
  }
}
]
```

- Para API obtener más información, consulte [DescribeImportSnapshotTasks](#) la Referencia de AWS CLI comandos.

## describe-instance-attribute

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-attribute`.

### AWS CLI

Para describir el tipo de instancia

En este ejemplo, se describe el tipo de instancia de la instancia especificada.

Comando:

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --  
attribute instanceType
```

Salida:

```
{  
  "InstanceId": "i-1234567890abcdef0"  
  "InstanceType": {  
    "Value": "t1.micro"  
  }  
}
```

Para describir el `disableApiTermination` atributo

En este ejemplo se describe el `disableApiTermination` atributo de la instancia especificada.

Comando:

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --  
attribute disableApiTermination
```



Salida:

```
{
  "InstanceId": "i-1234567890abcdef0"
  "DisableApiTermination": {
    "Value": "false"
  }
}
```

Para describir el mapeo de dispositivos de bloques de una instancia

En este ejemplo, se describe el `blockDeviceMapping` atributo de la instancia especificada.

Comando:

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --
attribute blockDeviceMapping
```

Salida:

```
{
  "InstanceId": "i-1234567890abcdef0"
  "BlockDeviceMappings": [
    {
      "DeviceName": "/dev/sda1",
      "Ebs": {
        "Status": "attached",
        "DeleteOnTermination": true,
        "VolumeId": "vol-049df61146c4d7901",
        "AttachTime": "2013-05-17T22:42:34.000Z"
      }
    },
    {
      "DeviceName": "/dev/sdf",
      "Ebs": {
        "Status": "attached",
        "DeleteOnTermination": false,
        "VolumeId": "vol-049df61146c4d7901",
        "AttachTime": "2013-09-10T23:07:00.000Z"
      }
    }
  ],
}
```

```
}
```

- Para API obtener más información, consulte [DescribeInstanceAttribute](#) la Referencia de AWS CLI comandos.

## describe-instance-connect-endpoints

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-connect-endpoints`.

### AWS CLI

Para describir un punto final de EC2 Instance Connect

En el siguiente `describe-instance-connect-endpoints` ejemplo, se describe el punto final EC2 Instance Connect especificado.

```
aws ec2 describe-instance-connect-endpoints \
  --region us-east-1 \
  --instance-connect-endpoint-ids eice-0123456789example
```

Salida:

```
{
  "InstanceConnectEndpoints": [
    {
      "OwnerId": "111111111111",
      "InstanceConnectEndpointId": "eice-0123456789example",
      "InstanceConnectEndpointArn": "arn:aws:ec2:us-
east-1:111111111111:instance-connect-endpoint/eice-0123456789example",
      "State": "create-complete",
      "StateMessage": "",
      "DnsName": "eice-0123456789example.b67b86ba.ec2-instance-connect-
endpoint.us-east-1.amazonaws.com",
      "NetworkInterfaceIds": [
        "eni-0123456789example"
      ],
      "VpcId": "vpc-0123abcd",
      "AvailabilityZone": "us-east-1d",
      "CreatedAt": "2023-02-07T12:05:37+00:00",
      "SubnetId": "subnet-0123abcd",
      "Tags": []
    }
  ]
}
```

```
    }  
  ]  
}
```

Para obtener más información, consulte [Create an EC2 Instance Connect Endpoint](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeInstanceConnectEndpoints](#) la Referencia de AWS CLI comandos.

## **describe-instance-credit-specifications**

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-credit-specifications`.

### AWS CLI

Para describir la opción de crédito por CPU el uso de una o más instancias

En el siguiente `describe-instance-credit-specifications` ejemplo, se describe la opción de CPU crédito para la instancia especificada.

```
aws ec2 describe-instance-credit-specifications \  
  --instance-ids i-1234567890abcdef0
```

Salida:

```
{  
  "InstanceCreditSpecifications": [  
    {  
      "InstanceId": "i-1234567890abcdef0",  
      "CpuCredits": "unlimited"  
    }  
  ]  
}
```

Para obtener más información, consulta Cómo [trabajar con instancias de rendimiento explosivo](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeInstanceCreditSpecifications](#) la Referencia de AWS CLI comandos.

## describe-instance-event-notification-attributes

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-event-notification-attributes`.

### AWS CLI

Para describir las etiquetas de las notificaciones de eventos programados

En el siguiente `describe-instance-event-notification-attributes` ejemplo, se describen las etiquetas que deben aparecer en las notificaciones de eventos programados.

```
aws ec2 describe-instance-event-notification-attributes
```

Salida:

```
{
  "InstanceTagAttribute": {
    "InstanceTagKeys": [],
    "IncludeAllTagsOfInstance": true
  }
}
```

Para obtener más información, consulte [Eventos programados para sus instancias](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [DescribeInstanceEventNotificationAttributes](#) la Referencia de AWS CLI comandos.

## describe-instance-event-windows

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-event-windows`.

### AWS CLI

Ejemplo 1: Para describir todas las ventanas de eventos

El siguiente `describe-instance-event-windows` ejemplo describe todas las ventanas de eventos de la región especificada.

```
aws ec2 describe-instance-event-windows \
  --region us-east-1
```

**Salida:**

```
{
  "InstanceEventWindows": [
    {
      "InstanceEventWindowId": "iew-0abcdef1234567890",
      "Name": "myEventWindowName",
      "CronExpression": "* 21-23 * * 2,3",
      "AssociationTarget": {
        "InstanceIds": [
          "i-1234567890abcdef0",
          "i-0598c7d356eba48d7"
        ],
        "Tags": [],
        "DedicatedHostIds": []
      },
      "State": "active",
      "Tags": []
    }
    ...
  ],
  "NextToken": "9d624e0c-388b-4862-a31e-a85c64fc1d4a"
}
```

**Ejemplo 2: Para describir una ventana de eventos específica**

El siguiente `describe-instance-event-windows` ejemplo describe un evento específico mediante el uso del `instance-event-window` parámetro para describir una ventana de eventos específica.

```
aws ec2 describe-instance-event-windows \
  --region us-east-1 \
  --instance-event-window-ids iew-0abcdef1234567890
```

**Salida:**

```
{
  "InstanceEventWindows": [
    {
      "InstanceEventWindowId": "iew-0abcdef1234567890",
```

```

    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [
        "i-1234567890abcdef0",
        "i-0598c7d356eba48d7"
      ],
      "Tags": [],
      "DedicatedHostIds": []
    },
    "State": "active",
    "Tags": []
  }
}

```

Ejemplo 3: Para describir las ventanas de eventos que coinciden con uno o más filtros

El siguiente `describe-instance-event-windows` ejemplo describe las ventanas de eventos que coinciden con uno o más filtros mediante el `filter` parámetro. El `instance-id` filtro se utiliza para describir todas las ventanas de eventos asociadas a la instancia especificada. Cuando se utiliza un filtro, este realiza una coincidencia directa. Sin embargo, el filtro `instance-id` es diferente. Si no hay una coincidencia directa con el ID de la instancia, se recurre a asociaciones indirectas con la ventana de eventos, como las etiquetas de la instancia o el ID de host dedicado (si la instancia es un host dedicado).

```

aws ec2 describe-instance-event-windows \
  --region us-east-1 \
  --filters Name=instance-id,Values=i-1234567890abcdef0 \
  --max-results 100 \
  --next-token <next-token-value>

```

Salida:

```

{
  "InstanceEventWindows": [
    {
      "InstanceEventWindowId": "iew-0dbc0adb66f235982",
      "TimeRanges": [
        {
          "StartWeekDay": "sunday",
          "StartHour": 2,
          "EndWeekDay": "sunday",

```

```

        "EndHour": 8
      }
    ],
    "Name": "myEventWindowName",
    "AssociationTarget": {
      "InstanceIds": [],
      "Tags": [],
      "DedicatedHostIds": [
        "h-0140d9a7ecbd102dd"
      ]
    },
    "State": "active",
    "Tags": []
  }
]
}

```

En el resultado del ejemplo, la instancia está en un host dedicado, que está asociado a la ventana de eventos.

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeInstanceEventWindows](#) la Referencia de AWS CLI comandos.

## describe-instance-status

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-status`.

### AWS CLI

Describir el estado de las instancias

En el siguiente ejemplo de `describe-instance-status`, se muestran los detalles de la instancia especificada.

```
aws ec2 describe-instance-status \
  --instance-ids i-1234567890abcdef0
```

Salida:

```
{
  "InstanceStatuses": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "InstanceState": {
        "Code": 16,
        "Name": "running"
      },
      "AvailabilityZone": "us-east-1d",
      "SystemStatus": {
        "Status": "ok",
        "Details": [
          {
            "Status": "passed",
            "Name": "reachability"
          }
        ]
      },
      "InstanceStatus": {
        "Status": "ok",
        "Details": [
          {
            "Status": "passed",
            "Name": "reachability"
          }
        ]
      }
    }
  ]
}
```

Para obtener más información, consulta [Supervisa el estado de tus instancias](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeInstanceStatus](#) la Referencia de AWS CLI comandos.

## **describe-instance-topology**

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-topology`.



## AWS CLI

Para describir la topología de instancias de todas tus instancias

En el siguiente describe-instance-topology ejemplo, se describe la topología de todas las instancias que coinciden con los tipos de instancias compatibles con este comando.

```
aws ec2 describe-instance-topology \  
  --region us-west-2
```

Salida:

```
{  
  "Instances": [  
    {  
      "InstanceId": "i-1111111111example",  
      "InstanceType": "p4d.24xlarge",  
      "GroupName": "my-ml-cpg",  
      "NetworkNodes": [  
        "nn-1111111111example",  
        "nn-2222222222example",  
        "nn-3333333333example"  
      ],  
      "ZoneId": "usw2-az2",  
      "AvailabilityZone": "us-west-2a"  
    },  
    {  
      "InstanceId": "i-2222222222example",  
      "InstanceType": "p4d.24xlarge",  
      "NetworkNodes": [  
        "nn-1111111111example",  
        "nn-2222222222example",  
        "nn-3333333333example"  
      ],  
      "ZoneId": "usw2-az2",  
      "AvailabilityZone": "us-west-2a"  
    },  
    {  
      "InstanceId": "i-3333333333example",  
      "InstanceType": "trn1.32xlarge",  
      "NetworkNodes": [  
        "nn-1212121212example",  
        "nn-1211122211example",
```

```

        "nn-1311133311example"
    ],
    "ZoneId": "usw2-az4",
    "AvailabilityZone": "us-west-2d"
  },
  {
    "InstanceId": "i-444444444example",
    "InstanceType": "trn1.2xlarge",
    "NetworkNodes": [
      "nn-111111111example",
      "nn-5434334334example",
      "nn-1235301234example"
    ],
    "ZoneId": "usw2-az2",
    "AvailabilityZone": "us-west-2a"
  }
],
"NextToken": "SomeEncryptedToken"
}

```

Para obtener más información, incluidos más ejemplos, consulte [Topología de EC2 instancias de Amazon](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeInstanceTopology](#) la Referencia de AWS CLI comandos.

## describe-instance-type-offerings

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-type-offerings`.

### AWS CLI

Ejemplo 1: Para enumerar los tipos de instancias que se ofrecen en una región

En el siguiente `describe-instance-type-offerings` ejemplo, se enumeran los tipos de instancias que se ofrecen en la región configurada como región predeterminada para la AWS CLI.

```
aws ec2 describe-instance-type-offerings
```

Para enumerar los tipos de instancias que se ofrecen en una región diferente, especifique la región mediante el `--region` parámetro.

```
aws ec2 describe-instance-type-offerings \
  --region us-east-2
```

Salida:

```
{
  "InstanceTypeOfferings": [
    {
      "InstanceType": "m5.2xlarge",
      "LocationType": "region",
      "Location": "us-east-2"
    },
    {
      "InstanceType": "t3.micro",
      "LocationType": "region",
      "Location": "us-east-2"
    },
    ...
  ]
}
```

Ejemplo 2: Para enumerar los tipos de instancias que se ofrecen en una zona de disponibilidad

En el siguiente `describe-instance-type-offerings` ejemplo, se enumeran los tipos de instancias que se ofrecen en la zona de disponibilidad especificada. La zona de disponibilidad debe estar en la región especificada.

```
aws ec2 describe-instance-type-offerings \
  --location-type availability-zone \
  --filters Name=location,Values=us-east-2a \
  --region us-east-2
```

Ejemplo 3: Para comprobar si se admite un tipo de instancia

El siguiente `describe-instance-type-offerings` comando indica si el tipo de `c5.xlarge` instancia es compatible con la región especificada.

```
aws ec2 describe-instance-type-offerings \
  --filters Name=instance-type,Values=c5.xlarge \
  --region us-east-2
```

En el siguiente `describe-instance-type-offerings` ejemplo, se enumeran todos los tipos de instancias C5 que se admiten en la región especificada.

```
aws ec2 describe-instance-type-offerings \
  --filters Name=instance-type,Values=c5* \
  --query "InstanceTypeOfferings[].InstanceType" \
  --region us-east-2
```

Salida:

```
[
  "c5d.12xlarge",
  "c5d.9xlarge",
  "c5n.xlarge",
  "c5.xlarge",
  "c5d.metal",
  "c5n.metal",
  "c5.large",
  "c5d.2xlarge",
  "c5n.4xlarge",
  "c5.2xlarge",
  "c5n.large",
  "c5n.9xlarge",
  "c5d.large",
  "c5.18xlarge",
  "c5d.18xlarge",
  "c5.12xlarge",
  "c5n.18xlarge",
  "c5.metal",
  "c5d.4xlarge",
  "c5.24xlarge",
  "c5d.xlarge",
  "c5n.2xlarge",
  "c5d.24xlarge",
  "c5.9xlarge",
  "c5.4xlarge"
]
```

- Para API obtener más información, consulte [DescribeInstanceTypeOfferings](#) la Referencia de AWS CLI comandos.

## describe-instance-types

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-types`.

### AWS CLI

#### Ejemplo 1: Describir un tipo de instancia

En el ejemplo siguiente de `describe-instance-types`, se muestran los detalles del tipo de instancia especificado.

```
aws ec2 describe-instance-types \  
  --instance-types t2.micro
```

Salida:

```
{  
  "InstanceTypes": [  
    {  
      "InstanceType": "t2.micro",  
      "CurrentGeneration": true,  
      "FreeTierEligible": true,  
      "SupportedUsageClasses": [  
        "on-demand",  
        "spot"  
      ],  
      "SupportedRootDeviceTypes": [  
        "ebs"  
      ],  
      "BareMetal": false,  
      "Hypervisor": "xen",  
      "ProcessorInfo": {  
        "SupportedArchitectures": [  
          "i386",  
          "x86_64"  
        ],  
        "SustainedClockSpeedInGhz": 2.5  
      },  
      "VCpuInfo": {  
        "DefaultVCpus": 1,  
        "DefaultCores": 1,  
        "DefaultThreadsPerCore": 1,  
        "ValidCores": [  

```

```
        1
        ],
        "ValidThreadsPerCore": [
            1
        ]
    },
    "MemoryInfo": {
        "SizeInMiB": 1024
    },
    "InstanceStorageSupported": false,
    "EbsInfo": {
        "EbsOptimizedSupport": "unsupported",
        "EncryptionSupport": "supported"
    },
    "NetworkInfo": {
        "NetworkPerformance": "Low to Moderate",
        "MaximumNetworkInterfaces": 2,
        "Ipv4AddressesPerInterface": 2,
        "Ipv6AddressesPerInterface": 2,
        "Ipv6Supported": true,
        "EnaSupport": "unsupported"
    },
    "PlacementGroupInfo": {
        "SupportedStrategies": [
            "partition",
            "spread"
        ]
    },
    "HibernationSupported": false,
    "BurstablePerformanceSupported": true,
    "DedicatedHostsSupported": false,
    "AutoRecoverySupported": true
    }
    ]
}
```

Para obtener más información, consulte [Tipos de instancias](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

Ejemplo 2: Filtrar los tipos de instancias disponibles

Puede especificar un filtro para limitar los resultados a los tipos de instancias que tienen una característica específica. En el siguiente ejemplo de `describe-instance-types`, se enumeran los tipos de instancias que admiten la hibernación.

```
aws ec2 describe-instance-types \  
  --filters Name=hibernation-supported,Values=true --query  
  'InstanceTypes[*].InstanceType'
```

Salida:

```
[  
  "m5.8xlarge",  
  "r3.large",  
  "c3.8xlarge",  
  "r5.large",  
  "m4.4xlarge",  
  "c4.large",  
  "m5.xlarge",  
  "m4.xlarge",  
  "c3.large",  
  "c4.8xlarge",  
  "c4.4xlarge",  
  "c5.xlarge",  
  "c5.12xlarge",  
  "r5.4xlarge",  
  "c5.4xlarge"  
]
```

Para obtener más información, consulte [Tipos de instancias](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [DescribeInstanceTypes](#) la Referencia de AWS CLI comandos.

## **describe-instances**

En el siguiente ejemplo de código, se muestra cómo usar `describe-instances`.

### AWS CLI

#### Ejemplo 1: Describir una instancia

En el siguiente ejemplo de `describe-instances`, se describe la instancia especificada.

```
aws ec2 describe-instances \  
  --instance-ids i-1234567890abcdef0
```

Salida:

```
{  
  "Reservations": [  
    {  
      "Groups": [],  
      "Instances": [  
        {  
          "AmiLaunchIndex": 0,  
          "ImageId": "ami-0abcdef1234567890",  
          "InstanceId": "i-1234567890abcdef0",  
          "InstanceType": "t3.nano",  
          "KeyName": "my-key-pair",  
          "LaunchTime": "2022-11-15T10:48:59+00:00",  
          "Monitoring": {  
            "State": "disabled"  
          },  
          "Placement": {  
            "AvailabilityZone": "us-east-2a",  
            "GroupName": "",  
            "Tenancy": "default"  
          },  
          "PrivateDnsName": "ip-10-0-0-157.us-east-2.compute.internal",  
          "PrivateIpAddress": "10-0-0-157",  
          "ProductCodes": [],  
          "PublicDnsName": "ec2-34-253-223-13.us-  
east-2.compute.amazonaws.com",  
          "PublicIpAddress": "34.253.223.13",  
          "State": {  
            "Code": 16,  
            "Name": "running"  
          },  
          "StateTransitionReason": "",  
          "SubnetId": "subnet-04a636d18e83cfacb",  
          "VpcId": "vpc-1234567890abcdef0",  
          "Architecture": "x86_64",  
          "BlockDeviceMappings": [  
            {
```



```
        "DeviceName": "/dev/xvda",
        "Ebs": {
            "AttachTime": "2022-11-15T10:49:00+00:00",
            "DeleteOnTermination": true,
            "Status": "attached",
            "VolumeId": "vol-02e6ccdca7de29cf2"
        }
    },
    "ClientToken": "1234abcd-1234-abcd-1234-d46a8903e9bc",
    "EbsOptimized": true,
    "EnaSupport": true,
    "Hypervisor": "xen",
    "IamInstanceProfile": {
        "Arn": "arn:aws:iam::111111111111:instance-profile/
AmazonSSMRoleForInstancesQuickSetup",
        "Id": "111111111111111111111111"
    },
    "NetworkInterfaces": [
        {
            "Association": {
                "IpOwnerId": "amazon",
                "PublicDnsName": "ec2-34-253-223-13.us-
east-2.compute.amazonaws.com",
                "PublicIp": "34.253.223.13"
            },
            "Attachment": {
                "AttachTime": "2022-11-15T10:48:59+00:00",
                "AttachmentId": "eni-attach-1234567890abcdefg",
                "DeleteOnTermination": true,
                "DeviceIndex": 0,
                "Status": "attached",
                "NetworkCardIndex": 0
            },
            "Description": "",
            "Groups": [
                {
                    "GroupName": "launch-wizard-146",
                    "GroupId": "sg-1234567890abcdefg"
                }
            ],
            "Ipv6Addresses": [],
            "MacAddress": "00:11:22:33:44:55",
            "NetworkInterfaceId": "eni-1234567890abcdefg",
```

```

    "OwnerId": "104024344472",
    "PrivateDnsName": "ip-10-0-0-157.us-
east-2.compute.internal",
    "PrivateIpAddress": "10-0-0-157",
    "PrivateIpAddresses": [
      {
        "Association": {
          "IpOwnerId": "amazon",
          "PublicDnsName": "ec2-34-253-223-13.us-
east-2.compute.amazonaws.com",
          "PublicIp": "34.253.223.13"
        },
        "Primary": true,
        "PrivateDnsName": "ip-10-0-0-157.us-
east-2.compute.internal",
        "PrivateIpAddress": "10-0-0-157"
      }
    ],
    "SourceDestCheck": true,
    "Status": "in-use",
    "SubnetId": "subnet-1234567890abcdefg",
    "VpcId": "vpc-1234567890abcdefg",
    "InterfaceType": "interface"
  }
],
"RootDeviceName": "/dev/xvda",
"RootDeviceType": "ebs",
"SecurityGroups": [
  {
    "GroupName": "launch-wizard-146",
    "GroupId": "sg-1234567890abcdefg"
  }
],
"SourceDestCheck": true,
"Tags": [
  {
    "Key": "Name",
    "Value": "my-instance"
  }
],
"VirtualizationType": "hvm",
"CpuOptions": {
  "CoreCount": 1,
  "ThreadsPerCore": 2
}

```

```

    },
    "CapacityReservationSpecification": {
      "CapacityReservationPreference": "open"
    },
    "HibernationOptions": {
      "Configured": false
    },
    "MetadataOptions": {
      "State": "applied",
      "HttpTokens": "optional",
      "HttpPutResponseHopLimit": 1,
      "HttpEndpoint": "enabled",
      "HttpProtocolIpv6": "disabled",
      "InstanceMetadataTags": "enabled"
    },
    "EnclaveOptions": {
      "Enabled": false
    },
    "PlatformDetails": "Linux/UNIX",
    "UsageOperation": "RunInstances",
    "UsageOperationUpdateTime": "2022-11-15T10:48:59+00:00",
    "PrivateDnsNameOptions": {
      "HostnameType": "ip-name",
      "EnableResourceNameDnsARecord": true,
      "EnableResourceNameDnsAAAARecord": false
    },
    "MaintenanceOptions": {
      "AutoRecovery": "default"
    }
  }
],
"OwnerId": "111111111111",
"ReservationId": "r-1234567890abcdefg"
}
]
}

```

## Ejemplo 2: Filtrar instancias con el tipo especificado

En el siguiente ejemplo de `describe-instances`, se usan filtros para limitar los resultados a las instancias del tipo especificado.

```
aws ec2 describe-instances \
```

```
--filters Name=instance-type,Values=m5.Large
```

Para ver un ejemplo del resultado, consulte el ejemplo 1.

Para obtener más información, consulta [Listar y filtrar mediante CLI la](#) Guía del EC2 usuario de Amazon.

Ejemplo 3: Filtrar instancias con el tipo y la zona de disponibilidad especificados

En el siguiente ejemplo de `describe-instances`, se usan varios filtros para limitar los resultados a las instancias del tipo especificado que también se encuentran en la zona de disponibilidad especificada.

```
aws ec2 describe-instances \  
  --filters Name=instance-type,Values=t2.micro,t3.micro Name=availability-  
zone,Values=us-east-2c
```

Para ver un ejemplo del resultado, consulte el ejemplo 1.

Ejemplo 4: Para filtrar las instancias con el tipo y la zona de disponibilidad especificados mediante un JSON archivo

En el siguiente `describe-instances` ejemplo, se utiliza un archivo de JSON entrada para realizar el mismo filtrado que en el ejemplo anterior. Cuando los filtros se vuelven más complicados, es más fácil especificarlos en un JSON archivo.

```
aws ec2 describe-instances \  
  --filters file://filters.json
```

Contenidos de `filters.json`:

```
[  
  {  
    "Name": "instance-type",  
    "Values": ["t2.micro", "t3.micro"]  
  },  
  {  
    "Name": "availability-zone",  
    "Values": ["us-east-2c"]  
  }  
]
```

```
] ]
```

Para ver un ejemplo del resultado, consulte el ejemplo 1.

Ejemplo 5: Filtrar instancias con la etiqueta de propietario especificada

En el siguiente ejemplo de `describe-instances`, se usan filtros de etiquetas para limitar los resultados a las instancias que tienen una etiqueta con la clave de etiqueta especificada (propietario), independientemente del valor de la etiqueta.

```
aws ec2 describe-instances \  
  --filters "Name=tag-key,Values=Owner"
```

Para ver un ejemplo del resultado, consulte el ejemplo 1.

Ejemplo 6: Filtrar las instancias con el valor de etiqueta de mi equipo especificado

En el siguiente ejemplo de `describe-instances`, se usan filtros de etiquetas para limitar los resultados a las instancias que tienen una etiqueta con el valor de etiqueta especificado (mi equipo), independientemente de la clave de la etiqueta.

```
aws ec2 describe-instances \  
  --filters "Name=tag-value,Values=my-team"
```

Para ver un ejemplo del resultado, consulte el ejemplo 1.

Ejemplo 7: Filtrar las instancias con la etiqueta de propietario y el valor de mi equipo especificados

En el siguiente ejemplo de `describe-instances`, se usan filtros de etiquetas para limitar los resultados a las instancias que tienen la etiqueta especificada (propietario = mi equipo).

```
aws ec2 describe-instances \  
  --filters "Name=tag:Owner,Values=my-team"
```

Para ver un ejemplo del resultado, consulte el ejemplo 1.

Ejemplo 8: Para mostrar solo la instancia y la subred de todas IDs las instancias

En los `describe-instances` ejemplos siguientes, se utiliza el `--query` parámetro para mostrar solo la instancia y la subred de todas IDs las instancias, en JSON formato.

Linux y macOS:

```
aws ec2 describe-instances \  
  --query 'Reservations[*].Instances[*].{Instance:InstanceId,Subnet:SubnetId}' \  
  --output json
```

Windows:

```
aws ec2 describe-instances ^  
  --query "Reservations[*].Instances[*].{Instance:InstanceId,Subnet:SubnetId}" ^  
  --output json
```

Salida:

```
[  
  {  
    "Instance": "i-057750d42936e468a",  
    "Subnet": "subnet-069beee9b12030077"  
  },  
  {  
    "Instance": "i-001efd250faaa6ffa",  
    "Subnet": "subnet-0b715c6b7db68927a"  
  },  
  {  
    "Instance": "i-027552a73f021f3bd",  
    "Subnet": "subnet-0250c25a1f4e15235"  
  }  
  ...  
]
```

Ejemplo 9: Para filtrar instancias del tipo especificado y mostrar solo su instancia IDs

En el siguiente `describe-instances` ejemplo, se utilizan filtros para limitar los resultados a las instancias del tipo especificado y el `--query` parámetro para mostrar solo la instanciaIDs.

```
aws ec2 describe-instances \  
  --filters "Name=instance-type,Values=t2.micro" \  
  --query "Reservations[*].Instances[*].[InstanceId]" \  
  --output text
```

Salida:

```
i-031c0dc19de2fb70c
i-00d8bfff789a736b75
i-0b715c6b7db68927a
i-0626d4edd54f1286d
i-00b8ae04f9f99908e
i-0fc71c25d2374130c
```

Ejemplo 10: Para filtrar instancias del tipo especificado y mostrar solo su instanciaIDs, la zona de disponibilidad y el valor de etiqueta especificado

En los siguientes ejemplos de `describe-instances`, se muestran el ID de la instancia, la zona de disponibilidad y el valor de la etiqueta `Name` para las instancias que tienen una etiqueta con el nombre `tag-key`, en formato de tabla.

Linux y macOS:

```
aws ec2 describe-instances \
  --filters Name=tag-key,Values=Name \
  --query 'Reservations[*].Instances[*].
{Instance:InstanceId,AZ:Placement.AvailabilityZone,Name:Tags[?Key==`Name`]]
[0].Value}' \
  --output table
```

Windows:

```
aws ec2 describe-instances ^
  --filters Name=tag-key,Values=Name ^
  --query "Reservations[*].Instances[*].
{Instance:InstanceId,AZ:Placement.AvailabilityZone,Name:Tags[?Key=='Name']]
[0].Value}" ^
  --output table
```

Salida:

```
-----
|                               DescribeInstances                               |
+-----+-----+-----+
|      AZ      | Instance      |      Name      |
+-----+-----+-----+
| us-east-2b  | i-057750d42936e468a | my-prod-server |
+-----+-----+-----+
```

```
| us-east-2a | i-001efd250faaa6ffa | test-server-1 |
| us-east-2a | i-027552a73f021f3bd | test-server-2 |
+-----+-----+-----+
```

### Ejemplo 11: Describir las instancias de un grupo con ubicación en particiones

En el siguiente ejemplo de `describe-instances`, se describe la instancia especificada. El resultado incluye la información de ubicación de la instancia, la cual contiene el nombre del grupo con ubicación y el número de partición de la instancia.

```
aws ec2 describe-instances \
  --instance-ids i-0123a456700123456 \
  --query "Reservations[*].Instances[*].Placement"
```

Salida:

```
[
  [
    {
      "AvailabilityZone": "us-east-1c",
      "GroupName": "HDFS-Group-A",
      "PartitionNumber": 3,
      "Tenancy": "default"
    }
  ]
]
```

Para obtener más información, consulte [Descripción de las instancias de un grupo de ubicaciones](#) en la Guía del EC2 usuario de Amazon.

### Ejemplo 12: Filtrar las instancias con el grupo con ubicación y el número de partición especificados

En el siguiente ejemplo de `describe-instances`, se filtran los resultados solo para las instancias con el grupo con ubicación y el número de partición especificados.

```
aws ec2 describe-instances \
  --filters "Name=placement-group-name,Values=HDFS-Group-A" "Name=placement-partition-number,Values=7"
```



A continuación, se muestra solo la información relevante de la salida.

```
"Instances": [  
  {  
    "InstanceId": "i-0123a456700123456",  
    "InstanceType": "r4.large",  
    "Placement": {  
      "AvailabilityZone": "us-east-1c",  
      "GroupName": "HDFS-Group-A",  
      "PartitionNumber": 7,  
      "Tenancy": "default"  
    }  
  },  
  {  
    "InstanceId": "i-9876a543210987654",  
    "InstanceType": "r4.large",  
    "Placement": {  
      "AvailabilityZone": "us-east-1c",  
      "GroupName": "HDFS-Group-A",  
      "PartitionNumber": 7,  
      "Tenancy": "default"  
    }  
  },  
]
```

Para obtener más información, consulte [Descripción de las instancias de un grupo de ubicaciones](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 13: Filtrar las instancias que están configuradas para permitir el acceso a las etiquetas desde los metadatos de la instancia

En el siguiente ejemplo de `describe-instances`, se filtran los resultados solo para las instancias que están configuradas para permitir el acceso a las etiquetas de la instancia desde los metadatos de la instancia.

```
aws ec2 describe-instances \  
  --filters "Name=metadata-options.instance-metadata-tags,Values=enabled" \  
  --query "Reservations[*].Instances[*].InstanceId" \  
  --output text
```

El resultado esperado es el siguiente.

```
i-1234567890abcdefg
```

```
i-abcdefg1234567890
i-11111111aaaaaaaaa
i-aaaaaaaa111111111
```

Para obtener más información, consulta Cómo [trabajar con etiquetas de instancia en metadatos](#) de instancias en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulta [DescribeInstances](#) la Referencia de AWS CLI comandos.

## describe-internet-gateways

En el siguiente ejemplo de código, se muestra cómo usar `describe-internet-gateways`.

### AWS CLI

Para describir una puerta de enlace a Internet

El siguiente `describe-internet-gateways` ejemplo describe la puerta de enlace de Internet especificada.

```
aws ec2 describe-internet-gateways \
  --internet-gateway-ids igw-0d0fb496b3EXAMPLE
```

Salida:

```
{
  "InternetGateways": [
    {
      "Attachments": [
        {
          "State": "available",
          "VpcId": "vpc-0a60eb65b4EXAMPLE"
        }
      ],
      "InternetGatewayId": "igw-0d0fb496b3EXAMPLE",
      "OwnerId": "123456789012",
      "Tags": [
        {
          "Key": "Name",
          "Value": "my-igw"
        }
      ]
    }
  ]
}
```

```

    ]
  }
}

```

Para obtener más información, consulta [las pasarelas de Internet](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DescribeInternetGateways](#) la Referencia de AWS CLI comandos.

## describe-ipam-pools

En el siguiente ejemplo de código, se muestra cómo usar `describe-ipam-pools`.

### AWS CLI

Para ver los detalles de un IPAM grupo

El siguiente `describe-ipam-pools` ejemplo muestra los detalles de los grupos.

(Linux):

```

aws ec2 describe-ipam-pools \
  --filters Name=owner-id,Values=123456789012 Name=ipam-scope-id,Values=ipam-
  scope-02fc38cd4c48e7d38

```

(Windows):

```

aws ec2 describe-ipam-pools ^
  --filters Name=owner-id,Values=123456789012 Name=ipam-scope-id,Values=ipam-
  scope-02fc38cd4c48e7d38

```

Salida:

```

{
  "IpamPools": [
    {
      "OwnerId": "123456789012",

```

```

    "IpamPoolId": "ipam-pool-02ec043a19bbe5d08",
    "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-02ec043a19bbe5d08",
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-02fc38cd4c48e7d38",
    "IpamScopeType": "private",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
    "IpamRegion": "us-east-1",
    "Locale": "None",
    "PoolDepth": 1,
    "State": "create-complete",
    "AutoImport": true,
    "AddressFamily": "ipv4",
    "AllocationMinNetmaskLength": 16,
    "AllocationMaxNetmaskLength": 26,
    "AllocationDefaultNetmaskLength": 24,
    "AllocationResourceTags": [
      {
        "Key": "Environment",
        "Value": "Preprod"
      }
    ],
    "Tags": [
      {
        "Key": "Name",
        "Value": "Preprod pool"
      }
    ]
  }
]
}

```

- Para API obtener más información, consulte [DescribeIpamPools](#) la Referencia de AWS CLI comandos.

## describe-ipam-resource-discoveries

En el siguiente ejemplo de código, se muestra cómo usar `describe-ipam-resource-discoveries`.

### AWS CLI

Ejemplo 1: Vea los detalles completos de los descubrimientos de recursos

En este ejemplo, usted es un IPAM administrador delegado que desea crear y compartir una detección de recursos con el IPAM administrador de otra AWS organización para que el administrador pueda administrar y supervisar las direcciones IP de los recursos de su organización.

Este ejemplo puede resultar útil si:

Intentaste crear una detección de recursos, pero aparece un error que indica que has alcanzado el límite de 1. Te das cuenta de que es posible que ya hayas creado un descubrimiento de recursos y quieras verlo en tu cuenta. Tienes recursos en una región que no están siendo descubiertos por el. IPAM Desea ver lo `--operating-regions` definido para el recurso y asegurarse de haber agregado la región correcta como región operativa para poder descubrir los recursos que contiene.

En el siguiente `describe-ipam-resource-discoveries` ejemplo, se enumeran los detalles del descubrimiento de recursos en su AWS cuenta. Puede realizar un descubrimiento de recursos por AWS región.

```
aws ec2 describe-ipam-resource-discoveries \  
  --region us-east-1
```

Salida:

```
{  
  "IpamResourceDiscoveries": [  
    {  
      "OwnerId": "149977607591",  
      "IpamResourceDiscoveryId": "ipam-res-disco-0f8bdee9067137c0d",  
      "IpamResourceDiscoveryArn": "arn:aws:ec2::149977607591:ipam-resource-  
discovery/ipam-res-disco-0f8bdee9067137c0d",  
      "IpamResourceDiscoveryRegion": "us-east-1",  
      "OperatingRegions": [  
        {  
          "RegionName": "us-east-1"  
        }  
      ],  
      "IsDefault": false,  
      "State": "create-complete",  
      "Tags": []  
    }  
  ]  
}
```

```
}
```

Para obtener más información, consulta la [sección Integración IPAM con cuentas ajenas a tu organización](#) en la Guía del VPC IPAM usuario de Amazon.

Ejemplo 2: Ver solo la detección de recursos IDs

En el siguiente `describe-ipam-resource-discoveries` ejemplo, se muestra el ID de la detección de recursos en su AWS cuenta. Puede realizar un descubrimiento de recursos por AWS región.

```
aws ec2 describe-ipam-resource-discoveries \
  --query "IpamResourceDiscoveries[*].IpamResourceDiscoveryId" \
  --output text
```

Salida:

```
ipam-res-disco-0481e39b242860333
```

Para obtener más información, consulta la [sección Integración IPAM con cuentas ajenas a tu organización](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [DescribeIpamResourceDiscoveries](#) la Referencia de AWS CLI comandos.

## **describe-ipam-resource-discovery-associations**

En el siguiente ejemplo de código, se muestra cómo usar `describe-ipam-resource-discovery-associations`.

AWS CLI

Para ver todas las asociaciones de descubrimiento de recursos con su IPAM

En este ejemplo, eres un administrador IPAM delegado que ha asociado las detecciones de recursos a las tuyas IPAM para integrar otras cuentas con las tuyas IPAM. Te has dado cuenta de que no estás descubriendo los recursos en las regiones operativas de la detección de recursos como esperabas. IPAM desea comprobar el estado y el estado de la detección de recursos para asegurarse de que la cuenta que la creó sigue activa y que la detección de recursos sigue siendo compartida.

`--region` Debe ser su región de origen IPAM.

En el siguiente `describe-ipam-resource-discovery-associations` ejemplo, se enumeran las asociaciones de descubrimiento de recursos de su AWS cuenta.

```
aws ec2 describe-ipam-resource-discovery-associations \  
  --region us-east-1
```

Salida:

```
{  
  "IpamResourceDiscoveryAssociations": [  
    {  
      "OwnerId": "320805250157",  
      "IpamResourceDiscoveryAssociationId": "ipam-res-disco-  
assoc-05e6b45eca5bf5cf7",  
      "IpamResourceDiscoveryAssociationArn": "arn:aws:ec2::320805250157:ipam-  
resource-discovery-association/ipam-res-disco-  
assoc-05e6b45eca5bf5cf7",  
      "IpamResourceDiscoveryId": "ipam-res-disco-0f4ef577a9f37a162",  
      "IpamId": "ipam-005f921c17ebd5107",  
      "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",  
      "IpamRegion": "us-east-1",  
      "IsDefault": true,  
      "ResourceDiscoveryStatus": "active",  
      "State": "associate-complete",  
      "Tags": []  
    },  
    {  
      "OwnerId": "149977607591",  
      "IpamResourceDiscoveryAssociationId": "ipam-res-disco-  
assoc-0dfd21ae189ab5f62",  
      "IpamResourceDiscoveryAssociationArn": "arn:aws:ec2::149977607591:ipam-  
resource-discovery-association/ipam-res-disco-  
assoc-0dfd21ae189ab5f62",  
      "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",  
      "IpamId": "ipam-005f921c17ebd5107",  
      "IpamArn": "arn:aws:ec2::149977607591:ipam/ipam-005f921c17ebd5107",  
      "IpamRegion": "us-east-1",  
      "IsDefault": false,  
      "ResourceDiscoveryStatus": "active",  
      "State": "create-complete",  
      "Tags": []  
    }  
  ]  
}
```

```
}

```

En este ejemplo, después de ejecutar este comando, observa que tiene una detección de recursos no predeterminada (`"IsDefault": false`) que es `"ResourceDiscoveryStatus": "not-found"` y `"State": "create-complete"`. Se ha cerrado la cuenta del propietario del descubrimiento de recursos. Si, en otro caso, observa que es `"ResourceDiscoveryStatus": "not-found"` y `"State": "associate-complete"`, esto indica que ha ocurrido una de las siguientes situaciones:

El propietario del descubrimiento de recursos eliminó la detección de recursos. El propietario de la detección de recursos dejó de compartir la detección de recursos.

Para obtener más información, consulta la [sección Integración IPAM con cuentas ajenas a tu organización](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [DescribeIpamResourceDiscoveryAssociations](#) la Referencia de AWS CLI comandos.

## describe-ipam-scopes

En el siguiente ejemplo de código, se muestra cómo usar `describe-ipam-scopes`.

### AWS CLI

Para ver los detalles de un IPAM osciloscopio

En el siguiente `describe-ipam-scopes` ejemplo, se muestran los detalles de los ámbitos.

```
aws ec2 describe-ipam-scopes \
  --filters Name=owner-id,Values=123456789012 Name=ipam-
  id,Values=ipam-08440e7a3acde3908
```

Salida:

```
{
  "IpamScopes": [
    {
      "OwnerId": "123456789012",
      "IpamScopeId": "ipam-scope-02fc38cd4c48e7d38",
      "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-02fc38cd4c48e7d38",
      "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
```



```
    "IpamRegion": "us-east-1",
    "IpamScopeType": "private",
    "IsDefault": true,
    "PoolCount": 2,
    "State": "create-complete",
    "Tags": []
  },
  {
    "OwnerId": "123456789012",
    "IpamScopeId": "ipam-scope-0b9eed026396dbc16",
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0b9eed026396dbc16",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
    "IpamRegion": "us-east-1",
    "IpamScopeType": "public",
    "IsDefault": true,
    "PoolCount": 0,
    "State": "create-complete",
    "Tags": []
  },
  {
    "OwnerId": "123456789012",
    "IpamScopeId": "ipam-scope-0f1aff29486355c22",
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0f1aff29486355c22",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
    "IpamRegion": "us-east-1",
    "IpamScopeType": "private",
    "IsDefault": false,
    "Description": "Example description",
    "PoolCount": 0,
    "State": "create-complete",
    "Tags": [
      {
        "Key": "Name",
        "Value": "Example name value"
      }
    ]
  }
]
}
```

- Para API obtener más información, consulte [DescribeIpamScopes](#) la Referencia de AWS CLI comandos.

## describe-ipams

En el siguiente ejemplo de código, se muestra cómo usar `describe-ipams`.

### AWS CLI

Para ver los detalles de un IPAM

En el siguiente `describe-ipams` ejemplo se muestran los detalles de un IPAM.

```
aws ec2 describe-ipams \
  --filters Name=owner-id,Values=123456789012
```

Salida:

```
{
  "Ipams": [
    {
      "OwnerId": "123456789012",
      "IpamId": "ipam-08440e7a3acde3908",
      "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
      "IpamRegion": "us-east-1",
      "PublicDefaultScopeId": "ipam-scope-0b9eed026396dbc16",
      "PrivateDefaultScopeId": "ipam-scope-02fc38cd4c48e7d38",
      "ScopeCount": 3,
      "OperatingRegions": [
        {
          "RegionName": "us-east-1"
        },
        {
          "RegionName": "us-east-2"
        },
        {
          "RegionName": "us-west-1"
        }
      ],
      "State": "create-complete",
      "Tags": [
        {
```

```

        "Key": "Name",
        "Value": "ExampleIPAM"
      }
    ]
  }
}

```

- Para API obtener más información, consulte [DescribeIPAMs](#) la Referencia de AWS CLI comandos.

## describe-ipv6-pools

En el siguiente ejemplo de código, se muestra cómo usar `describe-ipv6-pools`.

### AWS CLI

Para describir sus grupos IPv6 de direcciones

En el siguiente `describe-ipv6-pools` ejemplo, se muestran los detalles de todos los grupos de IPv6 direcciones.

```
aws ec2 describe-ipv6-pools
```

Salida:

```

{
  "Ipv6Pools": [
    {
      "PoolId": "ipv6pool-ec2-012345abc12345abc",
      "PoolCidrBlocks": [
        {
          "Cidr": "2001:db8:123::/48"
        }
      ],
      "Tags": [
        {
          "Key": "pool-1",
          "Value": "public"
        }
      ]
    }
  ]
}

```

```
]
}
```

- Para API obtener más información, consulte [DescribeIpv6Pools](#) en la referencia de AWS CLI comandos.

## describe-key-pairs

En el siguiente ejemplo de código, se muestra cómo usar `describe-key-pairs`.

### AWS CLI

Mostrar un par de claves

En el siguiente ejemplo de `describe-key-pairs`, se muestra información sobre el par de claves especificado.

```
aws ec2 describe-key-pairs \
  --key-names my-key-pair
```

Salida:

```
{
  "KeyPairs": [
    {
      "KeyPairId": "key-0b94643da6EXAMPLE",
      "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
      "KeyName": "my-key-pair",
      "KeyType": "rsa",
      "Tags": [],
      "CreateTime": "2022-05-27T21:51:16.000Z"
    }
  ]
}
```

Para obtener más información, consulta [Describir las claves públicas](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeKeyPairs](#) la Referencia de AWS CLI comandos.

## describe-launch-template-versions

En el siguiente ejemplo de código, se muestra cómo usar `describe-launch-template-versions`.

### AWS CLI

Para describir las versiones de las plantillas de lanzamiento

En este ejemplo se describen las versiones de la plantilla de lanzamiento especificada.

Comando:

```
aws ec2 describe-launch-template-versions --launch-template-id lt-068f72b72934aff71
```

Salida:

```
{
  "LaunchTemplateVersions": [
    {
      "LaunchTemplateId": "lt-068f72b72934aff71",
      "LaunchTemplateName": "Webservers",
      "VersionNumber": 3,
      "CreatedBy": "arn:aws:iam::123456789102:root",
      "LaunchTemplateData": {
        "KeyName": "kp-us-east",
        "ImageId": "ami-6057e21a",
        "InstanceType": "t2.small",
        "NetworkInterfaces": [
          {
            "SubnetId": "subnet-7b16de0c",
            "DeviceIndex": 0,
            "Groups": [
              "sg-7c227019"
            ]
          }
        ]
      },
      "DefaultVersion": false,
      "CreateTime": "2017-11-20T13:19:54.000Z"
    },
    {
      "LaunchTemplateId": "lt-068f72b72934aff71",
      "LaunchTemplateName": "Webservers",
```

```
"VersionNumber": 2,
"CreatedBy": "arn:aws:iam::123456789102:root",
"LaunchTemplateData": {
  "KeyName": "kp-us-east",
  "ImageId": "ami-6057e21a",
  "InstanceType": "t2.medium",
  "NetworkInterfaces": [
    {
      "SubnetId": "subnet-1a2b3c4d",
      "DeviceIndex": 0,
      "Groups": [
        "sg-7c227019"
      ]
    }
  ]
},
"DefaultVersion": false,
"CreateTime": "2017-11-20T13:12:32.000Z"
},
{
  "LaunchTemplateId": "lt-068f72b72934aff71",
  "LaunchTemplateName": "Webservers",
  "VersionNumber": 1,
  "CreatedBy": "arn:aws:iam::123456789102:root",
  "LaunchTemplateData": {
    "UserData": "",
    "KeyName": "kp-us-east",
    "ImageId": "ami-aabbcc11",
    "InstanceType": "t2.medium",
    "NetworkInterfaces": [
      {
        "SubnetId": "subnet-7b16de0c",
        "DeviceIndex": 0,
        "DeleteOnTermination": false,
        "Groups": [
          "sg-7c227019"
        ],
        "AssociatePublicIpAddress": true
      }
    ]
  },
  "DefaultVersion": true,
  "CreateTime": "2017-11-20T12:52:33.000Z"
}
```

```
]
}
```

- Para API obtener más información, consulte [DescribeLaunchTemplateVersions](#) la Referencia de AWS CLI comandos.

## describe-launch-templates

En el siguiente ejemplo de código, se muestra cómo usar `describe-launch-templates`.

### AWS CLI

Para describir las plantillas de lanzamiento

En este ejemplo se describen las plantillas de lanzamiento.

Comando:

```
aws ec2 describe-launch-templates
```

Salida:

```
{
  "LaunchTemplates": [
    {
      "LatestVersionNumber": 2,
      "LaunchTemplateId": "lt-0e06d290751193123",
      "LaunchTemplateName": "TemplateForWebServer",
      "DefaultVersionNumber": 2,
      "CreatedBy": "arn:aws:iam::123456789012:root",
      "CreateTime": "2017-11-27T09:30:23.000Z"
    },
    {
      "LatestVersionNumber": 6,
      "LaunchTemplateId": "lt-0c45b5e061ec98456",
      "LaunchTemplateName": "DBServersTemplate",
      "DefaultVersionNumber": 1,
      "CreatedBy": "arn:aws:iam::123456789012:root",
      "CreateTime": "2017-11-20T09:25:22.000Z"
    },
    {
      "LatestVersionNumber": 1,
```

```

    "LaunchTemplateId": "lt-0d47d774e8e52dabc",
    "LaunchTemplateName": "MyLaunchTemplate2",
    "DefaultVersionNumber": 1,
    "CreatedBy": "arn:aws:iam::123456789012:root",
    "CreateTime": "2017-11-02T12:06:21.000Z"
  },
  {
    "LatestVersionNumber": 3,
    "LaunchTemplateId": "lt-01e5f948eb4f589d6",
    "LaunchTemplateName": "testingtemplate2",
    "DefaultVersionNumber": 1,
    "CreatedBy": "arn:aws:sts::123456789012:assumed-role/AdminRole/
i-03ee35176e2e5aabc",
    "CreateTime": "2017-12-01T08:19:48.000Z"
  },
]
}

```

- Para API obtener más información, consulte [DescribeLaunchTemplates](#) la Referencia de AWS CLI comandos.

## describe-local-gateway-route-table-virtual-interface-group-associations

En el siguiente ejemplo de código, se muestra cómo usar `describe-local-gateway-route-table-virtual-interface-group-associations`.

### AWS CLI

Para describir las asociaciones entre los grupos de interfaces virtuales y las tablas de enrutamiento de las puertas de enlace locales

En el siguiente `describe-local-gateway-route-table-virtual-interface-group-associations` ejemplo, se describen las asociaciones entre los grupos de interfaces virtuales y las tablas de enrutamiento de las puertas de enlace locales de su AWS cuenta.

```
aws ec2 describe-local-gateway-route-table-virtual-interface-group-associations
```

Salida:

```

{
  "LocalGatewayRouteTableVirtualInterfaceGroupAssociations": [

```



```

    {
      "LocalGatewayRouteTableVirtualInterfaceGroupAssociationId": "lgw-vif-
grp-assoc-07145b276bEXAMPLE",
      "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
      "LocalGatewayId": "lgw-0ab1c23d4eEXAMPLE",
      "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE",
      "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:123456789012:local-
gateway-route-table/lgw-rtb-059615ef7dEXAMPLE",
      "OwnerId": "123456789012",
      "State": "associated",
      "Tags": []
    }
  ]
}

```

Para obtener más información, consulta Cómo [trabajar con puertas de enlace locales](#) en la Guía del usuario de AWS Outposts.

- Para API obtener más información, consulte la Referencia [DescribeLocalGatewayRouteTableVirtualInterfaceGroupAssociations](#) de AWS CLI comandos.

## describe-local-gateway-route-table-vpc-associations

En el siguiente ejemplo de código, se muestra cómo usar `describe-local-gateway-route-table-vpc-associations`.

### AWS CLI

Para describir las asociaciones entre VPCs las tablas de enrutamiento de una puerta de enlace local

El siguiente `describe-local-gateway-route-table-vpc-associations` ejemplo muestra información sobre la asociación especificada entre VPCs las tablas de enrutamiento de la puerta de enlace local.

```

aws ec2 describe-local-gateway-route-table-vpc-associations \
  --local-gateway-route-table-vpc-association-ids Lgw-vpc-assoc-0e0f27af15EXAMPLE

```

Salida:

```
{
```

```

    "LocalGatewayRouteTableVpcAssociation": {
      "LocalGatewayRouteTableVpcAssociationId": "lgw-vpc-assoc-0e0f27af1EXAMPLE",
      "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE",
      "LocalGatewayId": "lgw-09b493aa7cEXAMPLE",
      "VpcId": "vpc-0efe9bde08EXAMPLE",
      "State": "associated"
    }
  }
}

```

Para obtener más información, consulte [Tablas de enrutamiento de puerta de enlace locales](#) en la Guía del usuario de Outposts.

- Para API obtener más información, consulte [DescribeLocalGatewayRouteTableVpcAssociations](#) la Referencia de AWS CLI comandos.

## describe-local-gateway-route-tables

En el siguiente ejemplo de código, se muestra cómo usar `describe-local-gateway-route-tables`.

### AWS CLI

Para describir las tablas de rutas de su puerta de enlace local

El siguiente `describe-local-gateway-route-tables` ejemplo muestra detalles sobre las tablas de enrutamiento de la puerta de enlace local.

```
aws ec2 describe-local-gateway-route-tables
```

Salida:

```

{
  "LocalGatewayRouteTables": [
    {
      "LocalGatewayRouteTableId": "lgw-rtb-059615ef7deEXAMPLE",
      "LocalGatewayId": "lgw-09b493aa7cEXAMPLE",
      "OutpostArn": "arn:aws:outposts:us-west-2:111122223333:outpost/
op-0dc11b66edEXAMPLE",
      "State": "available"
    }
  ]
}

```

```
}
```

- Para API obtener más información, consulte [DescribeLocalGatewayRouteTables](#) la Referencia de AWS CLI comandos.

## describe-local-gateway-virtual-interface-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-local-gateway-virtual-interface-groups`.

### AWS CLI

Para describir los grupos de interfaces virtuales de las puertas de enlace locales

En el siguiente `describe-local-gateway-virtual-interface-groups` ejemplo, se describen los grupos de interfaces virtuales de la puerta de enlace local de su AWS cuenta.

```
aws ec2 describe-local-gateway-virtual-interface-groups
```

Salida:

```
{
  "LocalGatewayVirtualInterfaceGroups": [
    {
      "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
      "LocalGatewayVirtualInterfaceIds": [
        "lgw-vif-01a23bc4d5EXAMPLE",
        "lgw-vif-543ab21012EXAMPLE"
      ],
      "LocalGatewayId": "lgw-0ab1c23d4eEXAMPLE",
      "OwnerId": "123456789012",
      "Tags": []
    }
  ]
}
```

Para obtener más información, consulta [Cómo trabajar con puertas de enlace locales](#) en la Guía del usuario de AWS Outposts.

- Para API obtener más información, consulte la Referencia [DescribeLocalGatewayVirtualInterfaceGroups](#) de AWS CLI comandos.

## describe-local-gateway-virtual-interfaces

En el siguiente ejemplo de código, se muestra cómo usar `describe-local-gateway-virtual-interfaces`.

### AWS CLI

Para describir las interfaces virtuales de las puertas de enlace locales

En el siguiente `describe-local-gateway-virtual-interfaces` ejemplo, se describen las interfaces virtuales de la puerta de enlace local de su AWS cuenta.

```
aws ec2 describe-local-gateway-virtual-interfaces
```

Salida:

```
{
  "LocalGatewayVirtualInterfaces": [
    {
      "LocalGatewayVirtualInterfaceId": "lgw-vif-01a23bc4d5EXAMPLE",
      "LocalGatewayId": "lgw-0ab1c23d4eEXAMPLE",
      "Vlan": 2410,
      "LocalAddress": "0.0.0.0/0",
      "PeerAddress": "0.0.0.0/0",
      "LocalBgpAsn": 65010,
      "PeerBgpAsn": 65000,
      "OwnerId": "123456789012",
      "Tags": []
    },
    {
      "LocalGatewayVirtualInterfaceId": "lgw-vif-543ab21012EXAMPLE",
      "LocalGatewayId": "lgw-0ab1c23d4eEXAMPLE",
      "Vlan": 2410,
      "LocalAddress": "0.0.0.0/0",
      "PeerAddress": "0.0.0.0/0",
      "LocalBgpAsn": 65010,
      "PeerBgpAsn": 65000,
      "OwnerId": "123456789012",
      "Tags": []
    }
  ]
}
```

Para obtener más información, consulta Cómo [trabajar con puertas de enlace locales](#) en la Guía del usuario de AWS Outposts.

- Para API obtener más información, consulte la Referencia [DescribeLocalGatewayVirtualInterfaces](#) de AWS CLI comandos.

## describe-local-gateways

En el siguiente ejemplo de código, se muestra cómo usar `describe-local-gateways`.

### AWS CLI

Para describir sus puertas de enlace locales

En el siguiente `describe-local-gateways` ejemplo, se muestran los detalles de las puertas de enlace locales que están disponibles para usted.

```
aws ec2 describe-local-gateways
```

Salida:

```
{
  "LocalGateways": [
    {
      "LocalGatewayId": "lgw-09b493aa7cEXAMPLE",
      "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-0dc11b66ed59f995a",
      "OwnerId": "123456789012",
      "State": "available"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeLocalGateways](#) la Referencia de AWS CLI comandos.

## describe-locked-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `describe-locked-snapshots`.

## AWS CLI

Para describir el estado de bloqueo de una instantánea

El siguiente `describe-locked-snapshots` ejemplo describe el estado de bloqueo de la instantánea especificada.

```
aws ec2 describe-locked-snapshots \  
  --snapshot-ids snap-0b5e733b4a8df6e0d
```

Salida:

```
{  
  "Snapshots": [  
    {  
      "OwnerId": "123456789012",  
      "SnapshotId": "snap-0b5e733b4a8df6e0d",  
      "LockState": "governance",  
      "LockDuration": 365,  
      "LockCreatedOn": "2024-05-05T00:56:06.208000+00:00",  
      "LockDurationStartTime": "2024-05-05T00:56:06.208000+00:00",  
      "LockExpiresOn": "2025-05-05T00:56:06.208000+00:00"  
    }  
  ]  
}
```

Para obtener más información, consulta [Snapshot Lock](#) en la Guía del EBS usuario de Amazon.

- Para API obtener más información, consulte [DescribeLockedSnapshots](#) la Referencia de AWS CLI comandos.

## **describe-managed-prefix-lists**

En el siguiente ejemplo de código, se muestra cómo usar `describe-managed-prefix-lists`.

## AWS CLI

Para describir las listas de prefijos gestionadas

En el siguiente `describe-managed-prefix-lists` ejemplo, se describen las listas de prefijos propiedad AWS de la cuenta. 123456789012

```
aws ec2 describe-managed-prefix-lists \  
  --owner-ids 123456789012
```

```
--filters Name=owner-id,Values=123456789012
```

Salida:

```
{
  "PrefixLists": [
    {
      "PrefixListId": "pl-11223344556677aab",
      "AddressFamily": "IPv6",
      "State": "create-complete",
      "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-11223344556677aab",
      "PrefixListName": "vpc-ipv6-cidrs",
      "MaxEntries": 25,
      "Version": 1,
      "Tags": [],
      "OwnerId": "123456789012"
    },
    {
      "PrefixListId": "pl-0123456abcabcabc1",
      "AddressFamily": "IPv4",
      "State": "active",
      "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-0123456abcabcabc1",
      "PrefixListName": "vpc-cidrs",
      "MaxEntries": 10,
      "Version": 1,
      "Tags": [],
      "OwnerId": "123456789012"
    }
  ]
}
```

Para obtener más información, consulta [Listas de prefijos gestionadas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DescribeManagedPrefixLists](#) la Referencia de AWS CLI comandos.

## **describe-moving-addresses**

En el siguiente ejemplo de código, se muestra cómo usar `describe-moving-addresses`.

## AWS CLI

Para describir sus direcciones de mudanza

En este ejemplo, se describen todas las direcciones IP elásticas en movimiento.

Comando:

```
aws ec2 describe-moving-addresses
```

Salida:

```
{
  "MovingAddressStatuses": [
    {
      "PublicIp": "198.51.100.0",
      "MoveStatus": "MovingToVpc"
    }
  ]
}
```

En este ejemplo, se describen todas las direcciones que se están trasladando a la VPC plataforma EC2 -.

Comando:

```
aws ec2 describe-moving-addresses --filters Name=moving-status,Values=MovingToVpc
```

- Para API obtener más información, consulte [DescribeMovingAddresses](#) la Referencia de AWS CLI comandos.

## describe-nat-gateways

En el siguiente ejemplo de código, se muestra cómo usar `describe-nat-gateways`.

### AWS CLI

Ejemplo 1: Para describir una NAT puerta de enlace pública

El siguiente `describe-nat-gateways` ejemplo describe la NAT puerta de enlace pública especificada.



```
aws ec2 describe-nat-gateways \  
--nat-gateway-id nat-01234567890abcdef
```

Salida:

```
{  
  "NatGateways": [  
    {  
      "CreateTime": "2023-08-25T01:56:51.000Z",  
      "NatGatewayAddresses": [  
        {  
          "AllocationId": "eipalloc-0790180cd2EXAMPLE",  
          "NetworkInterfaceId": "eni-09cc4b2558794f7f9",  
          "PrivateIp": "10.0.0.211",  
          "PublicIp": "54.85.121.213",  
          "AssociationId": "eipassoc-04d295cc9b8815b24",  
          "IsPrimary": true,  
          "Status": "succeeded"  
        },  
        {  
          "AllocationId": "eipalloc-0be6ecac95EXAMPLE",  
          "NetworkInterfaceId": "eni-09cc4b2558794f7f9",  
          "PrivateIp": "10.0.0.74",  
          "PublicIp": "3.211.231.218",  
          "AssociationId": "eipassoc-0f96bdca17EXAMPLE",  
          "IsPrimary": false,  
          "Status": "succeeded"  
        }  
      ],  
      "NatGatewayId": "nat-01234567890abcdef",  
      "State": "available",  
      "SubnetId": "subnet-655eab5f08EXAMPLE",  
      "VpcId": "vpc-098eb5ef58EXAMPLE",  
      "Tags": [  
        {  
          "Key": "Name",  
          "Value": "public-nat"  
        }  
      ],  
      "ConnectivityType": "public"  
    }  
  ]  
}
```

```
}
```

Ejemplo 2: Para describir una NAT puerta de enlace privada

El siguiente describe-nat-gateways ejemplo describe la NAT puerta de enlace privada especificada.

```
aws ec2 describe-nat-gateways \  
  --nat-gateway-id nat-1234567890abcdef0
```

Salida:

```
{  
  "NatGateways": [  
    {  
      "CreateTime": "2023-08-25T00:50:05.000Z",  
      "NatGatewayAddresses": [  
        {  
          "NetworkInterfaceId": "eni-0065a61b324d1897a",  
          "PrivateIp": "10.0.20.240",  
          "IsPrimary": true,  
          "Status": "succeeded"  
        },  
        {  
          "NetworkInterfaceId": "eni-0065a61b324d1897a",  
          "PrivateIp": "10.0.20.33",  
          "IsPrimary": false,  
          "Status": "succeeded"  
        },  
        {  
          "NetworkInterfaceId": "eni-0065a61b324d1897a",  
          "PrivateIp": "10.0.20.197",  
          "IsPrimary": false,  
          "Status": "succeeded"  
        }  
      ],  
      "NatGatewayId": "nat-1234567890abcdef0",  
      "State": "available",  
      "SubnetId": "subnet-08fc749671EXAMPLE",  
      "VpcId": "vpc-098eb5ef58EXAMPLE",  
      "Tags": [  
        {  
          "Key": "Name",
```

```

        "Value": "private-nat"
      }
    ],
    "ConnectivityType": "private"
  }
]
}

```

Para obtener más información, consulta [NATlas pasarelas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DescribeNatGateways](#) la Referencia de AWS CLI comandos.

## describe-network-acls

En el siguiente ejemplo de código, se muestra cómo usar `describe-network-acls`.

### AWS CLI

Para describir su red ACLs

En el siguiente `describe-network-acls` ejemplo, se recuperan los detalles de la redACLs.

```
aws ec2 describe-network-acls
```

Salida:

```

{
  "NetworkAcls": [
    {
      "Associations": [
        {
          "NetworkAclAssociationId": "aclassoc-0c1679dc41EXAMPLE",
          "NetworkAclId": "acl-0ea1f54ca7EXAMPLE",
          "SubnetId": "subnet-0931fc2fa5EXAMPLE"
        }
      ],
      "Entries": [
        {
          "CidrBlock": "0.0.0.0/0",
          "Egress": true,
          "Protocol": "-1",

```

```
        "RuleAction": "allow",
        "RuleNumber": 100
    },
    {
        "CidrBlock": "0.0.0.0/0",
        "Egress": true,
        "Protocol": "-1",
        "RuleAction": "deny",
        "RuleNumber": 32767
    },
    {
        "CidrBlock": "0.0.0.0/0",
        "Egress": false,
        "Protocol": "-1",
        "RuleAction": "allow",
        "RuleNumber": 100
    },
    {
        "CidrBlock": "0.0.0.0/0",
        "Egress": false,
        "Protocol": "-1",
        "RuleAction": "deny",
        "RuleNumber": 32767
    }
],
"IsDefault": true,
"NetworkAclId": "acl-0ea1f54ca7EXAMPLE",
"Tags": [],
"VpcId": "vpc-06e4ab6c6cEXAMPLE",
"OwnerId": "111122223333"
},
{
    "Associations": [],
    "Entries": [
        {
            "CidrBlock": "0.0.0.0/0",
            "Egress": true,
            "Protocol": "-1",
            "RuleAction": "allow",
            "RuleNumber": 100
        },
        {
            "Egress": true,
            "Ipv6CidrBlock": ":::/0",
```

```
    "Protocol": "-1",
    "RuleAction": "allow",
    "RuleNumber": 101
  },
  {
    "CidrBlock": "0.0.0.0/0",
    "Egress": true,
    "Protocol": "-1",
    "RuleAction": "deny",
    "RuleNumber": 32767
  },
  {
    "Egress": true,
    "Ipv6CidrBlock": "::/0",
    "Protocol": "-1",
    "RuleAction": "deny",
    "RuleNumber": 32768
  },
  {
    "CidrBlock": "0.0.0.0/0",
    "Egress": false,
    "Protocol": "-1",
    "RuleAction": "allow",
    "RuleNumber": 100
  },
  {
    "Egress": false,
    "Ipv6CidrBlock": "::/0",
    "Protocol": "-1",
    "RuleAction": "allow",
    "RuleNumber": 101
  },
  {
    "CidrBlock": "0.0.0.0/0",
    "Egress": false,
    "Protocol": "-1",
    "RuleAction": "deny",
    "RuleNumber": 32767
  },
  {
    "Egress": false,
    "Ipv6CidrBlock": "::/0",
    "Protocol": "-1",
    "RuleAction": "deny",
```

```

        "RuleNumber": 32768
      }
    ],
    "IsDefault": true,
    "NetworkAclId": "acl-0e2a78e4e2EXAMPLE",
    "Tags": [],
    "VpcId": "vpc-03914afb3eEXAMPLE",
    "OwnerId": "111122223333"
  }
]
}

```

Para obtener más información, consulte [Red ACLs](#) en la Guía del AWS VPC usuario.

- Para API obtener más información, consulte [DescribeNetworkAcls](#) la Referencia de AWS CLI comandos.

## describe-network-insights-access-scope-analyses

En el siguiente ejemplo de código, se muestra cómo usar `describe-network-insights-access-scope-analyses`.

### AWS CLI

Para describir los análisis del alcance del acceso a Network Insights

En el siguiente `describe-network-insights-access-scope-analyses` ejemplo, se describe el análisis del alcance de acceso de su AWS cuenta.

```

aws ec2 describe-network-insights-access-scope-analyses \
  --region us-east-1

```

Salida:

```

{
  "NetworkInsightsAccessScopeAnalyses": [
    {
      "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789111",
      "NetworkInsightsAccessScopeAnalysisArn": "arn:aws:ec2:us-
east-1:123456789012:network-insights-access-scope-analysis/nisa-123456789111",
      "NetworkInsightsAccessScopeId": "nis-123456789222",
      "Status": "succeeded",
    }
  ]
}

```

```
        "StartDate": "2022-01-25T19:45:36.842000+00:00",
        "FindingsFound": "true",
        "Tags": []
    }
]
}
```

Para obtener más información, consulte [Cómo empezar a utilizar Network Access Analyzer mediante AWS CLI](#) la guía Network Access Analyzer.

- Para API obtener más información, consulte la Referencia [DescribeNetworkInsightsAccessScopeAnalyses](#) de AWS CLI comandos.

## describe-network-insights-access-scopes

En el siguiente ejemplo de código, se muestra cómo usar `describe-network-insights-access-scopes`.

### AWS CLI

Para describir los ámbitos de acceso de Network Insights

En el siguiente `describe-network-insights-access-scopes` ejemplo, se describen los análisis del alcance de acceso de su cuenta. AWS

```
aws ec2 describe-network-insights-access-scopes \
  --region us-east-1
```

Salida:

```
{
  "NetworkInsightsAccessScopes": [
    {
      "NetworkInsightsAccessScopeId": "nis-123456789111",
      "NetworkInsightsAccessScopeArn": "arn:aws:ec2:us-
east-1:123456789012:network-insights-access-scope/nis-123456789111",
      "CreateDate": "2021-11-29T21:12:41.416000+00:00",
      "UpdatedDate": "2021-11-29T21:12:41.416000+00:00",
      "Tags": []
    }
  ]
}
```

Para obtener más información, consulte [Cómo empezar a utilizar Network Access Analyzer mediante la Guía del AWS CLI](#) Network Access Analyzer.

- Para API obtener más información, consulte la Referencia [DescribeNetworkInsightsAccessScopes](#) de AWS CLI comandos.

## describe-network-insights-analyses

En el siguiente ejemplo de código, se muestra cómo usar `describe-network-insights-analyses`.

### AWS CLI

Para ver los resultados de un análisis de ruta

El siguiente `describe-network-insights-analyses` ejemplo describe el análisis especificado. En este ejemplo, el origen es una puerta de enlace a Internet, el destino es una EC2 instancia y el protocolo es TCP. El análisis Status se `succeeded` realizó correctamente y no se pudo acceder a la ruta `false`. `NetworkPathFound` El código explicativo `ENI_SG_RULES_MISMATCH` indica que el grupo de seguridad de la instancia no contiene una regla que permita el tráfico en el puerto de destino.

```
aws ec2 describe-network-insights-analyses \
  --network-insights-analysis-ids nia-02207aa13eb480c7a
```

Salida:

```
{
  "NetworkInsightsAnalyses": [
    {
      "NetworkInsightsAnalysisId": "nia-02207aa13eb480c7a",
      "NetworkInsightsAnalysisArn": "arn:aws:ec2:us-
east-1:123456789012:network-insights-analysis/nia-02207aa13eb480c7a",
      "NetworkInsightsPathId": "nip-0b26f224f1d131fa8",
      "StartDate": "2021-01-20T22:58:37.495Z",
      "Status": "succeeded",
      "NetworkPathFound": false,
      "Explanations": [
        {
          "Direction": "ingress",
          "ExplanationCode": "ENI_SG_RULES_MISMATCH",
          "NetworkInterface": {
```



```

        "Id": "eni-0a25edef15a6cc08c",
        "Arn": "arn:aws:ec2:us-east-1:123456789012:network-
interface/eni-0a25edef15a6cc08c"
    },
    "SecurityGroups": [
        {
            "Id": "sg-02f0d35a850ba727f",
            "Arn": "arn:aws:ec2:us-east-1:123456789012:security-
group/sg-02f0d35a850ba727f"
        }
    ],
    "Subnet": {
        "Id": "subnet-004ff41eccb4d1194",
        "Arn": "arn:aws:ec2:us-east-1:123456789012:subnet/
subnet-004ff41eccb4d1194"
    },
    "Vpc": {
        "Id": "vpc-f1663d98ad28331c7",
        "Arn": "arn:aws:ec2:us-east-1:123456789012:vpc/vpc-
f1663d98ad28331c7"
    }
}
],
"Tags": []
}
]
}

```

Para obtener más información, consulte [Cómo empezar a utilizar la Guía del AWS CLI Reachability Analyzer](#).

- Para API obtener más información, consulte la Referencia de [DescribeNetworkInsightsAnalyses](#) comandos AWS CLI .

## describe-network-insights-paths

En el siguiente ejemplo de código, se muestra cómo usar `describe-network-insights-paths`.

### AWS CLI

Para describir una ruta

El siguiente `describe-network-insights-paths` ejemplo describe la ruta especificada.

```
aws ec2 describe-network-insights-paths \  
--network-insights-path-ids nip-0b26f224f1d131fa8
```

Salida:

```
{  
  "NetworkInsightsPaths": [  
    {  
      "NetworkInsightsPathId": "nip-0b26f224f1d131fa8",  
      "NetworkInsightsPathArn": "arn:aws:ec2:us-east-1:123456789012:network-  
insights-path/nip-0b26f224f1d131fa8",  
      "CreateDate": "2021-01-20T22:43:46.933Z",  
      "Source": "igw-0797cccdc9d73b0e5",  
      "Destination": "i-0495d385ad28331c7",  
      "Protocol": "tcp"  
    }  
  ]  
}
```

Para obtener más información, consulte [Cómo empezar a utilizar la Guía del AWS CLI Reachability Analyzer](#).

- Para API obtener más información, consulte la Referencia de [DescribeNetworkInsightsPaths](#) comandos AWS CLI .

## describe-network-interface-attribute

En el siguiente ejemplo de código, se muestra cómo usar `describe-network-interface-attribute`.

AWS CLI

Para describir el atributo de conexión de una interfaz de red

Este comando de ejemplo describe el `attachment` atributo de la interfaz de red especificada.

Comando:

```
aws ec2 describe-network-interface-attribute --network-interface-id eni-686ea200 --  
attribute attachment
```

Salida:

```
{
  "NetworkInterfaceId": "eni-686ea200",
  "Attachment": {
    "Status": "attached",
    "DeviceIndex": 0,
    "AttachTime": "2015-05-21T20:02:20.000Z",
    "InstanceId": "i-1234567890abcdef0",
    "DeleteOnTermination": true,
    "AttachmentId": "eni-attach-43348162",
    "InstanceOwnerId": "123456789012"
  }
}
```

Para describir el atributo de descripción de una interfaz de red

Este comando de ejemplo describe el `description` atributo de la interfaz de red especificada.

Comando:

```
aws ec2 describe-network-interface-attribute --network-interface-id eni-686ea200 --
attribute description
```

Salida:

```
{
  "NetworkInterfaceId": "eni-686ea200",
  "Description": {
    "Value": "My description"
  }
}
```

Para describir el `groupSet` atributo de una interfaz de red

Este comando de ejemplo describe el `groupSet` atributo de la interfaz de red especificada.

Comando:

```
aws ec2 describe-network-interface-attribute --network-interface-id eni-686ea200 --
attribute groupSet
```

Salida:

```
{
  "NetworkInterfaceId": "eni-686ea200",
  "Groups": [
    {
      "GroupName": "my-security-group",
      "GroupId": "sg-903004f8"
    }
  ]
}
```

Para describir el `sourceDestCheck` atributo de una interfaz de red

Este comando de ejemplo describe el `sourceDestCheck` atributo de la interfaz de red especificada.

Comando:

```
aws ec2 describe-network-interface-attribute --network-interface-id eni-686ea200 --
attribute sourceDestCheck
```

Salida:

```
{
  "NetworkInterfaceId": "eni-686ea200",
  "SourceDestCheck": {
    "Value": true
  }
}
```

- Para API obtener más información, consulte [DescribeNetworkInterfaceAttribute](#) la Referencia de AWS CLI comandos.

## **describe-network-interface-permissions**

En el siguiente ejemplo de código, se muestra cómo usar `describe-network-interface-permissions`.

AWS CLI

Para describir los permisos de la interfaz de red

En este ejemplo se describen todos los permisos de la interfaz de red.

Comando:

```
aws ec2 describe-network-interface-permissions
```

Salida:

```
{
  "NetworkInterfacePermissions": [
    {
      "PermissionState": {
        "State": "GRANTED"
      },
      "NetworkInterfacePermissionId": "eni-perm-06fd19020ede149ea",
      "NetworkInterfaceId": "eni-b909511a",
      "Permission": "INSTANCE-ATTACH",
      "AwsAccountId": "123456789012"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeNetworkInterfacePermissions](#) la Referencia de AWS CLI comandos.

## describe-network-interfaces

En el siguiente ejemplo de código, se muestra cómo usar `describe-network-interfaces`.

AWS CLI

Para describir las interfaces de red

En este ejemplo se describen todas las interfaces de red.

Comando:

```
aws ec2 describe-network-interfaces
```

Salida:

```
{
```

```
"NetworkInterfaces": [
  {
    "Status": "in-use",
    "MacAddress": "02:2f:8f:b0:cf:75",
    "SourceDestCheck": true,
    "VpcId": "vpc-a01106c2",
    "Description": "my network interface",
    "Association": {
      "PublicIp": "203.0.113.12",
      "AssociationId": "eipassoc-0fbb766a",
      "PublicDnsName": "ec2-203-0-113-12.compute-1.amazonaws.com",
      "IpOwnerId": "123456789012"
    },
    "NetworkInterfaceId": "eni-e5aa89a3",
    "PrivateIpAddresses": [
      {
        "PrivateDnsName": "ip-10-0-1-17.ec2.internal",
        "Association": {
          "PublicIp": "203.0.113.12",
          "AssociationId": "eipassoc-0fbb766a",
          "PublicDnsName": "ec2-203-0-113-12.compute-1.amazonaws.com",
          "IpOwnerId": "123456789012"
        },
        "Primary": true,
        "PrivateIpAddress": "10.0.1.17"
      }
    ],
    "RequesterManaged": false,
    "Ipv6Addresses": [],
    "PrivateDnsName": "ip-10-0-1-17.ec2.internal",
    "AvailabilityZone": "us-east-1d",
    "Attachment": {
      "Status": "attached",
      "DeviceIndex": 1,
      "AttachTime": "2013-11-30T23:36:42.000Z",
      "InstanceId": "i-1234567890abcdef0",
      "DeleteOnTermination": false,
      "AttachmentId": "eni-attach-66c4350a",
      "InstanceOwnerId": "123456789012"
    },
    "Groups": [
      {
        "GroupName": "default",
        "GroupId": "sg-8637d3e3"
      }
    ]
  }
]
```

```
    }
  ],
  "SubnetId": "subnet-b61f49f0",
  "OwnerId": "123456789012",
  "TagSet": [],
  "PrivateIpAddress": "10.0.1.17"
},
{
  "Status": "in-use",
  "MacAddress": "02:58:f5:ef:4b:06",
  "SourceDestCheck": true,
  "VpcId": "vpc-a01106c2",
  "Description": "Primary network interface",
  "Association": {
    "PublicIp": "198.51.100.0",
    "IpOwnerId": "amazon"
  },
  "NetworkInterfaceId": "eni-f9ba99bf",
  "PrivateIpAddresses": [
    {
      "Association": {
        "PublicIp": "198.51.100.0",
        "IpOwnerId": "amazon"
      },
      "Primary": true,
      "PrivateIpAddress": "10.0.1.149"
    }
  ],
  "RequesterManaged": false,
  "Ipv6Addresses": [],
  "AvailabilityZone": "us-east-1d",
  "Attachment": {
    "Status": "attached",
    "DeviceIndex": 0,
    "AttachTime": "2013-11-30T23:35:33.000Z",
    "InstanceId": "i-0598c7d356eba48d7",
    "DeleteOnTermination": true,
    "AttachmentId": "eni-attach-1b9db777",
    "InstanceOwnerId": "123456789012"
  },
  "Groups": [
    {
      "GroupName": "default",
      "GroupId": "sg-8637d3e3"
    }
  ]
}
```

```

    }
  ],
  "SubnetId": "subnet-b61f49f0",
  "OwnerId": "123456789012",
  "TagSet": [],
  "PrivateIpAddress": "10.0.1.149"
}
]
}

```

En este ejemplo, se describen las interfaces de red que tienen una etiqueta con la clave Purpose y el valor Prod.

Comando:

```
aws ec2 describe-network-interfaces --filters Name=tag:Purpose,Values=Prod
```

Salida:

```

{
  "NetworkInterfaces": [
    {
      "Status": "available",
      "MacAddress": "12:2c:bd:f9:bf:17",
      "SourceDestCheck": true,
      "VpcId": "vpc-8941ebec",
      "Description": "ProdENI",
      "NetworkInterfaceId": "eni-b9a5ac93",
      "PrivateIpAddresses": [
        {
          "PrivateDnsName": "ip-10-0-1-55.ec2.internal",
          "Primary": true,
          "PrivateIpAddress": "10.0.1.55"
        },
        {
          "PrivateDnsName": "ip-10-0-1-117.ec2.internal",
          "Primary": false,
          "PrivateIpAddress": "10.0.1.117"
        }
      ],
      "RequesterManaged": false,
      "PrivateDnsName": "ip-10-0-1-55.ec2.internal",
      "AvailabilityZone": "us-east-1d",

```



```
    "Ipv6Addresses": [],
    "Groups": [
      {
        "GroupName": "MySG",
        "GroupId": "sg-905002f5"
      }
    ],
    "SubnetId": "subnet-31d6c219",
    "OwnerId": "123456789012",
    "TagSet": [
      {
        "Value": "Prod",
        "Key": "Purpose"
      }
    ],
    "PrivateIpAddress": "10.0.1.55"
  }
]
}
```

- Para API obtener más información, consulte [DescribeNetworkInterfaces](#) la Referencia de AWS CLI comandos.

## describe-placement-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-placement-groups`.

### AWS CLI

Para describir sus grupos de colocación

Este comando de ejemplo describe todos los grupos de colocación.

Comando:

```
aws ec2 describe-placement-groups
```

Salida:

```
{
  "PlacementGroups": [
    {
```

```
        "GroupName": "my-cluster",
        "State": "available",
        "Strategy": "cluster"
    },
    ...
]
}
```

- Para API obtener más información, consulte [DescribePlacementGroups](#) la Referencia de AWS CLI comandos.

## describe-prefix-lists

En el siguiente ejemplo de código, se muestra cómo usar `describe-prefix-lists`.

### AWS CLI

Para describir las listas de prefijos

En este ejemplo se enumeran todas las listas de prefijos disponibles para la región.

Comando:

```
aws ec2 describe-prefix-lists
```

Salida:

```
{
  "PrefixLists": [
    {
      "PrefixListName": "com.amazonaws.us-east-1.s3",
      "Cidrs": [
        "54.231.0.0/17"
      ],
      "PrefixListId": "pl-63a5400a"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribePrefixLists](#) la Referencia de AWS CLI comandos.

## describe-principal-id-format

En el siguiente ejemplo de código, se muestra cómo usar `describe-principal-id-format`.

### AWS CLI

Para describir el formato de ID para IAM los usuarios y roles con el formato de ID largo activado

En el siguiente `describe-principal-id-format` ejemplo, se describe el formato de ID del usuario raíz, todos los IAM roles y todos IAM los usuarios con el formato de ID largo activado.

```
aws ec2 describe-principal-id-format \  
  --resource instance
```

Salida:

```
{  
  "Principals": [  
    {  
      "Arn": "arn:aws:iam::123456789012:root",  
      "Statuses": [  
        {  
          "Deadline": "2016-12-15T00:00:00.000Z",  
          "Resource": "reservation",  
          "UseLongIds": true  
        },  
        {  
          "Deadline": "2016-12-15T00:00:00.000Z",  
          "Resource": "instance",  
          "UseLongIds": true  
        },  
        {  
          "Deadline": "2016-12-15T00:00:00.000Z",  
          "Resource": "volume",  
          "UseLongIds": true  
        }  
      ],  
    },  
    ...  
  ]  
}
```

- Para API obtener más información, consulte [DescribePrincipalIdFormat](#) la Referencia de AWS CLI comandos.

## **describe-public-ipv4-pools**

En el siguiente ejemplo de código, se muestra cómo usar `describe-public-ipv4-pools`.

### AWS CLI

Para describir sus grupos de IPv4 direcciones públicas

En el siguiente `describe-public-ipv4-pools` ejemplo, se muestran detalles sobre los grupos de direcciones que se crearon al aprovisionar rangos de IPv4 direcciones públicas mediante Bring Your Own IP Addresses (BYOIP).

```
aws ec2 describe-public-ipv4-pools
```

Salida:

```
{
  "PublicIpv4Pools": [
    {
      "PoolId": "ipv4pool-ec2-1234567890abcdef0",
      "PoolAddressRanges": [
        {
          "FirstAddress": "203.0.113.0",
          "LastAddress": "203.0.113.255",
          "AddressCount": 256,
          "AvailableAddressCount": 256
        }
      ],
      "TotalAddressCount": 256,
      "TotalAvailableAddressCount": 256
    }
  ]
}
```

- Para API obtener más información, consulte [DescribePublicIpv4Pools](#) en AWS CLI Command Reference.

## describe-regions

En el siguiente ejemplo de código, se muestra cómo usar `describe-regions`.

### AWS CLI

Ejemplo 1: Describir todas las regiones habilitadas

En el siguiente ejemplo de `describe-regions`, se describen las regiones que están habilitadas para su cuenta.

```
aws ec2 describe-regions
```

Salida:

```
{
  "Regions": [
    {
      "Endpoint": "ec2.eu-north-1.amazonaws.com",
      "RegionName": "eu-north-1",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.ap-south-1.amazonaws.com",
      "RegionName": "ap-south-1",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.eu-west-3.amazonaws.com",
      "RegionName": "eu-west-3",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.eu-west-2.amazonaws.com",
      "RegionName": "eu-west-2",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.eu-west-1.amazonaws.com",
      "RegionName": "eu-west-1",
      "OptInStatus": "opt-in-not-required"
    },
    {
```

```
    "Endpoint": "ec2.ap-northeast-3.amazonaws.com",
    "RegionName": "ap-northeast-3",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-northeast-2.amazonaws.com",
    "RegionName": "ap-northeast-2",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-northeast-1.amazonaws.com",
    "RegionName": "ap-northeast-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.sa-east-1.amazonaws.com",
    "RegionName": "sa-east-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ca-central-1.amazonaws.com",
    "RegionName": "ca-central-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-southeast-1.amazonaws.com",
    "RegionName": "ap-southeast-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-southeast-2.amazonaws.com",
    "RegionName": "ap-southeast-2",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.eu-central-1.amazonaws.com",
    "RegionName": "eu-central-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-east-1.amazonaws.com",
    "RegionName": "us-east-1",
    "OptInStatus": "opt-in-not-required"
  },
},
```

```

    {
      "Endpoint": "ec2.us-east-2.amazonaws.com",
      "RegionName": "us-east-2",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.us-west-1.amazonaws.com",
      "RegionName": "us-west-1",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.us-west-2.amazonaws.com",
      "RegionName": "us-west-2",
      "OptInStatus": "opt-in-not-required"
    }
  ]
}

```

Para obtener más información, consulta [Regiones y zonas](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Describir las regiones habilitadas con un punto de conexión cuyo nombre contiene una cadena específica

En el siguiente ejemplo de `describe-regions`, se describen todas las regiones que ha habilitado y que tienen la cadena “us” en el punto de conexión.

```

aws ec2 describe-regions \
  --filters "Name=endpoint,Values=*us*"

```

Salida:

```

{
  "Regions": [
    {
      "Endpoint": "ec2.us-east-1.amazonaws.com",
      "RegionName": "us-east-1"
    },
    {
      "Endpoint": "ec2.us-east-2.amazonaws.com",
      "RegionName": "us-east-2"
    },
    {

```

```
    "Endpoint": "ec2.us-west-1.amazonaws.com",
    "RegionName": "us-west-1"
  },
  {
    "Endpoint": "ec2.us-west-2.amazonaws.com",
    "RegionName": "us-west-2"
  }
]
```

Para obtener más información, consulta [Regiones y zonas](#) en la Guía del EC2 usuario de Amazon.

### Ejemplo 3: Describir todas las regiones

En el siguiente ejemplo de `describe-regions`, se describen todas las regiones disponibles, incluidas las que están deshabilitadas.

```
aws ec2 describe-regions \
  --all-regions
```

Salida:

```
{
  "Regions": [
    {
      "Endpoint": "ec2.eu-north-1.amazonaws.com",
      "RegionName": "eu-north-1",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.ap-south-1.amazonaws.com",
      "RegionName": "ap-south-1",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.eu-west-3.amazonaws.com",
      "RegionName": "eu-west-3",
      "OptInStatus": "opt-in-not-required"
    },
    {
      "Endpoint": "ec2.eu-west-2.amazonaws.com",
      "RegionName": "eu-west-2",
```



```
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.eu-west-1.amazonaws.com",
    "RegionName": "eu-west-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-northeast-3.amazonaws.com",
    "RegionName": "ap-northeast-3",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.me-south-1.amazonaws.com",
    "RegionName": "me-south-1",
    "OptInStatus": "not-opted-in"
  },
  {
    "Endpoint": "ec2.ap-northeast-2.amazonaws.com",
    "RegionName": "ap-northeast-2",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-northeast-1.amazonaws.com",
    "RegionName": "ap-northeast-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.sa-east-1.amazonaws.com",
    "RegionName": "sa-east-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ca-central-1.amazonaws.com",
    "RegionName": "ca-central-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-east-1.amazonaws.com",
    "RegionName": "ap-east-1",
    "OptInStatus": "not-opted-in"
  },
  {
    "Endpoint": "ec2.ap-southeast-1.amazonaws.com",
```

```
    "RegionName": "ap-southeast-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.ap-southeast-2.amazonaws.com",
    "RegionName": "ap-southeast-2",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.eu-central-1.amazonaws.com",
    "RegionName": "eu-central-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-east-1.amazonaws.com",
    "RegionName": "us-east-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-east-2.amazonaws.com",
    "RegionName": "us-east-2",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-west-1.amazonaws.com",
    "RegionName": "us-west-1",
    "OptInStatus": "opt-in-not-required"
  },
  {
    "Endpoint": "ec2.us-west-2.amazonaws.com",
    "RegionName": "us-west-2",
    "OptInStatus": "opt-in-not-required"
  }
]
}
```

Para obtener más información, consulta [Regiones y zonas](#) en la Guía del EC2 usuario de Amazon.

#### Ejemplo 4: Enumerar únicamente los nombres de las regiones

En el siguiente ejemplo de `describe-regions`, se usa el parámetro `--query` para filtrar la salida y devolver solo los nombres de las regiones como texto.

```
aws ec2 describe-regions \
  --all-regions \
  --query "Regions[].{Name:RegionName}" \
  --output text
```

Salida:

```
eu-north-1
ap-south-1
eu-west-3
eu-west-2
eu-west-1
ap-northeast-3
ap-northeast-2
me-south-1
ap-northeast-1
sa-east-1
ca-central-1
ap-east-1
ap-southeast-1
ap-southeast-2
eu-central-1
us-east-1
us-east-2
us-west-1
us-west-2
```

Para obtener más información, consulta [Regiones y zonas](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeRegions](#) la Referencia de AWS CLI comandos.

## **describe-replace-root-volume-tasks**

En el siguiente ejemplo de código, se muestra cómo usar `describe-replace-root-volume-tasks`.

AWS CLI

Ejemplo 1: Para ver información sobre una tarea específica de reemplazo de un volumen raíz

El siguiente `describe-replace-root-volume-tasks` ejemplo describe la tarea de reemplazo del volumen raíz `replacevol-0111122223333abcd`.

```
aws ec2 describe-replace-root-volume-tasks \
  --replace-root-volume-task-ids replacevol-0111122223333abcd
```

Salida:

```
{
  "ReplaceRootVolumeTasks": [
    {
      "ReplaceRootVolumeTaskId": "replacevol-0111122223333abcd",
      "Tags": [],
      "InstanceId": "i-0123456789abcdefa",
      "TaskState": "succeeded",
      "StartTime": "2022-03-14T15:16:28Z",
      "CompleteTime": "2022-03-14T15:16:52Z"
    }
  ]
}
```

Para obtener más información, consulte [Reemplazar un volumen raíz](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

Ejemplo 2: Para ver información sobre todas las tareas de reemplazo del volumen raíz de una instancia específica

El siguiente `describe-replace-root-volume-tasks` ejemplo describe todas las tareas de reemplazo del volumen raíz, por ejemplo, `i-0123456789abcdefa`.

```
aws ec2 describe-replace-root-volume-tasks \
  --filters Name=instance-id,Values=i-0123456789abcdefa
```

Salida:

```
{
  "ReplaceRootVolumeTasks": [
    {
      "ReplaceRootVolumeTaskId": "replacevol-0111122223333abcd",
      "Tags": [],

```

```

    "InstanceId": "i-0123456789abcdefa",
    "TaskState": "succeeded",
    "StartTime": "2022-03-14T15:06:38Z",
    "CompleteTime": "2022-03-14T15:07:03Z"
  },
  {
    "ReplaceRootVolumeTaskId": "replacevol-044445555555abcd",
    "Tags": [],
    "InstanceId": "i-0123456789abcdefa",
    "TaskState": "succeeded",
    "StartTime": "2022-03-14T15:16:28Z",
    "CompleteTime": "2022-03-14T15:16:52Z"
  }
]
}

```

Para obtener más información, consulte [Reemplazar un volumen raíz](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte [DescribeReplaceRootVolumeTasks](#) la Referencia de AWS CLI comandos.

## describe-reserved-instances-listings

En el siguiente ejemplo de código, se muestra cómo usar `describe-reserved-instances-listings`.

### AWS CLI

Para describir una lista de instancias reservadas

En el siguiente `describe-reserved-instances-listings` ejemplo, se recupera información sobre la lista de instancias reservadas especificada.

```

aws ec2 describe-reserved-instances-listings \
  --reserved-instances-listing-id 5ec28771-05ff-4b9b-aa31-9e57dexample

```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DescribeReservedInstancesListings](#) la Referencia de AWS CLI comandos.

## describe-reserved-instances-modifications

En el siguiente ejemplo de código, se muestra cómo usar `describe-reserved-instances-modifications`.

### AWS CLI

Para describir las modificaciones de las instancias reservadas

Este comando de ejemplo describe todas las solicitudes de modificación de instancias reservadas que se han enviado para su cuenta.

Comando:

```
aws ec2 describe-reserved-instances-modifications
```

Salida:

```
{
  "ReservedInstancesModifications": [
    {
      "Status": "fulfilled",
      "ModificationResults": [
        {
          "ReservedInstancesId": "93bbbca2-62f1-4d9d-b225-16bada29e6c7",
          "TargetConfiguration": {
            "AvailabilityZone": "us-east-1b",
            "InstanceType": "m1.large",
            "InstanceCount": 3
          }
        },
        {
          "ReservedInstancesId": "1ba8e2e3-aabb-46c3-bcf5-3fe2fda922e6",
          "TargetConfiguration": {
            "AvailabilityZone": "us-east-1d",
            "InstanceType": "m1.xlarge",
            "InstanceCount": 1
          }
        }
      ]
    },
    {
      "EffectiveDate": "2015-08-12T17:00:00.000Z",
      "CreateDate": "2015-08-12T17:52:52.630Z",
      "UpdateDate": "2015-08-12T18:08:06.698Z",
    }
  ]
}
```

```
    "ClientToken": "c9adb218-3222-4889-8216-0cf0e52dc37e":
    "ReservedInstancesModificationId": "rimod-d3ed4335-b1d3-4de6-
ab31-0f13aaf46687",
    "ReservedInstancesIds": [
        {
            "ReservedInstancesId": "b847fa93-e282-4f55-b59a-1342f5bd7c02"
        }
    ]
}
]
```

- Para API obtener más información, consulte [DescribeReservedInstancesModifications](#) la Referencia de AWS CLI comandos.

## describe-reserved-instances-offerings

En el siguiente ejemplo de código, se muestra cómo usar `describe-reserved-instances-offerings`.

### AWS CLI

Para describir las ofertas de instancias reservadas

Este comando de ejemplo describe todas las instancias reservadas que se pueden comprar en la región.

Comando:

```
aws ec2 describe-reserved-instances-offerings
```

Salida:

```
{
  "ReservedInstancesOfferings": [
    {
      "OfferingType": "Partial Upfront",
      "AvailabilityZone": "us-east-1b",
      "InstanceTenancy": "default",
      "PricingDetails": [],
      "ProductDescription": "Red Hat Enterprise Linux",
      "UsagePrice": 0.0,
```

```
    "RecurringCharges": [
      {
        "Amount": 0.088,
        "Frequency": "Hourly"
      }
    ],
    "Marketplace": false,
    "CurrencyCode": "USD",
    "FixedPrice": 631.0,
    "Duration": 94608000,
    "ReservedInstancesOfferingId": "9a06095a-bdc6-47fe-a94a-2a382f016040",
    "InstanceType": "c1.medium"
  },
  {
    "OfferingType": "PartialUpfront",
    "AvailabilityZone": "us-east-1b",
    "InstanceTenancy": "default",
    "PricingDetails": [],
    "ProductDescription": "Linux/UNIX",
    "UsagePrice": 0.0,
    "RecurringCharges": [
      {
        "Amount": 0.028,
        "Frequency": "Hourly"
      }
    ],
    "Marketplace": false,
    "CurrencyCode": "USD",
    "FixedPrice": 631.0,
    "Duration": 94608000,
    "ReservedInstancesOfferingId": "bfbefc6c-0d10-418d-b144-7258578d329d",
    "InstanceType": "c1.medium"
  },
  ...
}
```

Para describir sus ofertas de instancias reservadas mediante las opciones

En este ejemplo, se enumeran las instancias reservadas que se ofrecen AWS con las siguientes especificaciones: tipos de instancias t1.micro, productos de Windows VPC (Amazon) y ofertas de uso intensivo.

Comando:



```
aws ec2 describe-reserved-instances-offerings --no-include-marketplace --instance-type "t1.micro" --product-description "Windows (Amazon VPC)" --offering-type "no upfront"
```

Salida:

```
{
  "ReservedInstancesOfferings": [
    {
      "OfferingType": "No Upfront",
      "AvailabilityZone": "us-east-1b",
      "InstanceTenancy": "default",
      "PricingDetails": [],
      "ProductDescription": "Windows",
      "UsagePrice": 0.0,
      "RecurringCharges": [
        {
          "Amount": 0.015,
          "Frequency": "Hourly"
        }
      ],
      "Marketplace": false,
      "CurrencyCode": "USD",
      "FixedPrice": 0.0,
      "Duration": 31536000,
      "ReservedInstancesOfferingId": "c48ab04c-fe69-4f94-8e39-a23842292823",
      "InstanceType": "t1.micro"
    },
    ...
    {
      "OfferingType": "No Upfront",
      "AvailabilityZone": "us-east-1d",
      "InstanceTenancy": "default",
      "PricingDetails": [],
      "ProductDescription": "Windows (Amazon VPC)",
      "UsagePrice": 0.0,
      "RecurringCharges": [
        {
          "Amount": 0.015,
          "Frequency": "Hourly"
        }
      ],
    },
  ],
}
```

```
        "Marketplace": false,
        "CurrencyCode": "USD",
        "FixedPrice": 0.0,
        "Duration": 31536000,
        "ReservedInstancesOfferingId": "3a98bf7d-2123-42d4-b4f5-8dbec4b06dc6",
        "InstanceType": "t1.micro"
    }
]
}
```

- Para API obtener más información, consulte la Referencia [DescribeReservedInstancesOfferings](#) de AWS CLI comandos.

## describe-reserved-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-reserved-instances`.

### AWS CLI

Para describir sus instancias reservadas

Este comando de ejemplo describe las instancias reservadas que posee.

Comando:

```
aws ec2 describe-reserved-instances
```

Salida:

```
{
  "ReservedInstances": [
    {
      "ReservedInstancesId": "b847fa93-e282-4f55-b59a-1342fexample",
      "OfferingType": "No Upfront",
      "AvailabilityZone": "us-west-1c",
      "End": "2016-08-14T21:34:34.000Z",
      "ProductDescription": "Linux/UNIX",
      "UsagePrice": 0.00,
      "RecurringCharges": [
        {
          "Amount": 0.104,
          "Frequency": "Hourly"
        }
      ]
    }
  ]
}
```

```

    ],
    "Start": "2015-08-15T21:34:35.086Z",
    "State": "active",
    "FixedPrice": 0.0,
    "CurrencyCode": "USD",
    "Duration": 31536000,
    "InstanceTenancy": "default",
    "InstanceType": "m3.medium",
    "InstanceCount": 2
  },
  ...
]
}

```

Para describir sus instancias reservadas mediante filtros

En este ejemplo, se filtra la respuesta para incluir solo las instancias UNIX reservadas de Linux/t2.micro de tres años en us-west-1c.

Comando:

```

aws ec2 describe-reserved-instances --
filters Name=duration,Values=94608000 Name=instance-
type,Values=t2.micro Name=product-description,Values=Linux/UNIX Name=availability-
zone,Values=us-east-1e

```

Salida:

```

{
  "ReservedInstances": [
    {
      "ReservedInstancesId": "f127bd27-edb7-44c9-a0eb-0d7e09259af0",
      "OfferingType": "All Upfront",
      "AvailabilityZone": "us-east-1e",
      "End": "2018-03-26T21:34:34.000Z",
      "ProductDescription": "Linux/UNIX",
      "UsagePrice": 0.00,
      "RecurringCharges": [],
      "Start": "2015-03-27T21:34:35.848Z",
      "State": "active",
      "FixedPrice": 151.0,
      "CurrencyCode": "USD",
      "Duration": 94608000,

```

```

        "InstanceTenancy": "default",
        "InstanceType": "t2.micro",
        "InstanceCount": 1
    }
]
}

```

Para obtener más información, consulte [Uso de Amazon EC2 Instances](#) en la Guía del usuario de la interfaz de línea de AWS comandos.

- Para API obtener más información, consulte [DescribeReservedInstances](#) la Referencia de AWS CLI comandos.

## describe-route-tables

En el siguiente ejemplo de código, se muestra cómo usar `describe-route-tables`.

### AWS CLI

Para describir las tablas de rutas

En el siguiente `describe-route-tables` ejemplo, se recuperan los detalles de las tablas de rutas

```
aws ec2 describe-route-tables
```

Salida:

```

{
  "RouteTables": [
    {
      "Associations": [
        {
          "Main": true,
          "RouteTableAssociationId": "rtbassoc-0df3f54e06EXAMPLE",
          "RouteTableId": "rtb-09ba434c1bEXAMPLE"
        }
      ],
      "PropagatingVgws": [],
      "RouteTableId": "rtb-09ba434c1bEXAMPLE",
      "Routes": [
        {
          "DestinationCidrBlock": "10.0.0.0/16",

```

```
        "GatewayId": "local",
        "Origin": "CreateRouteTable",
        "State": "active"
    },
    {
        "DestinationCidrBlock": "0.0.0.0/0",
        "NatGatewayId": "nat-06c018cbd8EXAMPLE",
        "Origin": "CreateRoute",
        "State": "blackhole"
    }
],
"Tags": [],
"VpcId": "vpc-0065acced4EXAMPLE",
"OwnerId": "111122223333"
},
{
    "Associations": [
        {
            "Main": true,
            "RouteTableAssociationId": "rtbassoc-9EXAMPLE",
            "RouteTableId": "rtb-a1eec7de"
        }
    ],
    "PropagatingVgws": [],
    "RouteTableId": "rtb-a1eec7de",
    "Routes": [
        {
            "DestinationCidrBlock": "172.31.0.0/16",
            "GatewayId": "local",
            "Origin": "CreateRouteTable",
            "State": "active"
        },
        {
            "DestinationCidrBlock": "0.0.0.0/0",
            "GatewayId": "igw-fEXAMPLE",
            "Origin": "CreateRoute",
            "State": "active"
        }
    ],
    "Tags": [],
    "VpcId": "vpc-3EXAMPLE",
    "OwnerId": "111122223333"
},
{
```

```

    "Associations": [
      {
        "Main": false,
        "RouteTableAssociationId": "rtbassoc-0b100c28b2EXAMPLE",
        "RouteTableId": "rtb-07a98f76e5EXAMPLE",
        "SubnetId": "subnet-0d3d002af8EXAMPLE"
      }
    ],
    "PropagatingVgws": [],
    "RouteTableId": "rtb-07a98f76e5EXAMPLE",
    "Routes": [
      {
        "DestinationCidrBlock": "10.0.0.0/16",
        "GatewayId": "local",
        "Origin": "CreateRouteTable",
        "State": "active"
      },
      {
        "DestinationCidrBlock": "0.0.0.0/0",
        "GatewayId": "igw-06cf664d80EXAMPLE",
        "Origin": "CreateRoute",
        "State": "active"
      }
    ],
    "Tags": [],
    "VpcId": "vpc-0065acced4EXAMPLE",
    "OwnerId": "111122223333"
  }
]
}

```

Para obtener más información, consulte [Trabajar con tablas de rutas](#) en la Guía del AWS VPC usuario.

- Para API obtener más información, consulte [DescribeRouteTables](#) la Referencia de AWS CLI comandos.

### **describe-scheduled-instance-availability**

En el siguiente ejemplo de código, se muestra cómo usar `describe-scheduled-instance-availability`.

## AWS CLI

Para describir un horario disponible

En este ejemplo, se describe un horario que se realiza todos los domingos a partir de la fecha especificada.

Comando:

```
aws ec2 describe-scheduled-instance-availability --  
recurrence Frequency=Weekly,Interval=1,OccurrenceDays=[1] --first-slot-start-time-  
range EarliestTime=2016-01-31T00:00:00Z,LatestTime=2016-01-31T04:00:00Z
```

Salida:

```
{  
  "ScheduledInstanceAvailabilitySet": [  
    {  
      "AvailabilityZone": "us-west-2b",  
      "TotalScheduledInstanceHours": 1219,  
      "PurchaseToken": "eyJ2IjoiMSIsInMiOiJEsImMiOi...",  
      "MinTermDurationInDays": 366,  
      "AvailableInstanceCount": 20,  
      "Recurrence": {  
        "OccurrenceDaySet": [  
          1  
        ],  
        "Interval": 1,  
        "Frequency": "Weekly",  
        "OccurrenceRelativeToEnd": false  
      },  
      "Platform": "Linux/UNIX",  
      "FirstSlotStartTime": "2016-01-31T00:00:00Z",  
      "MaxTermDurationInDays": 366,  
      "SlotDurationInHours": 23,  
      "NetworkPlatform": "EC2-VPC",  
      "InstanceType": "c4.large",  
      "HourlyPrice": "0.095"  
    },  
    ...  
  ]  
}
```

Para limitar los resultados, puede agregar filtros que especifiquen el sistema operativo, la red y el tipo de instancia.

Comando:

```
--filters name=Plataforma, Values=Linux/ Name=Network-Platform, Values= - UNIX  
Name=instance-type, values=C4.large EC2 VPC
```

- Para API obtener más información, consulte [DescribeScheduledInstanceAvailability](#) la Referencia de AWS CLI comandos.

## describe-scheduled-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-scheduled-instances`.

AWS CLI

Para describir sus instancias programadas

En este ejemplo se describe la instancia programada especificada.

Comando:

```
aws ec2 describe-scheduled-instances --scheduled-instance-  
ids sci-1234-1234-1234-1234-123456789012
```

Salida:

```
{  
  "ScheduledInstanceSet": [  
    {  
      "AvailabilityZone": "us-west-2b",  
      "ScheduledInstanceId": "sci-1234-1234-1234-1234-123456789012",  
      "HourlyPrice": "0.095",  
      "CreateDate": "2016-01-25T21:43:38.612Z",  
      "Recurrence": {  
        "OccurrenceDaySet": [  
          1  
        ],  
        "Interval": 1,  
        "Frequency": "Weekly",  
        "OccurrenceRelativeToEnd": false,  
        "OccurrenceUnit": ""  
      }  
    }  
  ]  
}
```



```
    },
    "Platform": "Linux/UNIX",
    "TermEndDate": "2017-01-31T09:00:00Z",
    "InstanceCount": 1,
    "SlotDurationInHours": 32,
    "TermStartDate": "2016-01-31T09:00:00Z",
    "NetworkPlatform": "EC2-VPC",
    "TotalScheduledInstanceHours": 1696,
    "NextSlotStartTime": "2016-01-31T09:00:00Z",
    "InstanceType": "c4.large"
  }
]
}
```

En este ejemplo se describen todas las instancias programadas.

Comando:

```
aws ec2 describe-scheduled-instances
```

- Para API obtener más información, consulte [DescribeScheduledInstances](#) la Referencia de AWS CLI comandos.

## describe-security-group-references

En el siguiente ejemplo de código, se muestra cómo usar `describe-security-group-references`.

AWS CLI

Para describir las referencias a los grupos de seguridad

En este ejemplo se describen las referencias a los grupos de seguridad `desg-bbbb2222`. La respuesta indica que un grupo de seguridad `sg-bbbb2222` está haciendo referencia a ese grupo de seguridad en `VPCvpc-aaaaaaaa`.

Comando:

```
aws ec2 describe-security-group-references --group-id sg-bbbbb22222
```

Salida:

```
{
  "SecurityGroupsReferenceSet": [
    {
      "ReferencingVpcId": "vpc-aaaaaaaa ",
      "GroupId": "sg-bbbbb22222",
      "VpcPeeringConnectionId": "pcx-b04deed9"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeSecurityGroupReferences](#) la Referencia de AWS CLI comandos.

## describe-security-group-rules

En el siguiente ejemplo de código, se muestra cómo usar `describe-security-group-rules`.

### AWS CLI

Ejemplo 1: Para describir las reglas de un grupo de seguridad

En el siguiente `describe-security-group-rules` ejemplo, se describen las reglas del grupo de seguridad de un grupo de seguridad específico. Use la `filters` opción para limitar los resultados a un grupo de seguridad específico.

```
aws ec2 describe-security-group-rules \
  --filters Name="group-id",Values="sg-1234567890abcdef0"
```

Salida:

```
{
  "SecurityGroupRules": [
    {
      "SecurityGroupRuleId": "sgr-abcdef01234567890",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "111122223333",
      "IsEgress": false,
      "IpProtocol": "-1",
      "FromPort": -1,
      "ToPort": -1,
      "ReferencedGroupInfo": {
        "GroupId": "sg-1234567890abcdef0",
```

```

        "UserId": "111122223333"
      },
      "Tags": []
    },
    {
      "SecurityGroupId": "sg-1234567890abcdef0",
      "SecurityGroupName": "sg-1234567890abcdef0",
      "SecurityGroupRuleId": "sgr-bcdef01234567890a",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "111122223333",
      "IsEgress": true,
      "IpProtocol": "-1",
      "FromPort": -1,
      "ToPort": -1,
      "CidrIpv6": "::/0",
      "Tags": []
    },
    {
      "SecurityGroupId": "sg-1234567890abcdef0",
      "SecurityGroupName": "sg-1234567890abcdef0",
      "SecurityGroupRuleId": "sgr-cdef01234567890ab",
      "GroupId": "sg-1234567890abcdef0",
      "GroupOwnerId": "111122223333",
      "IsEgress": true,
      "IpProtocol": "-1",
      "FromPort": -1,
      "ToPort": -1,
      "CidrIpv4": "0.0.0.0/0",
      "Tags": []
    }
  ]
}

```

Ejemplo 2: Para describir una regla de grupo de seguridad

En el siguiente describe-security-group-rules ejemplo, se describe la regla de grupo de seguridad especificada.

```

aws ec2 describe-security-group-rules \
  --security-group-rule-ids sgr-cdef01234567890ab

```

Salida:

```

{
  "SecurityGroupRules": [
    {
      "SecurityGroupRuleId": "sgr-cdef01234567890ab",

```

```

    "GroupId": "sg-1234567890abcdef0",
    "GroupOwnerId": "111122223333",
    "IsEgress": true,
    "IpProtocol": "-1",
    "FromPort": -1,
    "ToPort": -1,
    "CidrIpv4": "0.0.0.0/0",
    "Tags": []
  }
]
}

```

Para obtener más información, consulta [las reglas de los grupos de seguridad](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DescribeSecurityGroupRules](#) la Referencia de AWS CLI comandos.

## describe-security-groups

En el siguiente ejemplo de código, se muestra cómo usar describe-security-groups.

### AWS CLI

Ejemplo 1: Describir un grupo de seguridad

En el siguiente ejemplo de describe-security-groups, se describe el grupo de seguridad especificado.

```

aws ec2 describe-security-groups \
  --group-ids sg-903004f8

```

Salida:

```

{
  "SecurityGroups": [
    {
      "IpPermissionsEgress": [
        {
          "IpProtocol": "-1",
          "IpRanges": [
            {
              "CidrIp": "0.0.0.0/0"
            }
          ]
        }
      ]
    }
  ]
}

```

```
        }
      ],
      "UserIdGroupPairs": [],
      "PrefixListIds": []
    }
  ],
  "Description": "My security group",
  "Tags": [
    {
      "Value": "SG1",
      "Key": "Name"
    }
  ],
  "IpPermissions": [
    {
      "IpProtocol": "-1",
      "IpRanges": [],
      "UserIdGroupPairs": [
        {
          "UserId": "123456789012",
          "GroupId": "sg-903004f8"
        }
      ],
      "PrefixListIds": []
    },
    {
      "PrefixListIds": [],
      "FromPort": 22,
      "IpRanges": [
        {
          "Description": "Access from NY office",
          "CidrIp": "203.0.113.0/24"
        }
      ],
      "ToPort": 22,
      "IpProtocol": "tcp",
      "UserIdGroupPairs": []
    }
  ],
  "GroupName": "MySecurityGroup",
  "VpcId": "vpc-1a2b3c4d",
  "OwnerId": "123456789012",
  "GroupId": "sg-903004f8",
}
```

```
]
}
```

## Ejemplo 2: Describir los grupos de seguridad que tienen reglas específicas

En el siguiente `describe-security-groups` ejemplo, se utilizan filtros para asignar los resultados a los grupos de seguridad que tienen una regla que permite el SSH tráfico (puerto 22) y otra que permite el tráfico desde todas las direcciones (`0.0.0.0/0`). En el ejemplo se usa el parámetro `--query` para mostrar solo los nombres de los grupos de seguridad. Los grupos de seguridad deben coincidir con todos los filtros para que se devuelvan en los resultados; sin embargo, una sola regla no tiene que coincidir con todos los filtros. Por ejemplo, el resultado devuelve un grupo de seguridad con una regla que permite el SSH tráfico desde una dirección IP específica y otra regla que permite el HTTP tráfico desde todas las direcciones.

```
aws ec2 describe-security-groups \
  --filters Name=ip-permission.from-port,Values=22 Name=ip-permission.to-
port,Values=22 Name=ip-permission.cidr,Values='0.0.0.0/0' \
  --query "SecurityGroups[*].[GroupName]" \
  --output text
```

Salida:

```
default
my-security-group
web-servers
launch-wizard-1
```

## Ejemplo 3: Describir los grupos de seguridad con base en las etiquetas

En el siguiente ejemplo de `describe-security-groups`, se usan filtros para limitar los resultados a los grupos de seguridad que incluyen `test` en el nombre del grupo de seguridad y que tienen la etiqueta `Test=To-delete`. En el ejemplo, se utiliza el `--query` parámetro para mostrar solo los nombres y los grupos IDs de seguridad.

```
aws ec2 describe-security-groups \
  --filters Name=group-name,Values=*test* Name=tag:Test,Values=To-delete \
  --query "SecurityGroups[*].{Name:GroupName, ID:GroupId}"
```

Salida:

```
[
  {
    "Name": "testfornewinstance",
    "ID": "sg-33bb22aa"
  },
  {
    "Name": "newgroupptest",
    "ID": "sg-1a2b3c4d"
  }
]
```

Para ver más ejemplos de uso de filtros de etiquetas, consulta [Cómo trabajar con etiquetas](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeSecurityGroups](#) la Referencia de AWS CLI comandos.

## describe-snapshot-attribute

En el siguiente ejemplo de código, se muestra cómo usar `describe-snapshot-attribute`.

### AWS CLI

Para describir los atributos de una instantánea

En el siguiente `describe-snapshot-attribute` ejemplo, se enumeran las cuentas con las que se comparte una instantánea.

```
aws ec2 describe-snapshot-attribute \
  --snapshot-id snap-01234567890abcdef \
  --attribute createVolumePermission
```

Salida:

```
{
  "SnapshotId": "snap-01234567890abcdef",
  "CreateVolumePermissions": [
    {
      "UserId": "123456789012"
    }
  ]
}
```

```
}
```

Para obtener más información, consulta [Compartir una EBS instantánea de Amazon](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte [DescribeSnapshotAttribute](#) la Referencia de AWS CLI comandos.

## describe-snapshot-tier-status

En el siguiente ejemplo de código, se muestra cómo usar `describe-snapshot-tier-status`.

### AWS CLI

Para ver la información archivada sobre una instantánea archivada

El siguiente `describe-snapshot-tier-status` ejemplo proporciona información de archivo sobre una instantánea archivada.

```
aws ec2 describe-snapshot-tier-status \
  --filters "Name=snapshot-id, Values=snap-01234567890abcdef"
```

Salida:

```
{
  "SnapshotTierStatuses": [
    {
      "Status": "completed",
      "ArchivalCompleteTime": "2021-09-15T17:33:16.147Z",
      "LastTieringProgress": 100,
      "Tags": [],
      "VolumeId": "vol-01234567890abcdef",
      "LastTieringOperationState": "archival-completed",
      "StorageTier": "archive",
      "OwnerId": "123456789012",
      "SnapshotId": "snap-01234567890abcdef",
      "LastTieringStartTime": "2021-09-15T16:44:37.574Z"
    }
  ]
}
```



Para obtener más información, consulte [Ver instantáneas archivadas](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte la Referencia [DescribeSnapshotTierStatus](#) de AWS CLI comandos.

## describe-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `describe-snapshots`.

### AWS CLI

Ejemplo 1: Describir una instantánea

En el siguiente ejemplo de `describe-snapshots`, se describe la instantánea especificada.

```
aws ec2 describe-snapshots \  
  --snapshot-ids snap-1234567890abcdef0
```

Salida:

```
{  
  "Snapshots": [  
    {  
      "Description": "This is my snapshot",  
      "Encrypted": false,  
      "VolumeId": "vol-049df61146c4d7901",  
      "State": "completed",  
      "VolumeSize": 8,  
      "StartTime": "2019-02-28T21:28:32.000Z",  
      "Progress": "100%",  
      "OwnerId": "012345678910",  
      "SnapshotId": "snap-01234567890abcdef",  
      "Tags": [  
        {  
          "Key": "Stack",  
          "Value": "test"  
        }  
      ]  
    }  
  ]  
}
```

Para obtener más información, consulta las [EBS instantáneas de Amazon](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Describir las instantáneas con base en filtros

En el siguiente `describe-snapshots` ejemplo, se utilizan filtros para limitar los resultados a las instantáneas que son propiedad de su AWS cuenta y que se encuentran en el `pending` estado. En el ejemplo, se usa el `--query` parámetro para mostrar solo la instantánea IDs y la hora en que se inició.

```
aws ec2 describe-snapshots \  
  --owner-ids self \  
  --filters Name=status,Values=pending \  
  --query "Snapshots[*].{ID:SnapshotId,Time:StartTime}"
```

Salida:

```
[  
  {  
    "ID": "snap-1234567890abcdef0",  
    "Time": "2019-08-04T12:48:18.000Z"  
  },  
  {  
    "ID": "snap-066877671789bd71b",  
    "Time": "2019-08-04T02:45:16.000Z"  
  },  
  ...  
]
```

En el siguiente ejemplo de `describe-snapshots`, se usan filtros para limitar los resultados a las instantáneas creadas a partir del volumen especificado. En el ejemplo, se usa el `--query` parámetro para mostrar solo la instantánea IDs.

```
aws ec2 describe-snapshots \  
  --filters Name=volume-id,Values=049df61146c4d7901 \  
  --query "Snapshots[*].[SnapshotId]" \  
  --output text
```

Salida:

```
snap-1234567890abcdef0
```

```
snap-08637175a712c3fb9
...
```

Para ver ejemplos adicionales de uso de filtros, consulta [Cómo publicar y filtrar tus recursos](#) en la Guía del EC2 usuario de Amazon.

### Ejemplo 3: Describir las instantáneas con base en etiquetas

En el siguiente ejemplo de `describe-snapshots`, se usan filtros de etiquetas para limitar los resultados a las instantáneas que tienen la etiqueta `Stack=Prod`.

```
aws ec2 describe-snapshots \
  --filters Name=tag:Stack,Values=prod
```

Para ver un ejemplo del resultado de `describe-snapshots`, consulte el ejemplo 1.

Para ver más ejemplos de uso de filtros de etiquetas, consulta [Cómo trabajar con etiquetas](#) en la Guía del EC2 usuario de Amazon.

### Ejemplo 4: Describir las instantáneas con base en la antigüedad

En el siguiente `describe-snapshots` ejemplo, se utilizan JMESPath expresiones para describir todas las instantáneas creadas por tu AWS cuenta antes de la fecha especificada. Muestra solo la instantáneaIDs.

```
aws ec2 describe-snapshots \
  --owner-ids 012345678910 \
  --query "Snapshots[?(StartTime<='2020-03-31')].[SnapshotId]"
```

Para ver ejemplos adicionales de uso de filtros, consulta [Cómo publicar y filtrar tus recursos](#) en la Guía del EC2 usuario de Amazon.

### Ejemplo 5: Ver solo las instantáneas archivadas

En el siguiente ejemplo de `describe-snapshots`, se muestran solo las instantáneas que se almacenan en el nivel de archivo.

```
aws ec2 describe-snapshots \
  --filters "Name=storage-tier,Values=archive"
```

Salida:

```
{
  "Snapshots": [
    {
      "Description": "Snap A",
      "Encrypted": false,
      "VolumeId": "vol-01234567890aaaaaa",
      "State": "completed",
      "VolumeSize": 8,
      "StartTime": "2021-09-07T21:00:00.000Z",
      "Progress": "100%",
      "OwnerId": "123456789012",
      "SnapshotId": "snap-01234567890aaaaaa",
      "StorageTier": "archive",
      "Tags": []
    },
  ]
}
```

Para obtener más información, consulte [Ver instantáneas archivadas](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte [DescribeSnapshots](#) la Referencia de AWS CLI comandos.

## **describe-spot-datafeed-subscription**

En el siguiente ejemplo de código, se muestra cómo usar `describe-spot-datafeed-subscription`.

AWS CLI

Para describir la suscripción a la fuente de datos de Spot Instance para una cuenta

Este comando de ejemplo describe la fuente de datos de la cuenta.

Comando:

```
aws ec2 describe-spot-datafeed-subscription
```

Salida:

```
{
  "SpotDatafeedSubscription": {
    "OwnerId": "123456789012",
    "Prefix": "spotdata",
    "Bucket": "my-s3-bucket",
    "State": "Active"
  }
}
```

- Para API obtener más información, consulte [DescribeSpotDatafeedSubscription](#) la Referencia de AWS CLI comandos.

## describe-spot-fleet-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-spot-fleet-instances`.

### AWS CLI

Para describir las instancias puntuales asociadas a una flota puntual

Este comando de ejemplo muestra las instancias de Spot asociadas a la flota de Spot especificada.

Comando:

```
aws ec2 describe-spot-fleet-instances --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Salida:

```
{
  "ActiveInstances": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "InstanceType": "m3.medium",
      "SpotInstanceRequestId": "sir-08b93456"
    },
    ...
  ],
  "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
}
```

```
}
```

- Para API obtener más información, consulte [DescribeSpotFleetInstances](#) la Referencia de AWS CLI comandos.

## describe-spot-fleet-request-history

En el siguiente ejemplo de código, se muestra cómo usar `describe-spot-fleet-request-history`.

### AWS CLI

Para describir el historial de la flota de Spot

Este comando de ejemplo devuelve el historial de la flota de Spot especificada a partir de la hora especificada.

Comando:

```
aws ec2 describe-spot-fleet-request-history --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE --start-time 2015-05-26T00:00:00Z
```

En el siguiente ejemplo, se muestran los lanzamientos satisfactorios de dos instancias de spot para la flota de Spot.

Salida:

```
{
  "HistoryRecords": [
    {
      "Timestamp": "2015-05-26T23:17:20.697Z",
      "EventInformation": {
        "EventSubType": "submitted"
      },
      "EventType": "fleetRequestChange"
    },
    {
      "Timestamp": "2015-05-26T23:17:20.873Z",
      "EventInformation": {
        "EventSubType": "active"
      },
    },
  ],
}
```

```

    "EventType": "fleetRequestChange"
  },
  {
    "Timestamp": "2015-05-26T23:21:21.712Z",
    "EventInformation": {
      "InstanceId": "i-1234567890abcdef0",
      "EventSubType": "launched"
    },
    "EventType": "instanceChange"
  },
  {
    "Timestamp": "2015-05-26T23:21:21.816Z",
    "EventInformation": {
      "InstanceId": "i-1234567890abcdef1",
      "EventSubType": "launched"
    },
    "EventType": "instanceChange"
  }
],
"SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
"NextToken": "CpHNsscimcV5oH7bSbub03CI2Qms5+ypNpNm
+53MNLr0YcXAkp0xFlfKf91yVxSExmbtma3awYxMFzNA663ZskT0AhtJ6TCb2Z8bQC2EnZgyELbymtWPfpZ1ZbauVg
+P+TfG1WxWWB/Vr5dk5d4LfdgA/DRAHUrYgxzrEXAMPLE=",
"StartTime": "2015-05-26T00:00:00Z"
}

```

- Para API obtener más información, consulte [DescribeSpotFleetRequestHistory](#) la Referencia de AWS CLI comandos.

## describe-spot-fleet-requests

En el siguiente ejemplo de código, se muestra cómo usar `describe-spot-fleet-requests`.

### AWS CLI

Para describir sus solicitudes de flota de Spot

En este ejemplo se describen todas sus solicitudes de flota de Spot.

Comando:

```
aws ec2 describe-spot-fleet-requests
```

**Salida:**

```
{
  "SpotFleetRequestConfigs": [
    {
      "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
      "SpotFleetRequestConfig": {
        "TargetCapacity": 20,
        "LaunchSpecifications": [
          {
            "EbsOptimized": false,
            "NetworkInterfaces": [
              {
                "SubnetId": "subnet-a61dafcf",
                "DeviceIndex": 0,
                "DeleteOnTermination": false,
                "AssociatePublicIpAddress": true,
                "SecondaryPrivateIpAddressCount": 0
              }
            ],
            "InstanceType": "cc2.8xlarge",
            "ImageId": "ami-1a2b3c4d"
          },
          {
            "EbsOptimized": false,
            "NetworkInterfaces": [
              {
                "SubnetId": "subnet-a61dafcf",
                "DeviceIndex": 0,
                "DeleteOnTermination": false,
                "AssociatePublicIpAddress": true,
                "SecondaryPrivateIpAddressCount": 0
              }
            ],
            "InstanceType": "r3.8xlarge",
            "ImageId": "ami-1a2b3c4d"
          }
        ],
        "SpotPrice": "0.05",
        "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role"
      },
      "SpotFleetRequestState": "active"
    },
    {
```



```

    "SpotFleetRequestId": "sfr-306341ed-9739-402e-881b-ce47bEXAMPLE",
    "SpotFleetRequestConfig": {
      "TargetCapacity": 20,
      "LaunchSpecifications": [
        {
          "EbsOptimized": false,
          "NetworkInterfaces": [
            {
              "SubnetId": "subnet-6e7f829e",
              "DeviceIndex": 0,
              "DeleteOnTermination": false,
              "AssociatePublicIpAddress": true,
              "SecondaryPrivateIpAddressCount": 0
            }
          ],
          "InstanceType": "m3.medium",
          "ImageId": "ami-1a2b3c4d"
        }
      ],
      "SpotPrice": "0.05",
      "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role"
    },
    "SpotFleetRequestState": "active"
  }
]
}

```

Para describir una solicitud de flota de Spot

En este ejemplo se describe la solicitud de flota de Spot especificada.

Comando:

```
aws ec2 describe-spot-fleet-requests --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Salida:

```

{
  "SpotFleetRequestConfigs": [
    {
      "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",

```

```
"SpotFleetRequestConfig": {
  "TargetCapacity": 20,
  "LaunchSpecifications": [
    {
      "EbsOptimized": false,
      "NetworkInterfaces": [
        {
          "SubnetId": "subnet-a61dafcf",
          "DeviceIndex": 0,
          "DeleteOnTermination": false,
          "AssociatePublicIpAddress": true,
          "SecondaryPrivateIpAddressCount": 0
        }
      ],
      "InstanceType": "cc2.8xlarge",
      "ImageId": "ami-1a2b3c4d"
    },
    {
      "EbsOptimized": false,
      "NetworkInterfaces": [
        {
          "SubnetId": "subnet-a61dafcf",
          "DeviceIndex": 0,
          "DeleteOnTermination": false,
          "AssociatePublicIpAddress": true,
          "SecondaryPrivateIpAddressCount": 0
        }
      ],
      "InstanceType": "r3.8xlarge",
      "ImageId": "ami-1a2b3c4d"
    }
  ],
  "SpotPrice": "0.05",
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role"
},
"SpotFleetRequestState": "active"
}
]
```

- Para API obtener más información, consulte [DescribeSpotFleetRequests](#) la Referencia de AWS CLI comandos.

## describe-spot-instance-requests

En el siguiente ejemplo de código, se muestra cómo usar `describe-spot-instance-requests`.

### AWS CLI

Ejemplo 1: Para describir una solicitud de instancia puntual

El siguiente `describe-spot-instance-requests` ejemplo describe la solicitud de instancia de spot especificada.

```
aws ec2 describe-spot-instance-requests \
  --spot-instance-request-ids sir-08b93456
```

Salida:

```
{
  "SpotInstanceRequests": [
    {
      "CreateTime": "2018-04-30T18:14:55.000Z",
      "InstanceId": "i-1234567890abcdef1",
      "LaunchSpecification": {
        "InstanceType": "t2.micro",
        "ImageId": "ami-003634241a8fcdec0",
        "KeyName": "my-key-pair",
        "SecurityGroups": [
          {
            "GroupName": "default",
            "GroupId": "sg-e38f24a7"
          }
        ],
        "BlockDeviceMappings": [
          {
            "DeviceName": "/dev/sda1",
            "Ebs": {
              "DeleteOnTermination": true,
              "SnapshotId": "snap-0e54a519c999adbbd",
              "VolumeSize": 8,
              "VolumeType": "standard",
              "Encrypted": false
            }
          }
        ]
      }
    }
  ],
}
```

```

    "NetworkInterfaces": [
      {
        "DeleteOnTermination": true,
        "DeviceIndex": 0,
        "SubnetId": "subnet-049df61146c4d7901"
      }
    ],
    "Placement": {
      "AvailabilityZone": "us-east-2b",
      "Tenancy": "default"
    },
    "Monitoring": {
      "Enabled": false
    }
  },
  "LaunchedAvailabilityZone": "us-east-2b",
  "ProductDescription": "Linux/UNIX",
  "SpotInstanceRequestId": "sir-08b93456",
  "SpotPrice": "0.010000",
  "State": "active",
  "Status": {
    "Code": "fulfilled",
    "Message": "Your Spot request is fulfilled.",
    "UpdateTime": "2018-04-30T18:16:21.000Z"
  },
  "Tags": [],
  "Type": "one-time",
  "InstanceInterruptionBehavior": "terminate"
}
]
}

```

Ejemplo 2: Para describir las solicitudes de instancias puntuales basadas en filtros

En el siguiente `describe-spot-instance-requests` ejemplo, se utilizan filtros para limitar los resultados a las solicitudes de instancias puntuales con el tipo de instancia especificado en la zona de disponibilidad especificada. En el ejemplo, se usa el `--query` parámetro para mostrar solo la `InstanceIDs`.

```

aws ec2 describe-spot-instance-requests \
  --filters Name=launch.instance-type,Values=m3.medium Name=launched-availability-
zone,Values=us-east-2a \
  --query "SpotInstanceRequests[*].[InstanceId]" \

```

```
--output text
```

Salida:

```
i-057750d42936e468a  
i-001efd250faaa6ffa  
i-027552a73f021f3bd  
...
```

Para ver ejemplos adicionales de uso de filtros, consulte [Cómo enumerar y filtrar los recursos](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

Ejemplo 3: Para describir las solicitudes de instancias puntuales basadas en etiquetas

En el siguiente `describe-spot-instance-requests` ejemplo, se utilizan filtros de etiquetas para limitar los resultados a las solicitudes de instancias puntuales que tienen la etiqueta `cost-center=cc123`.

```
aws ec2 describe-spot-instance-requests \  
  --filters Name=tag:cost-center,Values=cc123
```

Para ver un ejemplo del resultado de `describe-spot-instance-requests`, consulte el ejemplo 1.

Para ver más ejemplos de uso de filtros de etiquetas, consulta [Cómo trabajar con etiquetas](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeSpotInstanceRequests](#) la Referencia de AWS CLI comandos.

## **describe-spot-price-history**

En el siguiente ejemplo de código, se muestra cómo usar `describe-spot-price-history`.

AWS CLI

Para describir el historial de precios al contado

Este comando de ejemplo devuelve el historial de precios al contado de las instancias `m1.xlarge` de un día concreto de enero.

Comando:

```
aws ec2 describe-spot-price-history --instance-types m1.xlarge --start-time 2014-01-06T07:08:09 --end-time 2014-01-06T08:09:10
```

Salida:

```
{
  "SpotPriceHistory": [
    {
      "Timestamp": "2014-01-06T07:10:55.000Z",
      "ProductDescription": "SUSE Linux",
      "InstanceType": "m1.xlarge",
      "SpotPrice": "0.087000",
      "AvailabilityZone": "us-west-1b"
    },
    {
      "Timestamp": "2014-01-06T07:10:55.000Z",
      "ProductDescription": "SUSE Linux",
      "InstanceType": "m1.xlarge",
      "SpotPrice": "0.087000",
      "AvailabilityZone": "us-west-1c"
    },
    {
      "Timestamp": "2014-01-06T05:42:36.000Z",
      "ProductDescription": "SUSE Linux (Amazon VPC)",
      "InstanceType": "m1.xlarge",
      "SpotPrice": "0.087000",
      "AvailabilityZone": "us-west-1a"
    },
    ...
  ]
}
```

Para describir el historial de precios al contado para Linux/Amazon UNIX VPC

Este comando de ejemplo devuelve el historial de precios puntuales de las VPC instancias m1.xlarge de Linux/ UNIX Amazon de un día concreto de enero.

Comando:

```
aws ec2 describe-spot-price-history --instance-types m1.xlarge --product-description "Linux/UNIX (Amazon VPC)" --start-time 2014-01-06T07:08:09 --end-time 2014-01-06T08:09:10
```

Salida:

```
{
  "SpotPriceHistory": [
    {
      "Timestamp": "2014-01-06T04:32:53.000Z",
      "ProductDescription": "Linux/UNIX (Amazon VPC)",
      "InstanceType": "m1.xlarge",
      "SpotPrice": "0.080000",
      "AvailabilityZone": "us-west-1a"
    },
    {
      "Timestamp": "2014-01-05T11:28:26.000Z",
      "ProductDescription": "Linux/UNIX (Amazon VPC)",
      "InstanceType": "m1.xlarge",
      "SpotPrice": "0.080000",
      "AvailabilityZone": "us-west-1c"
    }
  ]
}
```

- Para obtener API más información, consulte la Referencia de comandos.

[DescribeSpotPriceHistory](#) AWS CLI

## describe-stale-security-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-stale-security-groups`.

AWS CLI

Para describir los grupos de seguridad obsoletos

En este ejemplo se describen las reglas anticuadas de los grupos de seguridad para.

`vpc-11223344` La respuesta muestra que el `sg-5fa68d3a` de su cuenta tiene una regla de entrada obsoleta que hace referencia `sg-279ab042` en el par VPC y que en su cuenta tiene una SSH regla de salida `sg-fe6fba9a` obsoleta que hace referencia en el par. SSH `sg-ef6fba8b` VPC

Comando:

```
aws ec2 describe-stale-security-groups --vpc-id vpc-11223344
```

## Salida:

```
{
  "StaleSecurityGroupSet": [
    {
      "VpcId": "vpc-11223344",
      "StaleIpPermissionsEgress": [
        {
          "ToPort": 22,
          "FromPort": 22,
          "UserIdGroupPairs": [
            {
              "VpcId": "vpc-7a20e51f",
              "GroupId": "sg-ef6fba8b",
              "VpcPeeringConnectionId": "pcx-b04deed9",
              "PeeringStatus": "active"
            }
          ],
          "IpProtocol": "tcp"
        }
      ],
      "GroupName": "MySG1",
      "StaleIpPermissions": [],
      "GroupId": "sg-fe6fba9a",
      "Description": "MySG1"
    },
    {
      "VpcId": "vpc-11223344",
      "StaleIpPermissionsEgress": [],
      "GroupName": "MySG2",
      "StaleIpPermissions": [
        {
          "ToPort": 22,
          "FromPort": 22,
          "UserIdGroupPairs": [
            {
              "VpcId": "vpc-7a20e51f",
              "GroupId": "sg-279ab042",
              "Description": "Access from pcx-b04deed9",
              "VpcPeeringConnectionId": "pcx-b04deed9",
              "PeeringStatus": "active"
            }
          ],
          "IpProtocol": "tcp"
        }
      ]
    }
  ]
}
```



```

    }
  ],
  "GroupId": "sg-5fa68d3a",
  "Description": "MySG2"
}
]
}

```

- Para [DescribeStaleSecurityGroups](#) obtener AWS CLI más información, consulta la Referencia de comandos. API

## describe-store-image-tasks

En el siguiente ejemplo de código, se muestra cómo usar `describe-store-image-tasks`.

### AWS CLI

Para describir el progreso de una tarea de AMI tienda

El siguiente `describe-store-image-tasks` ejemplo describe el progreso de una tarea de AMI tienda.

```
aws ec2 describe-store-image-tasks
```

Salida:

```

{
  "AmiId": "ami-1234567890abcdef0",
  "Bucket": "my-ami-bucket",
  "ProgressPercentage": 17,
  "S3ObjectKey": "ami-1234567890abcdef0.bin",
  "StoreTaskState": "InProgress",
  "StoreTaskFailureReason": null,
  "TaskStartTime": "2022-01-01T01:01:01.001Z"
}

```

Para obtener más información sobre el almacenamiento y la restauración y el AMI uso de S3, consulte [Almacenar y restaurar un AMI uso de S3 <https://docs.aws.amazon.com/AWS-EC2/latest/UserGuide/ami-store-restore.html>](https://docs.aws.amazon.com/AWS-EC2/latest/UserGuide/ami-store-restore.html) en la Guía del usuario de Amazon. EC2

- Para obtener API más información, consulte la Referencia de comandos. [DescribeStoreImageTasks](#) AWS CLI

## describe-subnets

En el siguiente ejemplo de código, se muestra cómo usar `describe-subnets`.

### AWS CLI

Ejemplo 1: Describir todas las subredes

En el siguiente ejemplo de `describe-subnets`, se muestran los detalles de las subredes.

```
aws ec2 describe-subnets
```

Salida:

```
{
  "Subnets": [
    {
      "AvailabilityZone": "us-east-1d",
      "AvailabilityZoneId": "use1-az2",
      "AvailableIpAddressCount": 4089,
      "CidrBlock": "172.31.80.0/20",
      "DefaultForAz": true,
      "MapPublicIpOnLaunch": false,
      "MapCustomerOwnedIpOnLaunch": true,
      "State": "available",
      "SubnetId": "subnet-0bb1c79de3EXAMPLE",
      "VpcId": "vpc-0ee975135dEXAMPLE",
      "OwnerId": "111122223333",
      "AssignIpv6AddressOnCreation": false,
      "Ipv6CidrBlockAssociationSet": [],
      "CustomerOwnedIpv4Pool": "pool-2EXAMPLE",
      "SubnetArn": "arn:aws:ec2:us-east-2:111122223333:subnet/
subnet-0bb1c79de3EXAMPLE",
      "EnableDns64": false,
      "Ipv6Native": false,
      "PrivateDnsNameOptionsOnLaunch": {
        "HostnameType": "ip-name",
        "EnableResourceNameDnsARecord": false,
        "EnableResourceNameDnsAAAARecord": false
      }
    },
    {
      "AvailabilityZone": "us-east-1d",
```

```

    "AvailabilityZoneId": "use1-az2",
    "AvailableIpAddressCount": 4089,
    "CidrBlock": "172.31.80.0/20",
    "DefaultForAz": true,
    "MapPublicIpOnLaunch": true,
    "MapCustomerOwnedIpOnLaunch": false,
    "State": "available",
    "SubnetId": "subnet-8EXAMPLE",
    "VpcId": "vpc-3EXAMPLE",
    "OwnerId": "111122223333",
    "AssignIpv6AddressOnCreation": false,
    "Ipv6CidrBlockAssociationSet": [],
    "Tags": [
      {
        "Key": "Name",
        "Value": "MySubnet"
      }
    ],
    "SubnetArn": "arn:aws:ec2:us-east-1:111122223333:subnet/
subnet-8EXAMPLE",
    "EnableDns64": false,
    "Ipv6Native": false,
    "PrivateDnsNameOptionsOnLaunch": {
      "HostnameType": "ip-name",
      "EnableResourceNameDnsARecord": false,
      "EnableResourceNameDnsAAAARecord": false
    }
  }
]
}

```

Para obtener más información, consulte [Trabajo con subredes VPCs y subredes](#) en la Guía del AWS VPC usuario.

Ejemplo 2: Para describir las subredes de una red específica VPC

En el siguiente `describe-subnets` ejemplo, se utiliza un filtro para recuperar los detalles de las subredes de la unidad especificada. VPC

```

aws ec2 describe-subnets \
  --filters "Name=vpc-id,Values=vpc-3EXAMPLE"

```

Salida:

```

{
  "Subnets": [
    {
      "AvailabilityZone": "us-east-1d",
      "AvailabilityZoneId": "use1-az2",
      "AvailableIpAddressCount": 4089,
      "CidrBlock": "172.31.80.0/20",
      "DefaultForAz": true,
      "MapPublicIpOnLaunch": true,
      "MapCustomerOwnedIpOnLaunch": false,
      "State": "available",
      "SubnetId": "subnet-8EXAMPLE",
      "VpcId": "vpc-3EXAMPLE",
      "OwnerId": "1111222233333",
      "AssignIpv6AddressOnCreation": false,
      "Ipv6CidrBlockAssociationSet": [],
      "Tags": [
        {
          "Key": "Name",
          "Value": "MySubnet"
        }
      ],
      "SubnetArn": "arn:aws:ec2:us-east-1:111122223333:subnet/
subnet-8EXAMPLE",
      "EnableDns64": false,
      "Ipv6Native": false,
      "PrivateDnsNameOptionsOnLaunch": {
        "HostnameType": "ip-name",
        "EnableResourceNameDnsARecord": false,
        "EnableResourceNameDnsAAAARecord": false
      }
    }
  ]
}

```

Para obtener más información, consulte [Trabajar con subredes VPCs y subredes](#) en la Guía del AWS VPC usuario.

### Ejemplo 3: Describir las subredes con una etiqueta específica

En el siguiente `describe-subnets` ejemplo, se utiliza un filtro para recuperar los detalles de las subredes con la etiqueta `CostCenter=123` y el `--query` parámetro para mostrar la subred IDs de las subredes con esta etiqueta.

```
aws ec2 describe-subnets \  
  --filters "Name=tag:CostCenter,Values=123" \  
  --query "Subnets[*].SubnetId" \  
  --output text
```

Salida:

```
subnet-0987a87c8b37348ef  
subnet-02a95061c45f372ee  
subnet-03f720e7de2788d73
```

Para obtener más información, consulte [Trabajar con subredes VPCs y subredes](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DescribeSubnets](#) la Referencia de AWS CLI comandos.

## describe-tags

En el siguiente ejemplo de código, se muestra cómo usar `describe-tags`.

AWS CLI

Ejemplo 1: Para describir todas las etiquetas de un único recurso

En el siguiente `describe-tags` ejemplo, se describen las etiquetas de la instancia especificada.

```
aws ec2 describe-tags \  
  --filters "Name=resource-id,Values=i-1234567890abcdef8"
```

Salida:

```
{  
  "Tags": [  
    {  
      "ResourceType": "instance",  
      "ResourceId": "i-1234567890abcdef8",  
      "Value": "Test",  
      "Key": "Stack"  
    },  
    {
```

```
        "ResourceType": "instance",
        "ResourceId": "i-1234567890abcdef8",
        "Value": "Beta Server",
        "Key": "Name"
    }
]
}
```

Ejemplo 2: Para describir todas las etiquetas de un tipo de recurso

En el siguiente `describe-tags` ejemplo, se describen las etiquetas de los volúmenes.

```
aws ec2 describe-tags \
  --filters "Name=resource-type,Values=volume"
```

Salida:

```
{
  "Tags": [
    {
      "ResourceType": "volume",
      "ResourceId": "vol-1234567890abcdef0",
      "Value": "Project1",
      "Key": "Purpose"
    },
    {
      "ResourceType": "volume",
      "ResourceId": "vol-049df61146c4d7901",
      "Value": "Logs",
      "Key": "Purpose"
    }
  ]
}
```

Ejemplo 3: Para describir todas las etiquetas

En el siguiente `describe-tags` ejemplo, se describen las etiquetas de todos los recursos.

```
aws ec2 describe-tags
```

Ejemplo 4: Para describir las etiquetas de los recursos en función de una clave de etiqueta

En el siguiente `describe-tags` ejemplo, se describen las etiquetas de los recursos que tienen una etiqueta con la clave `Stack`.

```
aws ec2 describe-tags \  
  --filters Name=key,Values=Stack
```

Salida:

```
{  
  "Tags": [  
    {  
      "ResourceType": "volume",  
      "ResourceId": "vol-027552a73f021f3b",  
      "Value": "Production",  
      "Key": "Stack"  
    },  
    {  
      "ResourceType": "instance",  
      "ResourceId": "i-1234567890abcdef8",  
      "Value": "Test",  
      "Key": "Stack"  
    }  
  ]  
}
```

Ejemplo 5: Para describir las etiquetas de sus recursos en función de una clave de etiqueta y un valor de etiqueta

En el siguiente `describe-tags` ejemplo, se describen las etiquetas de los recursos que tienen la etiqueta `Stack=Test`.

```
aws ec2 describe-tags \  
  --filters Name=key,Values=Stack Name=value,Values=Test
```

Salida:

```
{  
  "Tags": [  
    {  
      "ResourceType": "image",
```

```

    "ResourceId": "ami-3ac336533f021f3bd",
    "Value": "Test",
    "Key": "Stack"
  },
  {
    "ResourceType": "instance",
    "ResourceId": "i-1234567890abcdef8",
    "Value": "Test",
    "Key": "Stack"
  }
]
}

```

En el siguiente `describe-tags` ejemplo, se utiliza una sintaxis alternativa para describir los recursos con la etiqueta `Stack=Test`.

```

aws ec2 describe-tags \
  --filters "Name=tag:Stack,Values=Test"

```

En el siguiente `describe-tags` ejemplo, se describen las etiquetas de todas las instancias que tienen una etiqueta con la clave `Purpose` y sin ningún valor.

```

aws ec2 describe-tags \
  --filters "Name=resource-
type,Values=instance" "Name=key,Values=Purpose" "Name=value,Values="

```

Salida:

```

{
  "Tags": [
    {
      "ResourceType": "instance",
      "ResourceId": "i-1234567890abcdef5",
      "Value": null,
      "Key": "Purpose"
    }
  ]
}

```

- Para API obtener más información, consulta [DescribeTags](#) la Referencia de AWS CLI comandos.



## describe-traffic-mirror-filters

En el siguiente ejemplo de código, se muestra cómo usar `describe-traffic-mirror-filters`.

### AWS CLI

Para ver los filtros espejo de tráfico

En el siguiente `describe-traffic-mirror-filters` ejemplo, se muestran los detalles de todos los filtros espejo de tráfico.

```
aws ec2 describe-traffic-mirror-filters
```

Salida:

```
{
  "TrafficMirrorFilters": [
    {
      "TrafficMirrorFilterId": "tmf-0293f26e86EXAMPLE",
      "IngressFilterRules": [
        {
          "TrafficMirrorFilterRuleId": "tmfr-0ca76e0e08EXAMPLE",
          "TrafficMirrorFilterId": "tmf-0293f26e86EXAMPLE",
          "TrafficDirection": "ingress",
          "RuleNumber": 100,
          "RuleAction": "accept",
          "Protocol": 6,
          "DestinationCidrBlock": "10.0.0.0/24",
          "SourceCidrBlock": "10.0.0.0/24",
          "Description": "TCP Rule"
        }
      ],
      "EgressFilterRules": [],
      "NetworkServices": [],
      "Description": "Example filter",
      "Tags": []
    }
  ]
}
```

Para obtener más información, [consulte Ver los filtros reflejados de tráfico](#) en la Guía de duplicación de tráfico.

- Para API obtener más información, consulte la Referencia [DescribeTrafficMirrorFilters](#) de AWS CLI comandos.

## describe-traffic-mirror-sessions

En el siguiente ejemplo de código, se muestra cómo usar `describe-traffic-mirror-sessions`.

### AWS CLI

Para describir una sesión de Traffic Mirror

En el siguiente `describe-traffic-mirror-sessions` ejemplo, se muestran los detalles de sus sesiones de Traffic Mirror.

```
aws ec2 describe-traffic-mirror-sessions
```

Salida:

```
{
  "TrafficMirrorSessions": [
    {
      "Tags": [],
      "VirtualNetworkId": 42,
      "OwnerId": "111122223333",
      "Description": "TCP Session",
      "NetworkInterfaceId": "eni-0a471a5cf3EXAMPLE",
      "TrafficMirrorTargetId": "tmt-0dabe9b0a6EXAMPLE",
      "TrafficMirrorFilterId": "tmf-083e18f985EXAMPLE",
      "PacketLength": 20,
      "SessionNumber": 1,
      "TrafficMirrorSessionId": "tms-0567a4c684EXAMPLE"
    },
    {
      "Tags": [
        {
          "Key": "Name",
          "Value": "tag test"
        }
      ],
      "VirtualNetworkId": 13314501,
      "OwnerId": "111122223333",
```

```

        "Description": "TCP Session",
        "NetworkInterfaceId": "eni-0a471a5cf3EXAMPLE",
        "TrafficMirrorTargetId": "tmt-03665551cbEXAMPLE",
        "TrafficMirrorFilterId": "tmf-06c787846cEXAMPLE",
        "SessionNumber": 2,
        "TrafficMirrorSessionId": "tms-0060101cf8EXAMPLE"
    }
]
}

```

Para obtener más información, consulte [Ver los detalles de la sesión de Traffic Mirroring](#) en la Guía de AWS Traffic Mirroring.

- Para API obtener más información, consulte la Referencia [DescribeTrafficMirrorSessions](#) de AWS CLI comandos.

## describe-traffic-mirror-targets

En el siguiente ejemplo de código, se muestra cómo usar `describe-traffic-mirror-targets`.

### AWS CLI

Para describir un objetivo de espejo de tráfico

En el siguiente `describe-traffic-mirror-targets` ejemplo, se muestra información sobre el objetivo de duplicación de tráfico especificado.

```

aws ec2 describe-traffic-mirror-targets \
  --traffic-mirror-target-ids tmt-0dabe9b0a6EXAMPLE

```

Salida:

```

{
  "TrafficMirrorTargets": [
    {
      "TrafficMirrorTargetId": "tmt-0dabe9b0a6EXAMPLE",
      "NetworkLoadBalancerArn": "arn:aws:elasticloadbalancing:us-east-1:111122223333:loadbalancer/net/NLB/7cdec873fEXAMPLE",
      "Type": "network-load-balancer",
      "Description": "Example Network Load Balancer target",
      "OwnerId": "111122223333",
      "Tags": []
    }
  ]
}

```

```
]
}
```

Para obtener más información, consulta los [objetivos de las réplicas de tráfico](#) en la Guía de duplicación de VPC tráfico de Amazon.

- Para API obtener más información, consulte la Referencia [DescribeTrafficMirrorTargets](#) de AWS CLI comandos.

## describe-transit-gateway-attachments

En el siguiente ejemplo de código, se muestra cómo usar `describe-transit-gateway-attachments`.

### AWS CLI

Para ver los archivos adjuntos de su pasarela de transporte

En el siguiente `describe-transit-gateway-attachments` ejemplo, se muestran los detalles de los archivos adjuntos de tu pasarela de transporte público.

```
aws ec2 describe-transit-gateway-attachments
```

Salida:

```
{
  "TransitGatewayAttachments": [
    {
      "TransitGatewayAttachmentId": "tgw-attach-01f8100bc7EXAMPLE",
      "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
      "TransitGatewayOwnerId": "123456789012",
      "ResourceOwnerId": "123456789012",
      "ResourceType": "vpc",
      "ResourceId": "vpc-3EXAMPLE",
      "State": "available",
      "Association": {
        "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
        "State": "associated"
      },
      "CreationTime": "2019-08-26T14:59:25.000Z",
      "Tags": [
        {
```

```
        "Key": "Name",
        "Value": "Example"
    }
]
},
{
    "TransitGatewayAttachmentId": "tgw-attach-0b5968d3b6EXAMPLE",
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
    "TransitGatewayOwnerId": "123456789012",
    "ResourceOwnerId": "123456789012",
    "ResourceType": "vpc",
    "ResourceId": "vpc-0065acced4EXAMPLE",
    "State": "available",
    "Association": {
        "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
        "State": "associated"
    },
    "CreationTime": "2019-08-07T17:03:07.000Z",
    "Tags": []
},
{
    "TransitGatewayAttachmentId": "tgw-attach-08e0bc912cEXAMPLE",
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
    "TransitGatewayOwnerId": "123456789012",
    "ResourceOwnerId": "123456789012",
    "ResourceType": "direct-connect-gateway",
    "ResourceId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
    "State": "available",
    "Association": {
        "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
        "State": "associated"
    },
    "CreationTime": "2019-08-14T20:27:44.000Z",
    "Tags": []
},
{
    "TransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",
    "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
    "TransitGatewayOwnerId": "123456789012",
    "ResourceOwnerId": "123456789012",
    "ResourceType": "direct-connect-gateway",
    "ResourceId": "8384da05-13ce-4a91-aada-5a1baEXAMPLE",
    "State": "available",
    "Association": {
```

```

        "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
        "State": "associated"
    },
    "CreationTime": "2019-08-14T20:33:02.000Z",
    "Tags": []
}
]
}

```

Para obtener más información, consulte Cómo [trabajar con pasarelas de tránsito](#) en la Guía de pasarelas de tránsito.

- Para API obtener más información, consulte la Referencia [DescribeTransitGatewayAttachments](#) de AWS CLI comandos.

## describe-transit-gateway-connect-peers

En el siguiente ejemplo de código, se muestra cómo usar `describe-transit-gateway-connect-peers`.

### AWS CLI

Para describir a un compañero de Transit Gateway Connect

En el siguiente `describe-transit-gateway-connect-peers` ejemplo, se describe el par Connect especificado.

```

aws ec2 describe-transit-gateway-connect-peers \
  --transit-gateway-connect-peer-ids tgw-connect-peer-0666adbac4EXAMPLE

```

Salida:

```

{
  "TransitGatewayConnectPeers": [
    {
      "TransitGatewayAttachmentId": "tgw-attach-0f0927767cEXAMPLE",
      "TransitGatewayConnectPeerId": "tgw-connect-peer-0666adbac4EXAMPLE",
      "State": "available",
      "CreationTime": "2021-10-13T03:35:17.000Z",
      "ConnectPeerConfiguration": {
        "TransitGatewayAddress": "10.0.0.234",
        "PeerAddress": "172.31.1.11",

```

```

    "InsideCidrBlocks": [
      "169.254.6.0/29"
    ],
    "Protocol": "gre",
    "BgpConfigurations": [
      {
        "TransitGatewayAsn": 64512,
        "PeerAsn": 64512,
        "TransitGatewayAddress": "169.254.6.2",
        "PeerAddress": "169.254.6.1",
        "BgpStatus": "down"
      },
      {
        "TransitGatewayAsn": 64512,
        "PeerAsn": 64512,
        "TransitGatewayAddress": "169.254.6.3",
        "PeerAddress": "169.254.6.1",
        "BgpStatus": "down"
      }
    ]
  },
  "Tags": []
}
]
}

```

Para obtener más información, consulte los [archivos adjuntos de Transit Gateway Connect y los pares de Transit Gateway Connect](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [DescribeTransitGatewayConnectPeers](#) la Referencia de AWS CLI comandos.

## describe-transit-gateway-connects

En el siguiente ejemplo de código, se muestra cómo usar `describe-transit-gateway-connects`.

### AWS CLI

Para describir un adjunto de Transit Gateway Connect

En el siguiente `describe-transit-gateway-connects` ejemplo, se describe el adjunto Connect especificado.

```
aws ec2 describe-transit-gateway-connects \
  --transit-gateway-attachment-ids tgw-attach-037012e5dcEXAMPLE
```

Salida:

```
{
  "TransitGatewayConnects": [
    {
      "TransitGatewayAttachmentId": "tgw-attach-037012e5dcEXAMPLE",
      "TransportTransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",
      "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
      "State": "available",
      "CreationTime": "2021-03-09T19:59:17+00:00",
      "Options": {
        "Protocol": "gre"
      },
      "Tags": []
    }
  ]
}
```

Para obtener más información, consulte los [archivos adjuntos de Transit Gateway Connect](#) y los [pares de Transit Gateway Connect](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [DescribeTransitGatewayConnects](#) la Referencia de AWS CLI comandos.

## describe-transit-gateway-multicast-domains

En el siguiente ejemplo de código, se muestra cómo usar `describe-transit-gateway-multicast-domains`.

AWS CLI

Para describir los dominios de multidifusión de Transit Gateway

En el siguiente `describe-transit-gateway-multicast-domains` ejemplo, se muestran los detalles de todos los dominios de multidifusión de Transit Gateway.

```
aws ec2 describe-transit-gateway-multicast-domains
```



**Salida:**

```
{
  "TransitGatewayMulticastDomains": [
    {
      "TransitGatewayMulticastDomainId": "tgw-mcast-domain-000fb24d04EXAMPLE",
      "TransitGatewayId": "tgw-0bf0bfffefaEXAMPLE",
      "TransitGatewayMulticastDomainArn": "arn:aws:ec2:us-east-1:123456789012:transit-gateway-multicast-domain/tgw-mcast-domain-000fb24d04EXAMPLE",
      "OwnerId": "123456789012",
      "Options": {
        "Icmpv2Support": "disable",
        "StaticSourcesSupport": "enable",
        "AutoAcceptSharedAssociations": "disable"
      },
      "State": "available",
      "CreationTime": "2019-12-10T18:32:50+00:00",
      "Tags": [
        {
          "Key": "Name",
          "Value": "mc1"
        }
      ]
    }
  ]
}
```

Para obtener más información, consulte [Administrar dominios de multidifusión](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [DescribeTransitGatewayMulticastDomains](#) de AWS CLI comandos.

**describe-transit-gateway-peering-attachments**

En el siguiente ejemplo de código, se muestra cómo usar `describe-transit-gateway-peering-attachments`.

**AWS CLI**

Para describir los archivos adjuntos de peering de Transit Gateway

En el siguiente `describe-transit-gateway-peering-attachments` ejemplo, se muestran los detalles de todos los archivos adjuntos de interconexión de Transit Gateway.

```
aws ec2 describe-transit-gateway-peering-attachments
```

Salida:

```
{
  "TransitGatewayPeeringAttachments": [
    {
      "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",
      "RequesterTgwInfo": {
        "TransitGatewayId": "tgw-123abc05e04123abc",
        "OwnerId": "123456789012",
        "Region": "us-west-2"
      },
      "AcceptorTgwInfo": {
        "TransitGatewayId": "tgw-11223344aabbcc112",
        "OwnerId": "123456789012",
        "Region": "us-east-2"
      },
      "State": "pendingAcceptance",
      "CreationTime": "2019-12-09T11:38:05.000Z",
      "Tags": []
    }
  ]
}
```

Para obtener más información, consulte los [adjuntos de peering de Transit Gateway](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [DescribeTransitGatewayPeeringAttachments](#) de AWS CLI comandos.

## `describe-transit-gateway-policy-tables`

En el siguiente ejemplo de código, se muestra cómo usar `describe-transit-gateway-policy-tables`.

AWS CLI

Para describir una tabla de políticas de pasarelas de tránsito

El siguiente `describe-transit-gateway-policy-tables` ejemplo describe la tabla de políticas de la pasarela de tránsito especificada.

```
aws ec2 describe-transit-gateway-policy-tables \
  --transit-gateway-policy-table-ids tgw-ptb-0a16f134b78668a81
```

Salida:

```
{
  "TransitGatewayPolicyTables": [
    {
      "TransitGatewayPolicyTableId": "tgw-ptb-0a16f134b78668a81",
      "TransitGatewayId": "tgw-067f8505c18f0bd6e",
      "State": "available",
      "CreationTime": "2023-11-28T16:36:43+00:00",
      "Tags": []
    }
  ]
}
```

Para obtener más información, consulte [las tablas de políticas de Transit Gateway](#) en la Guía del usuario de Transit Gateway.

- Para API obtener más información, consulte [DescribeTransitGatewayPolicyTables](#) la Referencia de AWS CLI comandos.

## `describe-transit-gateway-route-tables`

En el siguiente ejemplo de código, se muestra cómo usar `describe-transit-gateway-route-tables`.

AWS CLI

Para describir las tablas de rutas de su puerta de enlace de tránsito

En el siguiente `describe-transit-gateway-route-tables` ejemplo, se muestran los detalles de las tablas de rutas de la puerta de enlace de tránsito.

```
aws ec2 describe-transit-gateway-route-tables
```

Salida:

```
{
  "TransitGatewayRouteTables": [
    {
      "TransitGatewayRouteTableId": "tgw-rtb-0ca78a549EXAMPLE",
      "TransitGatewayId": "tgw-0bc994abffEXAMPLE",
      "State": "available",
      "DefaultAssociationRouteTable": true,
      "DefaultPropagationRouteTable": true,
      "CreationTime": "2018-11-28T14:24:49.000Z",
      "Tags": []
    },
    {
      "TransitGatewayRouteTableId": "tgw-rtb-0e8f48f148EXAMPLE",
      "TransitGatewayId": "tgw-0043d72bb4EXAMPLE",
      "State": "available",
      "DefaultAssociationRouteTable": true,
      "DefaultPropagationRouteTable": true,
      "CreationTime": "2018-11-28T14:24:00.000Z",
      "Tags": []
    }
  ]
}
```

Para obtener más información, consulta [Ver las tablas de rutas de Transit Gateways](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [DescribeTransitGatewayRouteTables](#) la Referencia de AWS CLI comandos.

## describe-transit-gateway-vpc-attachments

En el siguiente ejemplo de código, se muestra cómo usar `describe-transit-gateway-vpc-attachments`.

### AWS CLI

Para describir los VPC archivos adjuntos de su pasarela de transporte

En el siguiente `describe-transit-gateway-vpc-attachments` ejemplo, se muestran los detalles de los VPC archivos adjuntos de tu pasarela de transporte público.

```
aws ec2 describe-transit-gateway-vpc-attachments
```

Salida:

```
{
  "TransitGatewayVpcAttachments": [
    {
      "TransitGatewayAttachmentId": "tgw-attach-0a08e88308EXAMPLE",
      "TransitGatewayId": "tgw-0043d72bb4EXAMPLE",
      "VpcId": "vpc-0f501f7ee8EXAMPLE",
      "VpcOwnerId": "111122223333",
      "State": "available",
      "SubnetIds": [
        "subnet-045d586432EXAMPLE",
        "subnet-0a0ad478a6EXAMPLE"
      ],
      "CreationTime": "2019-02-13T11:04:02.000Z",
      "Options": {
        "DnsSupport": "enable",
        "Ipv6Support": "disable"
      },
      "Tags": [
        {
          "Key": "Name",
          "Value": "attachment name"
        }
      ]
    }
  ]
}
```

Para obtener más información, consulta [Ver tus VPC archivos adjuntos](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [DescribeTransitGatewayVpcAttachments](#) la Referencia de AWS CLI comandos.

## describe-transit-gateways

En el siguiente ejemplo de código, se muestra cómo usar `describe-transit-gateways`.

### AWS CLI

Para describir sus pasarelas de tránsito

El siguiente describe-transit-gateways ejemplo recupera detalles sobre tus pasarelas de tránsito.

```
aws ec2 describe-transit-gateways
```

Salida:

```
{
  "TransitGateways": [
    {
      "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
      "TransitGatewayArn": "arn:aws:ec2:us-east-2:111122223333:transit-
gateway/tgw-0262a0e521EXAMPLE",
      "State": "available",
      "OwnerId": "111122223333",
      "Description": "MyTGW",
      "CreationTime": "2019-07-10T14:02:12.000Z",
      "Options": {
        "AmazonSideAsn": 64516,
        "AutoAcceptSharedAttachments": "enable",
        "DefaultRouteTableAssociation": "enable",
        "AssociationDefaultRouteTableId": "tgw-rtb-018774adf3EXAMPLE",
        "DefaultRouteTablePropagation": "enable",
        "PropagationDefaultRouteTableId": "tgw-rtb-018774adf3EXAMPLE",
        "VpnEcmpSupport": "enable",
        "DnsSupport": "enable"
      },
      "Tags": []
    },
    {
      "TransitGatewayId": "tgw-0fb8421e2dEXAMPLE",
      "TransitGatewayArn": "arn:aws:ec2:us-east-2:111122223333:transit-
gateway/tgw-0fb8421e2da853bf3",
      "State": "available",
      "OwnerId": "111122223333",
      "CreationTime": "2019-03-15T22:57:33.000Z",
      "Options": {
        "AmazonSideAsn": 65412,
        "AutoAcceptSharedAttachments": "disable",
        "DefaultRouteTableAssociation": "enable",
        "AssociationDefaultRouteTableId": "tgw-rtb-06a241a3d8EXAMPLE",
        "DefaultRouteTablePropagation": "enable",
        "PropagationDefaultRouteTableId": "tgw-rtb-06a241a3d8EXAMPLE",
```

```

        "VpnEcmpSupport": "enable",
        "DnsSupport": "enable"
    },
    "Tags": [
        {
            "Key": "Name",
            "Value": "TGW1"
        }
    ]
}
]
}

```

- Para API obtener más información, consulte la Referencia [DescribeTransitGateways](#) de AWS CLI comandos.

## describe-verified-access-endpoints

En el siguiente ejemplo de código, se muestra cómo usar `describe-verified-access-endpoints`.

### AWS CLI

Para describir un punto final de acceso verificado

En el siguiente `delete-verified-access-endpoints` ejemplo, se describe el punto final de Verified Access especificado.

```

aws ec2 describe-verified-access-endpoints \
  --verified-access-endpoint-ids vae-066fac616d4d546f2

```

Salida:

```

{
  "VerifiedAccessEndpoints": [
    {
      "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
      "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
      "VerifiedAccessEndpointId": "vae-066fac616d4d546f2",
      "ApplicationDomain": "example.com",
      "EndpointType": "network-interface",
    }
  ]
}

```

```

        "AttachmentType": "vpc",
        "DomainCertificateArn": "arn:aws:acm:us-east-2:123456789012:certificate/
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE",
        "EndpointDomain": "my-ava-
app.edge-00c3372d53b1540bb.vai-0ce000c0b7643abea.prod.verified-access.us-
east-2.amazonaws.com",
        "SecurityGroupIds": [
            "sg-004915970c4c8f13a"
        ],
        "NetworkInterfaceOptions": {
            "NetworkInterfaceId": "eni-0aec70418c8d87a0f",
            "Protocol": "https",
            "Port": 443
        },
        "Status": {
            "Code": "active"
        },
        "Description": "",
        "CreationTime": "2023-08-25T20:54:43",
        "LastUpdatedTime": "2023-08-25T22:17:26",
        "Tags": [
            {
                "Key": "Name",
                "Value": "my-va-endpoint"
            }
        ]
    }
]
}

```

Para obtener más información, consulte los [puntos finales de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [DescribeVerifiedAccessEndpoints](#) la Referencia de AWS CLI comandos.

## describe-verified-access-groups

En el siguiente ejemplo de código, se muestra cómo usar describe-verified-access-groups.

### AWS CLI

Para describir un grupo de acceso verificado



El siguiente `describe-verified-access-groups` ejemplo describe el grupo de acceso verificado especificado.

```
aws ec2 describe-verified-access-groups \
  --verified-access-group-ids vagr-0dbe967baf14b7235
```

Salida:

```
{
  "VerifiedAccessGroups": [
    {
      "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
      "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
      "Description": "Testing Verified Access",
      "Owner": "123456789012",
      "VerifiedAccessGroupArn": "arn:aws:ec2:us-east-2:123456789012:verified-
access-group/vagr-0dbe967baf14b7235",
      "CreationTime": "2023-08-25T19:55:19",
      "LastUpdatedTime": "2023-08-25T22:17:25",
      "Tags": [
        {
          "Key": "Name",
          "Value": "my-va-group"
        }
      ]
    }
  ]
}
```

Para obtener más información, consulte los [grupos de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [DescribeVerifiedAccessGroups](#) la Referencia de AWS CLI comandos.

## **describe-verified-access-instance-logging-configurations**

En el siguiente ejemplo de código, se muestra cómo usar `describe-verified-access-instance-logging-configurations`.

## AWS CLI

Para describir la configuración de registro de una instancia de acceso verificado

En el siguiente `describe-verified-access-instance-logging-configurations` ejemplo, se describe la configuración de registro de la instancia de acceso verificado especificada.

```
aws ec2 describe-verified-access-instance-logging-configurations \
  --verified-access-instance-ids vai-0ce000c0b7643abea
```

Salida:

```
{
  "LoggingConfigurations": [
    {
      "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
      "AccessLogs": {
        "S3": {
          "Enabled": false
        },
        "CloudWatchLogs": {
          "Enabled": true,
          "DeliveryStatus": {
            "Code": "success"
          },
          "LogGroup": "my-log-group"
        },
        "KinesisDataFirehose": {
          "Enabled": false
        },
        "LogVersion": "ocsf-1.0.0-rc.2",
        "IncludeTrustContext": false
      }
    }
  ]
}
```

Para obtener más información, consulte [los registros de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [DescribeVerifiedAccessInstanceLoggingConfigurations](#) la Referencia de AWS CLI comandos.

## describe-verified-access-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-verified-access-instances`.

### AWS CLI

Para describir una instancia de acceso verificado

En el siguiente `describe-verified-access-instances` ejemplo, se describe la instancia de acceso verificado especificada.

```
aws ec2 describe-verified-access-instances \
  --verified-access-instance-ids vai-0ce000c0b7643abea
```

Salida:

```
{
  "VerifiedAccessInstances": [
    {
      "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
      "Description": "Testing Verified Access",
      "VerifiedAccessTrustProviders": [
        {
          "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
          "TrustProviderType": "user",
          "UserTrustProviderType": "iam-identity-center"
        }
      ],
      "CreationTime": "2023-08-25T18:27:56",
      "LastUpdatedTime": "2023-08-25T19:03:32",
      "Tags": [
        {
          "Key": "Name",
          "Value": "my-ava-instance"
        }
      ]
    }
  ]
}
```

Para obtener más información, consulte las [instancias de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [DescribeVerifiedAccessInstances](#) la Referencia de AWS CLI comandos.

## describe-verified-access-trust-providers

En el siguiente ejemplo de código, se muestra cómo usar `describe-verified-access-trust-providers`.

### AWS CLI

Para describir un proveedor de confianza de Verified Access

En el siguiente `describe-verified-access-trust-providers` ejemplo, se describe el proveedor de confianza de Verified Access especificado.

```
aws ec2 describe-verified-access-trust-providers \
  --verified-access-trust-provider-ids vatp-0bb32de759a3e19e7
```

Salida:

```
{
  "VerifiedAccessTrustProviders": [
    {
      "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
      "Description": "Testing Verified Access",
      "TrustProviderType": "user",
      "UserTrustProviderType": "iam-identity-center",
      "PolicyReferenceName": "idc",
      "CreationTime": "2023-08-25T19:00:38",
      "LastUpdatedTime": "2023-08-25T19:03:32",
      "Tags": [
        {
          "Key": "Name",
          "Value": "my-va-trust-provider"
        }
      ]
    }
  ]
}
```

Para obtener más información, consulte [Proveedores de confianza para el acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [DescribeVerifiedAccessTrustProviders](#) la Referencia de AWS CLI comandos.

## describe-volume-attribute

En el siguiente ejemplo de código, se muestra cómo usar `describe-volume-attribute`.

### AWS CLI

Para describir un atributo de volumen

Este comando de ejemplo describe el `autoEnableIo` atributo del volumen con el ID `vol-049df61146c4d7901`.

Comando:

```
aws ec2 describe-volume-attribute --volume-id vol-049df61146c4d7901 --  
attribute autoEnableIO
```

Salida:

```
{  
  "AutoEnableIO": {  
    "Value": false  
  },  
  "VolumeId": "vol-049df61146c4d7901"  
}
```

- Para API obtener más información, consulte [DescribeVolumeAttribute](#) la Referencia de AWS CLI comandos.

## describe-volume-status

En el siguiente ejemplo de código, se muestra cómo usar `describe-volume-status`.

### AWS CLI

Para describir el estado de un solo volumen

Este comando de ejemplo describe el estado del volumen `vol-1234567890abcdef0`.

Comando:

```
aws ec2 describe-volume-status --volume-ids vol-1234567890abcdef0
```

Salida:

```
{
  "VolumeStatuses": [
    {
      "VolumeStatus": {
        "Status": "ok",
        "Details": [
          {
            "Status": "passed",
            "Name": "io-enabled"
          },
          {
            "Status": "not-applicable",
            "Name": "io-performance"
          }
        ]
      },
      "AvailabilityZone": "us-east-1a",
      "VolumeId": "vol-1234567890abcdef0",
      "Actions": [],
      "Events": []
    }
  ]
}
```

Para describir el estado de los volúmenes deteriorados

Este comando de ejemplo describe el estado de todos los volúmenes deteriorados. En este ejemplo de salida, no hay volúmenes deteriorados.

Comando:

```
aws ec2 describe-volume-status --filters Name=volume-status.status,Values=impaired
```

Salida:

```
{
```

```
"VolumeStatuses": []
}
```

Si tienes un volumen cuya comprobación de estado ha fallado (el estado es defectuoso), consulta [Cómo trabajar con un volumen deteriorado](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeVolumeStatus](#) la Referencia de AWS CLI comandos.

## describe-volumes-modifications

En el siguiente ejemplo de código, se muestra cómo usar `describe-volumes-modifications`.

### AWS CLI

Para describir el estado de modificación de un volumen

En el siguiente `describe-volumes-modifications` ejemplo, se describe el estado de modificación del volumen especificado.

```
aws ec2 describe-volumes-modifications \
  --volume-ids vol-1234567890abcdef0
```

Salida:

```
{
  "VolumeModification": {
    "TargetSize": 150,
    "TargetVolumeType": "io1",
    "ModificationState": "optimizing",
    "VolumeId": "vol-1234567890abcdef0",
    "TargetIops": 100,
    "StartTime": "2019-05-17T11:27:19.000Z",
    "Progress": 70,
    "OriginalVolumeType": "io1",
    "OriginalIops": 100,
    "OriginalSize": 100
  }
}
```

- Para API obtener más información, consulte [DescribeVolumesModifications](#) la Referencia de AWS CLI comandos.

## describe-volumes

En el siguiente ejemplo de código, se muestra cómo usar `describe-volumes`.

### AWS CLI

Ejemplo 1: Para describir un volumen

El siguiente `describe-volumes` ejemplo describe los volúmenes especificados en la región actual.

```
aws ec2 describe-volumes \
  --volume-ids vol-049df61146c4d7901 vol-1234567890abcdef0
```

Salida:

```
{
  "Volumes": [
    {
      "AvailabilityZone": "us-east-1a",
      "Attachments": [
        {
          "AttachTime": "2013-12-18T22:35:00.000Z",
          "InstanceId": "i-1234567890abcdef0",
          "VolumeId": "vol-049df61146c4d7901",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
        }
      ],
      "Encrypted": true,
      "KmsKeyId": "arn:aws:kms:us-east-2a:123456789012:key/8c5b2c63-b9bc-45a3-a87a-5513eEXAMPLE",
      "VolumeType": "gp2",
      "VolumeId": "vol-049df61146c4d7901",
      "State": "in-use",
      "Iops": 100,
      "SnapshotId": "snap-1234567890abcdef0",
      "CreateTime": "2019-12-18T22:35:00.084Z",
      "Size": 8
    },
    {
      "AvailabilityZone": "us-east-1a",
```



```

    "Attachments": [],
    "Encrypted": false,
    "VolumeType": "gp2",
    "VolumeId": "vol-1234567890abcdef0",
    "State": "available",
    "Iops": 300,
    "SnapshotId": "",
    "CreateTime": "2020-02-27T00:02:41.791Z",
    "Size": 100
  }
]
}

```

Ejemplo 2: Para describir los volúmenes que están adjuntos a una instancia específica

En el siguiente `describe-volumes` ejemplo, se describen todos los volúmenes que están adjuntos a la instancia especificada y configurados para que se eliminen cuando la instancia finalice.

```

aws ec2 describe-volumes \
  --region us-east-1 \
  --filters Name=attachment.instance-
id,Values=i-1234567890abcdef0 Name=attachment.delete-on-termination,Values=true

```

Para ver un ejemplo del resultado de `describe-volumes`, consulte el ejemplo 1.

Ejemplo 3: Para describir los volúmenes disponibles en una zona de disponibilidad específica

El siguiente `describe-volumes` ejemplo describe todos los volúmenes que tienen un estado igual o se encuentran en la zona de disponibilidad especificada. `available`

```

aws ec2 describe-volumes \
  --filters Name=status,Values=available Name=availability-zone,Values=us-east-1a

```

Para ver un ejemplo del resultado de `describe-volumes`, consulte el ejemplo 1.

Ejemplo 4: Para describir los volúmenes en función de las etiquetas

El siguiente `describe-volumes` ejemplo describe todos los volúmenes que tienen la clave de etiqueta `Name` y un valor que comienza por `Test`. A continuación, el resultado se filtra con una consulta que muestra únicamente las etiquetas y IDs los volúmenes.

```
aws ec2 describe-volumes \  
  --filters Name=tag:Name,Values=Test* \  
  --query "Volumes[*].{ID:VolumeId,Tag:Tags}"
```

Salida:

```
[  
  {  
    "Tag": [  
      {  
        "Value": "Test2",  
        "Key": "Name"  
      }  
    ],  
    "ID": "vol-1234567890abcdef0"  
  },  
  {  
    "Tag": [  
      {  
        "Value": "Test1",  
        "Key": "Name"  
      }  
    ],  
    "ID": "vol-049df61146c4d7901"  
  }  
]
```

Para ver más ejemplos de uso de filtros de etiquetas, consulta [Cómo trabajar con etiquetas](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DescribeVolumes](#) la Referencia de AWS CLI comandos.

## describe-vpc-attribute

En el siguiente ejemplo de código, se muestra cómo usar `describe-vpc-attribute`.

AWS CLI

Para describir el `enableDnsSupport` atributo

En este ejemplo se describe el `enableDnsSupport` atributo. Este atributo indica si DNS la resolución está habilitada para VPC. Si este atributo lo es `true`, el DNS servidor de Amazon resuelve DNS los nombres de host de sus instancias en sus direcciones IP correspondientes; de lo contrario, no lo hace.

Comando:

```
aws ec2 describe-vpc-attribute --vpc-id vpc-a01106c2 --attribute enableDnsSupport
```

Salida:

```
{
  "VpcId": "vpc-a01106c2",
  "EnableDnsSupport": {
    "Value": true
  }
}
```

Para describir el atributo `enableDnsHostnames`

En este ejemplo se describe el `enableDnsHostnames` atributo. Este atributo indica si las instancias se lanzaron en VPC get DNS hostnames. Si este atributo lo es `true`, las instancias están en el comando VPC get DNS hostnames; de lo contrario, no lo hacen.

Comando:

```
aws ec2 describe-vpc-attribute --vpc-id vpc-a01106c2 --attribute enableDnsHostnames
```

Salida:

```
{
  "VpcId": "vpc-a01106c2",
  "EnableDnsHostnames": {
    "Value": true
  }
}
```

- Para API obtener más información, consulte [DescribeVpcAttribute](#) la Referencia de AWS CLI comandos.

## describe-vpc-classic-link-dns-support

En el siguiente ejemplo de código, se muestra cómo usar `describe-vpc-classic-link-dns-support`.

### AWS CLI

Para describir ClassicLink DNS el soporte para su VPCs

Este ejemplo describe el estado de ClassicLink DNS soporte de todos susVPCs.

Comando:

```
aws ec2 describe-vpc-classic-link-dns-support
```

Salida:

```
{
  "Vpcs": [
    {
      "VpcId": "vpc-88888888",
      "ClassicLinkDnsSupported": true
    },
    {
      "VpcId": "vpc-1a2b3c4d",
      "ClassicLinkDnsSupported": false
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeVpcClassicLinkDnsSupport](#) la Referencia de AWS CLI comandos.

## describe-vpc-classic-link

En el siguiente ejemplo de código, se muestra cómo usar `describe-vpc-classic-link`.

### AWS CLI

Para describir el ClassicLink estado de su VPCs

En este ejemplo se muestra el ClassicLink estado de `vpc-88888888`.

Comando:

```
aws ec2 describe-vpc-classic-link --vpc-id vpc-88888888
```

Salida:

```
{
  "Vpcs": [
    {
      "ClassicLinkEnabled": true,
      "VpcId": "vpc-88888888",
      "Tags": [
        {
          "Value": "classiclinkvpc",
          "Key": "Name"
        }
      ]
    }
  ]
}
```

En este ejemplo, solo se muestran los VPCs que están habilitados para Classiclink (el valor del filtro `is-classic-link-enabled` está establecido en). `true`

Comando:

```
aws ec2 describe-vpc-classic-link --filter "Name=is-classic-link-enabled,Values=true"
```

- Para API obtener más información, consulte la Referencia [DescribeVpcClassicLink](#) de AWS CLI comandos.

## **describe-vpc-endpoint-connection-notifications**

En el siguiente ejemplo de código, se muestra cómo usar `describe-vpc-endpoint-connection-notifications`.

AWS CLI

Para describir las notificaciones de conexión al punto final

En el siguiente `describe-vpc-endpoint-connection-notifications` ejemplo, se describen todas las notificaciones de conexión del punto final.

```
aws ec2 describe-vpc-endpoint-connection-notifications
```

Salida:

```
{
  "ConnectionNotificationSet": [
    {
      "ConnectionNotificationState": "Enabled",
      "ConnectionNotificationType": "Topic",
      "ConnectionEvents": [
        "Accept",
        "Reject",
        "Delete",
        "Connect"
      ],
      "ConnectionNotificationId": "vpce-nfn-04bcb952bc8af7abc",
      "ConnectionNotificationArn": "arn:aws:sns:us-east-1:123456789012:VpceNotification",
      "VpcEndpointId": "vpce-0324151a02f327123"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeVpcEndpointConnectionNotifications](#) la Referencia de AWS CLI comandos.

## **describe-vpc-endpoint-connections**

En el siguiente ejemplo de código, se muestra cómo usar `describe-vpc-endpoint-connections`.

### AWS CLI

Para describir las conexiones de los VPC puntos finales

En este ejemplo, se describen las conexiones de los puntos finales de la interfaz con el servicio de puntos finales y se filtran los resultados para mostrar los puntos finales que sí lo están `PendingAcceptance`.

Comando:

```
aws ec2 describe-vpc-endpoint-connections --filters Name=vpc-endpoint-  
state,Values=pendingAcceptance
```

Salida:

```
{  
  "VpcEndpointConnections": [  
    {  
      "VpcEndpointId": "vpce-0abed31004e618123",  
      "ServiceId": "vpce-svc-0abced088d20def56",  
      "CreationTimestamp": "2017-11-30T10:00:24.350Z",  
      "VpcEndpointState": "pendingAcceptance",  
      "VpcEndpointOwner": "123456789012"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [DescribeVpcEndpointConnections](#) la Referencia de AWS CLI comandos.

## describe-vpc-endpoint-service-configurations

En el siguiente ejemplo de código, se muestra cómo usar `describe-vpc-endpoint-service-configurations`.

AWS CLI

Para describir las configuraciones de los servicios de punto final

En el siguiente `describe-vpc-endpoint-service-configurations` ejemplo, se describen las configuraciones del servicio de puntos finales.

```
aws ec2 describe-vpc-endpoint-service-configurations
```

Salida:

```
{  
  "ServiceConfigurations": [  
    {
```

```
    "ServiceType": [
      {
        "ServiceType": "GatewayLoadBalancer"
      }
    ],
    "ServiceId": "vpce-svc-012d33a1c4321cab",
    "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-
svc-012d33a1c4321cab",
    "ServiceState": "Available",
    "AvailabilityZones": [
      "us-east-1d"
    ],
    "AcceptanceRequired": false,
    "ManagesVpcEndpoints": false,
    "GatewayLoadBalancerArns": [
      "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/
gwy/GWLBSvc/123210844e429123"
    ],
    "Tags": []
  },
  {
    "ServiceType": [
      {
        "ServiceType": "Interface"
      }
    ],
    "ServiceId": "vpce-svc-123cab125efa123",
    "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-123cab125efa123",
    "ServiceState": "Available",
    "AvailabilityZones": [
      "us-east-1a"
    ],
    "AcceptanceRequired": true,
    "ManagesVpcEndpoints": false,
    "NetworkLoadBalancerArns": [
      "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/
net/NLBforSvc/1238753950b25123"
    ],
    "BaseEndpointDnsNames": [
      "vpce-svc-123cab125efa123.us-east-1.vpce.amazonaws.com"
    ],
    "PrivateDnsName": "example.com",
    "PrivateDnsNameConfiguration": {
      "State": "failed",
```



```

        "Type": "TXT",
        "Value": "vpce:qUAth3FdeABCApUiXabc",
        "Name": "_1d367jvbg34znqvyefrj"
    },
    "Tags": []
}
]
}

```

Para obtener más información, consulta [los servicios de VPC punto final](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DescribeVpcEndpointServiceConfigurations](#) la Referencia de AWS CLI comandos.

### **describe-vpc-endpoint-service-permissions**

En el siguiente ejemplo de código, se muestra cómo usar `describe-vpc-endpoint-service-permissions`.

#### AWS CLI

Para describir los permisos de Endpoint Service

En este ejemplo, se describen los permisos del servicio de punto final especificado.

Comando:

```
aws ec2 describe-vpc-endpoint-service-permissions --service-id vpce-  
svc-03d5ebb7d9579a2b3
```

Salida:

```

{
  "AllowedPrincipals": [
    {
      "PrincipalType": "Account",
      "Principal": "arn:aws:iam::123456789012:root"
    }
  ]
}

```

- Para API obtener más información, consulte [DescribeVpcEndpointServicePermissions](#) la Referencia de AWS CLI comandos.

## describe-vpc-endpoint-services

En el siguiente ejemplo de código, se muestra cómo usar `describe-vpc-endpoint-services`.

### AWS CLI

Ejemplo 1: Para describir todos los servicios de VPC punto final

En el siguiente ejemplo «`describe-vpc-endpoint-services`» se enumeran todos los servicios de VPC punto final de una AWS región.

```
aws ec2 describe-vpc-endpoint-services
```

Salida:

```
{
  "ServiceDetails": [
    {
      "ServiceType": [
        {
          "ServiceType": "Gateway"
        }
      ],
      "AcceptanceRequired": false,
      "ServiceName": "com.amazonaws.us-east-1.dynamodb",
      "VpcEndpointPolicySupported": true,
      "Owner": "amazon",
      "AvailabilityZones": [
        "us-east-1a",
        "us-east-1b",
        "us-east-1c",
        "us-east-1d",
        "us-east-1e",
        "us-east-1f"
      ],
      "BaseEndpointDnsNames": [
        "dynamodb.us-east-1.amazonaws.com"
      ]
    }
  ],
}
```

```
{
  "ServiceType": [
    {
      "ServiceType": "Interface"
    }
  ],
  "PrivateDnsName": "ec2.us-east-1.amazonaws.com",
  "ServiceName": "com.amazonaws.us-east-1.ec2",
  "VpcEndpointPolicySupported": false,
  "Owner": "amazon",
  "AvailabilityZones": [
    "us-east-1a",
    "us-east-1b",
    "us-east-1c",
    "us-east-1d",
    "us-east-1e",
    "us-east-1f"
  ],
  "AcceptanceRequired": false,
  "BaseEndpointDnsNames": [
    "ec2.us-east-1.vpce.amazonaws.com"
  ]
},
{
  "ServiceType": [
    {
      "ServiceType": "Interface"
    }
  ],
  "PrivateDnsName": "ssm.us-east-1.amazonaws.com",
  "ServiceName": "com.amazonaws.us-east-1.ssm",
  "VpcEndpointPolicySupported": true,
  "Owner": "amazon",
  "AvailabilityZones": [
    "us-east-1a",
    "us-east-1b",
    "us-east-1c",
    "us-east-1d",
    "us-east-1e"
  ],
  "AcceptanceRequired": false,
  "BaseEndpointDnsNames": [
    "ssm.us-east-1.vpce.amazonaws.com"
  ]
}
```

```

    }
  ],
  "ServiceNames": [
    "com.amazonaws.us-east-1.dynamodb",
    "com.amazonaws.us-east-1.ec2",
    "com.amazonaws.us-east-1.ec2messages",
    "com.amazonaws.us-east-1.elasticloadbalancing",
    "com.amazonaws.us-east-1.kinesis-streams",
    "com.amazonaws.us-east-1.s3",
    "com.amazonaws.us-east-1.ssm"
  ]
}

```

Para obtener más información, consulte [Ver los nombres AWS de los servicios disponibles](#) en la Guía del usuario de AWS PrivateLink.

Ejemplo 2: Para describir los detalles de un servicio de punto final

En el siguiente ejemplo `describe-vpc-endpoint-services` «" se enumeran los detalles del servicio de punto final de la interfaz Amazon S3.

```

aws ec2 describe-vpc-endpoint-services \
  --filter "Name=service-type,Values=Interface" Name=service-
  name,Values=com.amazonaws.us-east-1.s3

```

Salida:

```

{
  "ServiceDetails": [
    {
      "ServiceName": "com.amazonaws.us-east-1.s3",
      "ServiceId": "vpce-svc-081d84efcdEXAMPLE",
      "ServiceType": [
        {
          "ServiceType": "Interface"
        }
      ],
      "AvailabilityZones": [
        "us-east-1a",
        "us-east-1b",
        "us-east-1c",
        "us-east-1d",
        "us-east-1e",

```

```

        "us-east-1f"
    ],
    "Owner": "amazon",
    "BaseEndpointDnsNames": [
        "s3.us-east-1.vpce.amazonaws.com"
    ],
    "VpcEndpointPolicySupported": true,
    "AcceptanceRequired": false,
    "ManagesVpcEndpoints": false,
    "Tags": []
    }
],
"ServiceNames": [
    "com.amazonaws.us-east-1.s3"
]
}

```

Para obtener más información, consulte [Ver los nombres de los AWS servicios disponibles](#) en la Guía del usuario de AWS PrivateLink.

- Para API obtener más información, consulte [DescribeVpcEndpointServices](#) la Referencia de AWS CLI comandos.

## describe-vpc-endpoints

En el siguiente ejemplo de código, se muestra cómo usar `describe-vpc-endpoints`.

### AWS CLI

Para describir sus puntos VPC finales

En el siguiente `describe-vpc-endpoints` ejemplo, se muestran los detalles de todos los puntos de conexión VPC.

```
aws ec2 describe-vpc-endpoints
```

Salida:

```

{
  "VpcEndpoints": [
    {
      "PolicyDocument": "{\"Version\":\"2008-10-17\",\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":\"*\",\"Action\":\"*\",\"Resource\":\"*\"}]}",

```

```

    "VpcId": "vpc-aabb1122",
    "NetworkInterfaceIds": [],
    "SubnetIds": [],
    "PrivateDnsEnabled": true,
    "State": "available",
    "ServiceName": "com.amazonaws.us-east-1.dynamodb",
    "RouteTableIds": [
        "rtb-3d560345"
    ],
    "Groups": [],
    "VpcEndpointId": "vpce-032a826a",
    "VpcEndpointType": "Gateway",
    "CreationTimestamp": "2017-09-05T20:41:28Z",
    "DnsEntries": [],
    "OwnerId": "123456789012"
},
{
    "PolicyDocument": "{\n  \"Statement\": [\n    {\n      \"Action\": \"*\n\", \n    \"Effect\": \"Allow\", \n      \"Principal\": \"*\", \n      \"Resource\n\": \"*\">\n  ]\n }",
    "VpcId": "vpc-1a2b3c4d",
    "NetworkInterfaceIds": [
        "eni-2ec2b084",
        "eni-1b4a65cf"
    ],
    "SubnetIds": [
        "subnet-d6fcaa8d",
        "subnet-7b16de0c"
    ],
    "PrivateDnsEnabled": false,
    "State": "available",
    "ServiceName": "com.amazonaws.us-east-1.elasticloadbalancing",
    "RouteTableIds": [],
    "Groups": [
        {
            "GroupName": "default",
            "GroupId": "sg-54e8bf31"
        }
    ],
    "VpcEndpointId": "vpce-0f89a33420c1931d7",
    "VpcEndpointType": "Interface",
    "CreationTimestamp": "2017-09-05T17:55:27.583Z",
    "DnsEntries": [
        {

```

```

        "HostedZoneId": "Z7HUB22UULQXV",
        "DnsName": "vpce-0f89a33420c1931d7-
bluzidnv.elasticloadbalancing.us-east-1.vpce.amazonaws.com"
    },
    {
        "HostedZoneId": "Z7HUB22UULQXV",
        "DnsName": "vpce-0f89a33420c1931d7-bluzidnv-us-
east-1b.elasticloadbalancing.us-east-1.vpce.amazonaws.com"
    },
    {
        "HostedZoneId": "Z7HUB22UULQXV",
        "DnsName": "vpce-0f89a33420c1931d7-bluzidnv-us-
east-1a.elasticloadbalancing.us-east-1.vpce.amazonaws.com"
    }
],
"OwnerId": "123456789012"
},
{
    "VpcEndpointId": "vpce-aabbaabbaabbaabba",
    "VpcEndpointType": "GatewayLoadBalancer",
    "VpcId": "vpc-111122223333aabbc",
    "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-
svc-123123a1c43abc123",
    "State": "available",
    "SubnetIds": [
        "subnet-0011aabbcc2233445"
    ],
    "RequesterManaged": false,
    "NetworkInterfaceIds": [
        "eni-01010120203030405"
    ],
    "CreationTimestamp": "2020-11-11T08:06:03.522Z",
    "Tags": [],
    "OwnerId": "123456789012"
}
]
}

```

Para obtener más información, consulta los [VPC puntos finales](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DescribeVpcEndpoints](#) la Referencia de AWS CLI comandos.

## describe-vpc-peering-connections

En el siguiente ejemplo de código, se muestra cómo usar `describe-vpc-peering-connections`.

### AWS CLI

Para describir sus conexiones de VPC interconexión

En este ejemplo se describen todas las conexiones de VPC emparejamiento.

Comando:

```
aws ec2 describe-vpc-peering-connections
```

Salida:

```
{
  "VpcPeeringConnections": [
    {
      "Status": {
        "Message": "Active",
        "Code": "active"
      },
      "Tags": [
        {
          "Value": "Peering-1",
          "Key": "Name"
        }
      ],
      "AccepterVpcInfo": {
        "OwnerId": "111122223333",
        "VpcId": "vpc-1a2b3c4d",
        "CidrBlock": "10.0.1.0/28"
      },
      "VpcPeeringConnectionId": "pcx-11122233",
      "RequesterVpcInfo": {
        "PeeringOptions": {
          "AllowEgressFromLocalVpcToRemoteClassicLink": false,
          "AllowEgressFromLocalClassicLinkToRemoteVpc": false
        },
        "OwnerId": "444455556666",
        "VpcId": "vpc-123abc45",

```



```

        "CidrBlock": "192.168.0.0/16"
      }
    },
    {
      "Status": {
        "Message": "Pending Acceptance by 444455556666",
        "Code": "pending-acceptance"
      },
      "Tags": [],
      "RequesterVpcInfo": {
        "PeeringOptions": {
          "AllowEgressFromLocalVpcToRemoteClassicLink": false,
          "AllowEgressFromLocalClassicLinkToRemoteVpc": false
        },
        "OwnerId": "444455556666",
        "VpcId": "vpc-11aa22bb",
        "CidrBlock": "10.0.0.0/28"
      },
      "VpcPeeringConnectionId": "pcx-abababab",
      "ExpirationTime": "2014-04-03T09:12:43.000Z",
      "AccepterVpcInfo": {
        "OwnerId": "444455556666",
        "VpcId": "vpc-33cc44dd"
      }
    }
  ]
}

```

Para describir conexiones de VPC emparejamiento específicas

En este ejemplo, se describen todas las conexiones entre VPC pares que se encuentran en estado de aceptación pendiente.

Comando:

```
aws ec2 describe-vpc-peering-connections --filters Name=status-code,Values=pending-acceptance
```

En este ejemplo, se describen todas las conexiones entre pares que tienen la VPC etiqueta Owner=Finance.

Comando:

```
aws ec2 describe-vpc-peering-connections --filters Name=tag:Owner,Values=Finance
```

En este ejemplo, se describen todas las conexiones de VPC emparejamiento que ha solicitado para el vpc-1a2b3c4d especificado. VPC

Comando:

```
aws ec2 describe-vpc-peering-connections --filters Name=requester-vpc-info.vpc-id,Values=vpc-1a2b3c4d
```

- Para obtener más información, consulte la Referencia de comandos. API [DescribeVpcPeeringConnections](#) AWS CLI

## describe-vpcs

En el siguiente ejemplo de código, se muestra cómo usar describe-vpcs.

AWS CLI

Ejemplo 1: Para describir todos sus VPCs

El siguiente describe-vpcs ejemplo recupera detalles sobre su. VPCs

```
aws ec2 describe-vpcs
```

Salida:

```
{
  "Vpcs": [
    {
      "CidrBlock": "30.1.0.0/16",
      "DhcpOptionsId": "dopt-19edf471",
      "State": "available",
      "VpcId": "vpc-0e9801d129EXAMPLE",
      "OwnerId": "111122223333",
      "InstanceTenancy": "default",
      "CidrBlockAssociationSet": [
        {
          "AssociationId": "vpc-cidr-assoc-062c64cfafEXAMPLE",
```

```

        "CidrBlock": "30.1.0.0/16",
        "CidrBlockState": {
            "State": "associated"
        }
    },
    "IsDefault": false,
    "Tags": [
        {
            "Key": "Name",
            "Value": "Not Shared"
        }
    ]
},
{
    "CidrBlock": "10.0.0.0/16",
    "DhcpOptionsId": "dopt-19edf471",
    "State": "available",
    "VpcId": "vpc-06e4ab6c6cEXAMPLE",
    "OwnerId": "222222222222",
    "InstanceTenancy": "default",
    "CidrBlockAssociationSet": [
        {
            "AssociationId": "vpc-cidr-assoc-00b17b4eddEXAMPLE",
            "CidrBlock": "10.0.0.0/16",
            "CidrBlockState": {
                "State": "associated"
            }
        }
    ],
    "IsDefault": false,
    "Tags": [
        {
            "Key": "Name",
            "Value": "Shared VPC"
        }
    ]
}
]
}
}

```

Ejemplo 2: Para describir un objeto específico VPC

En el siguiente `describe-vpcs` ejemplo, se recuperan los detalles de lo especificado VPC.

```
aws ec2 describe-vpcs \  
--vpc-ids vpc-06e4ab6c6cEXAMPLE
```

Salida:

```
{  
  "Vpcs": [  
    {  
      "CidrBlock": "10.0.0.0/16",  
      "DhcpOptionsId": "dopt-19edf471",  
      "State": "available",  
      "VpcId": "vpc-06e4ab6c6cEXAMPLE",  
      "OwnerId": "111122223333",  
      "InstanceTenancy": "default",  
      "CidrBlockAssociationSet": [  
        {  
          "AssociationId": "vpc-cidr-assoc-00b17b4eddEXAMPLE",  
          "CidrBlock": "10.0.0.0/16",  
          "CidrBlockState": {  
            "State": "associated"  
          }  
        }  
      ],  
      "IsDefault": false,  
      "Tags": [  
        {  
          "Key": "Name",  
          "Value": "Shared VPC"  
        }  
      ]  
    }  
  ]  
}
```

- Para API obtener más información, consulte [DescribeVpcs](#) la Referencia de AWS CLI comandos.

## describe-vpn-connections

En el siguiente ejemplo de código, se muestra cómo usar `describe-vpn-connections`.

## AWS CLI

### Ejemplo 1: Para describir sus VPN conexiones

El siguiente describe-`vpn-connections` ejemplo describe todas sus conexiones de sitio a sitioVPN.

```
aws ec2 describe-vpn-connections
```

Salida:

```
{
  "VpnConnections": [
    {
      "CustomerGatewayConfiguration": "...configuration information...",
      "CustomerGatewayId": "cgw-01234567abcde1234",
      "Category": "VPN",
      "State": "available",
      "Type": "ipsec.1",
      "VpnConnectionId": "vpn-1122334455aabbccd",
      "TransitGatewayId": "tgw-00112233445566aab",
      "Options": {
        "EnableAcceleration": false,
        "StaticRoutesOnly": true,
        "LocalIpv4NetworkCidr": "0.0.0.0/0",
        "RemoteIpv4NetworkCidr": "0.0.0.0/0",
        "TunnelInsideIpVersion": "ipv4"
      },
      "Routes": [],
      "Tags": [
        {
          "Key": "Name",
          "Value": "CanadaVPN"
        }
      ],
      "VgwTelemetry": [
        {
          "AcceptedRouteCount": 0,
          "LastStatusChange": "2020-07-29T10:35:11.000Z",
          "OutsideIpAddress": "203.0.113.3",
          "Status": "DOWN",
          "StatusMessage": ""
        }
      ],
    }
  ]
}
```

```

        {
            "AcceptedRouteCount": 0,
            "LastStatusChange": "2020-09-02T09:09:33.000Z",
            "OutsideIpAddress": "203.0.113.5",
            "Status": "UP",
            "StatusMessage": ""
        }
    ]
}

```

Para obtener más información, consulte [Cómo VPN funciona un AWS sitio a otro en la Guía del usuario de sitio a sitio](#).AWS VPN

Ejemplo 2: Para describir las conexiones disponibles VPN

En el siguiente `describe-vpn-connections` ejemplo, se describen VPN las conexiones de sitio a sitio con un estado de `available`

```

aws ec2 describe-vpn-connections \
  --filters "Name=state,Values=available"

```

Para obtener más información, consulte [Cómo VPN funciona un AWS sitio a otro en la Guía del usuario de sitio a sitio](#).AWS VPN

- Para obtener API más información, consulte la Referencia de comandos [DescribeVpnConnections](#).AWS CLI

## **describe-vpn-gateways**

En el siguiente ejemplo de código, se muestra cómo usar `describe-vpn-gateways`.

AWS CLI

Para describir sus puertas de enlace privadas virtuales

En este ejemplo se describen sus puertas de enlace privadas virtuales.

Comando:

```

aws ec2 describe-vpn-gateways

```

Salida:

```
{
  "VpnGateways": [
    {
      "State": "available",
      "Type": "ipsec.1",
      "VpnGatewayId": "vgw-f211f09b",
      "VpcAttachments": [
        {
          "State": "attached",
          "VpcId": "vpc-98eb5ef5"
        }
      ]
    },
    {
      "State": "available",
      "Type": "ipsec.1",
      "VpnGatewayId": "vgw-9a4cacf3",
      "VpcAttachments": [
        {
          "State": "attaching",
          "VpcId": "vpc-a01106c2"
        }
      ]
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeVpnGateways](#) la Referencia de AWS CLI comandos.

## **detach-classic-link-vpc**

En el siguiente ejemplo de código, se muestra cómo usar `detach-classic-link-vpc`.

AWS CLI

Para desvincular (separar) una instancia EC2 -Classic de una VPC

Este ejemplo desvincula la instancia `i-0598c7d356eba48d7` de `vpc-88888888`. VPC

Comando:

```
aws ec2 detach-classic-link-vpc --instance-id i-0598c7d356eba48d7 --vpc-id vpc-88888888
```

Salida:

```
{  
  "Return": true  
}
```

- API Para obtener más [DetachClassicLinkVpc](#) información, consulte la Referencia de comandos.AWS CLI

## detach-internet-gateway

En el siguiente ejemplo de código, se muestra cómo usar detach-internet-gateway.

AWS CLI

Para separar una puerta de enlace de Internet de su VPC

El siguiente detach-internet-gateway ejemplo separa la puerta de enlace de Internet especificada de la específica. VPC

```
aws ec2 detach-internet-gateway \  
  --internet-gateway-id igw-0d0fb496b3EXAMPLE \  
  --vpc-id vpc-0a60eb65b4EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [las pasarelas de Internet](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DetachInternetGateway](#) la Referencia de AWS CLI comandos.

## detach-network-interface

En el siguiente ejemplo de código, se muestra cómo usar detach-network-interface.



## AWS CLI

Para separar una interfaz de red de la instancia

En este ejemplo, se separa la interfaz de red especificada de la instancia especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 detach-network-interface --attachment-id eni-attach-66c4350a
```

- Para API obtener más información, consulte [DetachNetworkInterface](#) la Referencia de AWS CLI comandos.

## **detach-verified-access-trust-provider**

En el siguiente ejemplo de código, se muestra cómo usar `detach-verified-access-trust-provider`.

## AWS CLI

Para separar un proveedor de confianza de una instancia

En el siguiente `detach-verified-access-trust-provider` ejemplo, se separa el proveedor de confianza de Verified Access especificado de la instancia de Verified Access especificada.

```
aws ec2 detach-verified-access-trust-provider \  
  --verified-access-instance-id vai-0ce000c0b7643abea \  
  --verified-access-trust-provider-id vatp-0bb32de759a3e19e7
```

Salida:

```
{  
  "VerifiedAccessTrustProvider": {  
    "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",  
    "Description": "Testing Verified Access",  
    "TrustProviderType": "user",  
    "UserTrustProviderType": "iam-identity-center",  
    "PolicyReferenceName": "idc",
```

```
    "CreationTime": "2023-08-25T19:00:38",
    "LastUpdatedTime": "2023-08-25T19:00:38"
  },
  "VerifiedAccessInstance": {
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "Description": "Testing Verified Access",
    "VerifiedAccessTrustProviders": [],
    "CreationTime": "2023-08-25T18:27:56",
    "LastUpdatedTime": "2023-08-25T18:27:56"
  }
}
```

Para obtener más información, consulte las [instancias de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [DetachVerifiedAccessTrustProvider](#) la Referencia de AWS CLI comandos.

## detach-volume

En el siguiente ejemplo de código, se muestra cómo usar detach-volume.

### AWS CLI

Para separar un volumen de una instancia

Este comando de ejemplo separa el volumen (vol-049df61146c4d7901) de la instancia a la que está conectado.

Comando:

```
aws ec2 detach-volume --volume-id vol-1234567890abcdef0
```

Salida:

```
{
  "AttachTime": "2014-02-27T19:23:06.000Z",
  "InstanceId": "i-1234567890abcdef0",
  "VolumeId": "vol-049df61146c4d7901",
  "State": "detaching",
  "Device": "/dev/sdb"
```

```
}
```

- Para API obtener más información, consulte [DetachVolume](#) la Referencia de AWS CLI comandos.

## **detach-vpn-gateway**

En el siguiente ejemplo de código, se muestra cómo usar `detach-vpn-gateway`.

### AWS CLI

Para separar una puerta de enlace privada virtual de su VPC

En este ejemplo, se separa la puerta de enlace privada virtual especificada de la especificada. VPC Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 detach-vpn-gateway --vpn-gateway-id vgw-9a4cacf3 --vpc-id vpc-a01106c2
```

- Para API obtener más información, consulte [DetachVpnGateway](#) la Referencia de AWS CLI comandos.

## **disable-address-transfer**

En el siguiente ejemplo de código, se muestra cómo usar `disable-address-transfer`.

### AWS CLI

Para deshabilitar una transferencia de direcciones IP elásticas

El siguiente `disable-address-transfer` ejemplo desactiva la transferencia de direcciones IP elásticas para la dirección IP elástica especificada.

```
aws ec2 disable-address-transfer \  
  --allocation-id eipalloc-09ad461b0d03f6aaf
```

Salida:

```
{
```

```
"AddressTransfer": {
  "PublicIp": "100.21.184.216",
  "AllocationId": "eipalloc-09ad461b0d03f6aaf",
  "AddressTransferStatus": "disabled"
}
}
```

Para obtener más información, consulta [Transferir direcciones IP elásticas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DisableAddressTransfer](#) la Referencia de AWS CLI comandos.

## disable-aws-network-performance-metric-subscription

En el siguiente ejemplo de código, se muestra cómo usar `disable-aws-network-performance-metric-subscription`.

### AWS CLI

Para deshabilitar una suscripción métrica

El siguiente `disable-aws-network-performance-metric-subscription` ejemplo deshabilita la supervisión de la latencia de red agregada entre las regiones de origen y destino especificadas.

```
aws ec2 disable-aws-network-performance-metric-subscription \
  --source us-east-1 \
  --destination eu-west-1 \
  --metric aggregate-latency \
  --statistic p50
```

Salida:

```
{
  "Output": true
}
```

Para obtener más información, consulte [Administrar las suscripciones](#) en la Guía del usuario de Infrastructure Performance.

- Para API obtener más información, consulte [DisableAwsNetworkPerformanceMetricSubscription](#) la Referencia de AWS CLI comandos.

## **disable-ebs-encryption-by-default**

En el siguiente ejemplo de código, se muestra cómo usar `disable-ebs-encryption-by-default`.

### AWS CLI

Para deshabilitar el EBS cifrado de forma predeterminada

El siguiente `disable-ebs-encryption-by-default` ejemplo deshabilita el EBS cifrado de forma predeterminada para su AWS cuenta en la región actual.

```
aws ec2 disable-ebs-encryption-by-default
```

Salida:

```
{
  "EbsEncryptionByDefault": false
}
```

- Para API obtener más información, consulte [DisableEbsEncryptionByDefault](#) la Referencia de AWS CLI comandos.

## **disable-fast-launch**

En el siguiente ejemplo de código, se muestra cómo usar `disable-fast-launch`.

### AWS CLI

Para interrumpir el inicio rápido de una imagen

En el siguiente `disable-fast-launch` ejemplo, se interrumpe el inicio rápido en la zona especificada AMI y se limpian las instantáneas preaprovisionadas existentes.

```
aws ec2 disable-fast-launch \
  --image-id ami-01234567890abcdef
```

Salida:

```
{
  "ImageId": "ami-01234567890abcdef",
  "ResourceType": "snapshot",
  "SnapshotConfiguration": {},
  "LaunchTemplate": {
    "LaunchTemplateId": "lt-01234567890abcdef",
    "LaunchTemplateName": "EC2FastLaunchDefaultResourceCreation-
a8c6215d-94e6-441b-9272-dbd1f87b07e2",
    "Version": "1"
  },
  "MaxParallelLaunches": 6,
  "OwnerId": "0123456789123",
  "State": "disabling",
  "StateTransitionReason": "Client.UserInitiated",
  "StateTransitionTime": "2022-01-27T22:47:29.265000+00:00"
}
```

Para obtener más información sobre cómo configurar un Windows AMI para un inicio más rápido, consulte [Configurar su sistema AMI para un inicio más rápido](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DisableFastLaunch](#) la Referencia de AWS CLI comandos.

## **disable-fast-snapshot-restores**

En el siguiente ejemplo de código, se muestra cómo usar `disable-fast-snapshot-restores`.

AWS CLI

Para deshabilitar la restauración rápida de instantáneas

El siguiente `disable-fast-snapshot-restores` ejemplo deshabilita la restauración rápida de instantáneas para la instantánea especificada en la zona de disponibilidad especificada.

```
aws ec2 disable-fast-snapshot-restores \
  --availability-zones us-east-2a \
  --source-snapshot-ids snap-1234567890abcdef0
```

Salida:

```
{
  "Successful": [
    {
      "SnapshotId": "snap-1234567890abcdef0"
      "AvailabilityZone": "us-east-2a",
      "State": "disabling",
      "StateTransitionReason": "Client.UserInitiated",
      "OwnerId": "123456789012",
      "EnablingTime": "2020-01-25T23:57:49.602Z"
    }
  ],
  "Unsuccessful": []
}
```

- Para API obtener más información, consulte [DisableFastSnapshotRestores](#) la Referencia de AWS CLI comandos.

## disable-image-block-public-access

En el siguiente ejemplo de código, se muestra cómo usar `disable-image-block-public-access`.

### AWS CLI

Para deshabilitar, bloquear el acceso público AMIs en la región especificada

El siguiente `disable-image-block-public-access` ejemplo desactiva el bloqueo del acceso público AMIs a nivel de cuenta en la región especificada.

```
aws ec2 disable-image-block-public-access \
  --region us-east-1
```

Salida:

```
{
  "ImageBlockPublicAccessState": "unblocked"
}
```

Para obtener más información, consulta [Bloquear el acceso público a tu contenido AMIs](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DisableImageBlockPublicAccess](#) la Referencia de AWS CLI comandos.

## disable-image-deprecation

En el siguiente ejemplo de código, se muestra cómo usar `disable-image-deprecation`.

### AWS CLI

Para cancelar la obsolescencia de un AMI

El siguiente `disable-image-deprecation` ejemplo cancela la obsolescencia de un AMI, lo que elimina el `DeprecationTime` campo de la salida. `describe-images` Debe ser el AMI propietario para realizar este procedimiento.

```
aws ec2 disable-image-deprecation \  
  --image-id ami-1234567890abcdef0
```

Salida:

```
{  
  "RequestID": "11aabb229-4eac-35bd-99ed-be587EXAMPLE",  
  "Return": "true"  
}
```

Para obtener más información, consulte Eliminar un AMI <<https://docs.aws.amazon.com/AWS-EC2/latest/UserGuide/ami-deprecate.html#deprecate-ami>> en la Guía del usuario de Amazon. EC2

- Para obtener API más información, consulte la Referencia de comandos. [DisableImageDeprecation](#) AWS CLI

## disable-image

En el siguiente ejemplo de código, se muestra cómo usar `disable-image`.

### AWS CLI

Para deshabilitar un AMI

En el siguiente `disable-image` ejemplo, se deshabilita lo especificado AMI.



```
aws ec2 disable-image \  
  --image-id ami-1234567890abcdef0
```

Salida:

```
{  
  "Return": "true"  
}
```

Para obtener más información, consulta Cómo [deshabilitar una AMI](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DisableImage](#) la Referencia de AWS CLI comandos.

## **disable-ipam-organization-admin-account**

En el siguiente ejemplo de código, se muestra cómo usar `disable-ipam-organization-admin-account`.

### AWS CLI

Para deshabilitar la administración delegada IPAM

En algunos casos, se integrará IPAM con AWS Organizations. Al hacerlo, la cuenta de administración de AWS AWS Organizations delega como IPAM administrador a una cuenta de miembro de Organizations.

En este ejemplo, usted es la AWS cuenta de administración de Organizations que delegó la cuenta de IPAM administrador y quiere inhabilitar esa cuenta para que no sea la de IPAM administrador.

Puedes usar cualquier AWS región `--region` al realizar esta solicitud. No es necesario que utilices la región en la que delegaste originalmente el administrador, donde IPAM se creó o una región IPAM operativa. Si inhabilitas la cuenta de administrador delegado, puedes volver a habilitarla en cualquier momento o delegar una nueva cuenta como administrador. IPAM

En el siguiente `disable-ipam-organization-admin-account` ejemplo, se inhabilita el IPAM administrador delegado en tu cuenta. AWS

```
aws ec2 disable-ipam-organization-admin-account \  
  --delegated-admin-account-id 320805250157 \  
  --region ap-south-1
```

Salida:

```
{  
  "Success": true  
}
```

Para obtener más información, consulta [Integrar IPAM con cuentas de una AWS organización](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [DisableIpamOrganizationAdminAccount](#) la Referencia de AWS CLI comandos.

## **disable-serial-console-access**

En el siguiente ejemplo de código, se muestra cómo usar `disable-serial-console-access`.

AWS CLI

Para deshabilitar el acceso a la consola EC2 en serie de su cuenta

En el siguiente `disable-serial-console-access` ejemplo, se deshabilita el acceso de la cuenta a la consola en serie.

```
aws ec2 disable-serial-console-access
```

Salida:

```
{  
  "SerialConsoleAccessEnabled": false  
}
```

Para obtener más información, consulta [EC2Serial Console](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DisableSerialConsoleAccess](#) la Referencia de AWS CLI comandos.

## **disable-snapshot-block-public-access**

En el siguiente ejemplo de código, se muestra cómo usar `disable-snapshot-block-public-access`.

### AWS CLI

Para deshabilitar el bloqueo del acceso público a las instantáneas

El siguiente `disable-snapshot-block-public-access` ejemplo desactiva el bloqueo del acceso público a las instantáneas para permitir que se compartan públicamente las instantáneas.

```
aws ec2 disable-snapshot-block-public-access
```

Salida:

```
{
  "State": "unblocked"
}
```

Para obtener más información, consulta [Bloquear el acceso público a las instantáneas](#) en la Guía del EBS usuario de Amazon.

- Para API obtener más información, consulte [DisableSnapshotBlockPublicAccess](#) la Referencia de AWS CLI comandos.

## **disable-transit-gateway-route-table-propagation**

En el siguiente ejemplo de código, se muestra cómo usar `disable-transit-gateway-route-table-propagation`.

### AWS CLI

Para deshabilitar un adjunto a una pasarela de tránsito para propagar las rutas a la tabla de rutas de propagación especificada

El siguiente `disable-transit-gateway-route-table-propagation` ejemplo deshabilita el adjunto especificado para propagar las rutas a la tabla de rutas de propagación especificada.

```
aws ec2 disable-transit-gateway-route-table-propagation \
```

```
--transit-gateway-route-table-id tgw-rtb-0a823edbdeEXAMPLE \  
--transit-gateway-attachment-id tgw-attach-09b52ccdb5EXAMPLE
```

Salida:

```
{  
  "Propagation": {  
    "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",  
    "ResourceId": "vpc-4d7de228",  
    "ResourceType": "vpc",  
    "TransitGatewayRouteTableId": "tgw-rtb-0a823edbdeEXAMPLE",  
    "State": "disabled"  
  }  
}
```

Para obtener más información, consulte [las tablas de rutas de Transit Gateways](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [DisableTransitGatewayRouteTablePropagation](#) la Referencia de AWS CLI comandos.

## **disable-vgw-route-propagation**

En el siguiente ejemplo de código, se muestra cómo usar `disable-vgw-route-propagation`.

### AWS CLI

Para deshabilitar la propagación de rutas

En este ejemplo, se impide que la puerta de enlace privada virtual especificada propague rutas estáticas a la tabla de rutas especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 disable-vgw-route-propagation --route-table-id rtb-22574640 --gateway-  
id vgw-9a4cacf3
```

- Para API obtener más información, consulte la Referencia [DisableVgwRoutePropagation](#) de AWS CLI comandos.

## **disable-vpc-classic-link-dns-support**

En el siguiente ejemplo de código, se muestra cómo usar `disable-vpc-classic-link-dns-support`.

### AWS CLI

Para deshabilitar la ClassicLink DNS compatibilidad con un VPC

En este ejemplo se inhabilita la ClassicLink DNS compatibilidad con. `vpc-88888888`

Comando:

```
aws ec2 disable-vpc-classic-link-dns-support --vpc-id vpc-88888888
```

Salida:

```
{
  "Return": true
}
```

- Para API obtener más información, consulte [DisableVpcClassicLinkDnsSupport](#) la Referencia de AWS CLI comandos.

## **disable-vpc-classic-link**

En el siguiente ejemplo de código, se muestra cómo usar `disable-vpc-classic-link`.

### AWS CLI

Para deshabilitar ClassicLink para un VPC

En este ejemplo, se desactiva la ClassicLink `vpc-88888888`.

Comando:

```
aws ec2 disable-vpc-classic-link --vpc-id vpc-88888888
```

Salida:

```
{
```

```
"Return": true
}
```

- Para API obtener más información, consulte la Referencia de [DisableVpcClassicLink](#) comandos AWS CLI .

## **disassociate-address**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-address`.

### AWS CLI

Para desasociar una dirección IP elástica en -Classic EC2

En este ejemplo, se disocia una dirección IP elástica de una instancia de -Classic. EC2 Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 disassociate-address --public-ip 198.51.100.0
```

Para desasociar una dirección IP elástica en - EC2 VPC

En este ejemplo, se disocia una dirección IP elástica de una instancia de un. VPC Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 disassociate-address --association-id eipassoc-2bebb745
```

- Para API obtener más información, consulte [DisassociateAddress](#) la Referencia de AWS CLI comandos.

## **disassociate-client-vpn-target-network**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-client-vpn-target-network`.

### AWS CLI

Para desasociar una red de un punto final del cliente VPN

El siguiente `disassociate-client-vpn-target-network` ejemplo desasocia la red de destino que está asociada al ID de `cvpn-assoc-12312312312312312` asociación del punto final del cliente VPN especificado.

```
aws ec2 disassociate-client-vpn-target-network \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
  --association-id cvpn-assoc-12312312312312312
```

Salida:

```
{
  "AssociationId": "cvpn-assoc-12312312312312312",
  "Status": {
    "Code": "disassociating"
  }
}
```

Para obtener más información, consulte [Target Networks](#) en la Guía del VPN administrador de AWS clientes.

- Para API obtener más información, consulte [DisassociateClientVpnTargetNetwork](#) la Referencia de AWS CLI comandos.

## **disassociate-iam-instance-profile**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-iam-instance-profile`.

AWS CLI

Para desasociar un perfil de IAM instancia

En este ejemplo, se disocia un perfil de IAM instancia del ID de asociación. `iip-assoc-05020b59952902f5f`

Comando:

```
aws ec2 disassociate-iam-instance-profile --association-id iip-
assoc-05020b59952902f5f
```

Salida:

```
{
  "IamInstanceProfileAssociation": {
    "InstanceId": "i-123456789abcde123",
    "State": "disassociating",
    "AssociationId": "iip-assoc-05020b59952902f5f",
    "IamInstanceProfile": {
      "Id": "AIPAI5IVIHMFYY2DKV5Y",
      "Arn": "arn:aws:iam::123456789012:instance-profile/admin-role"
    }
  }
}
```

- Para API obtener más información, consulte [Disasociar el perfil de instancia IAM](#) la Referencia de AWS CLI comandos.

## disassociate-instance-event-window

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-instance-event-window`.

AWS CLI

Ejemplo 1: Para desasociar una o más instancias de una ventana de eventos

El siguiente `disassociate-instance-event-window` ejemplo desasocia una o más instancias de una ventana de eventos. Especifique el `instance-event-window-id` parámetro para especificar la ventana de eventos. Para desasociar las instancias, especifique el `association-target` parámetro y, para los valores de los parámetros, especifique una o más instanciasIDs.

```
aws ec2 disassociate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

Salida:

```
{
```



```

    "InstanceEventWindow": {
      "InstanceEventWindowId": "iew-0abcdef1234567890",
      "Name": "myEventWindowName",
      "CronExpression": "* 21-23 * * 2,3",
      "AssociationTarget": {
        "InstanceIds": [],
        "Tags": [],
        "DedicatedHostIds": []
      },
      "State": "creating"
    }
  }
}

```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para desasociar las etiquetas de instancia de una ventana de eventos

El siguiente `disassociate-instance-event-window` ejemplo desasocia las etiquetas de instancia de una ventana de eventos. Especifique el `instance-event-window-id` parámetro para especificar la ventana de eventos. Para desasociar las etiquetas de instancia, especifique el parámetro `association-target`, y para los valores del parámetro, especifique una o varias etiquetas.

```

aws ec2 disassociate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target "InstanceTags=[{Key=k2, Value=v2}, {Key=k1, Value=v1}]"

```

Salida:

```

{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [],
      "Tags": [],
      "DedicatedHostIds": []
    },
  },
}

```

```

    "State": "creating"
  }
}

```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

Ejemplo 3: Para desasociar un anfitrión dedicado de una ventana de eventos

El siguiente `disassociate-instance-event-window` ejemplo desasocia un host dedicado de una ventana de eventos. Especifique el `instance-event-window-id` parámetro para especificar la ventana de eventos. Para desasociar un host dedicado, especifique el `association-target` parámetro y, para los valores de los parámetros, especifique uno o más hosts IDs dedicados.

```

aws ec2 disassociate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target DedicatedHostIds=h-029fa35a02b99801d

```

Salida:

```

{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [],
      "Tags": [],
      "DedicatedHostIds": []
    },
    "State": "creating"
  }
}

```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DisassociateInstanceEventWindow](#) la Referencia de AWS CLI comandos.

## disassociate-ipam-resource-discovery

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-ipam-resource-discovery`.

### AWS CLI

Para desasociar un descubrimiento de recursos de un IPAM

En este ejemplo, usted es una cuenta de administrador IPAM delegado y desea desasociar una detección de IPAM recursos de la suya. IPAM Ejecutó el comando describe y observó que quería desasociarlo del suyo para dejar espacio IPAM a otras asociaciones.

```
"ResourceDiscoveryStatus": "not-found"
```

En el siguiente `disassociate-ipam-resource-discovery` ejemplo, se disocia una detección IPAM de recursos en tu AWS cuenta.

```
aws ec2 disassociate-ipam-resource-discovery \  
  --ipam-resource-discovery-association-id ipam-res-disco-assoc-04382a6346357cf82 \  
  --region us-east-1
```

Salida:

```
{  
  "IpamResourceDiscoveryAssociation": {  
    "OwnerId": "320805250157",  
    "IpamResourceDiscoveryAssociationId": "ipam-res-disco-  
assoc-04382a6346357cf82",  
    "IpamResourceDiscoveryAssociationArn":  
"arn:aws:ec2::320805250157:ipam-resource-discovery-association/ipam-res-disco-  
assoc-04382a6346357cf82",  
    "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",  
    "IpamId": "ipam-005f921c17ebd5107",  
    "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",  
    "IpamRegion": "us-east-1",  
    "IsDefault": false,  
    "ResourceDiscoveryStatus": "not-found",  
    "State": "disassociate-in-progress"  
  }  
}
```

Para obtener más información, consulta la [sección Integración IPAM con cuentas ajenas a tu organización](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [DisassociateIpamResourceDiscovery](#) la Referencia de AWS CLI comandos.

## **disassociate-nat-gateway-address**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-nat-gateway-address`.

### AWS CLI

Para desasociar una dirección IP elástica de una puerta de enlace pública NAT

El siguiente `disassociate-nat-gateway-address` ejemplo desasocia la dirección IP elástica especificada de la puerta de enlace pública NAT especificada.

```
aws ec2 disassociate-nat-gateway-address \
  --nat-gateway-id nat-1234567890abcdef0 \
  --association-ids eipassoc-0f96bdca17EXAMPLE
```

Salida:

```
{
  "NatGatewayId": "nat-1234567890abcdef0",
  "NatGatewayAddresses": [
    {
      "AllocationId": "eipalloc-0be6ecac95EXAMPLE",
      "NetworkInterfaceId": "eni-09cc4b2558794f7f9",
      "PrivateIp": "10.0.0.74",
      "PublicIp": "3.211.231.218",
      "AssociationId": "eipassoc-0f96bdca17EXAMPLE",
      "IsPrimary": false,
      "Status": "disassociating"
    }
  ]
}
```

Para obtener más información, consulta [NAT las pasarelas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [DisassociateNatGatewayAddress](#) la Referencia de AWS CLI comandos.

## **disassociate-route-table**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-route-table`.

### AWS CLI

Para desasociar una tabla de enrutamiento

Este ejemplo desasocia la tabla de rutas especificada de la subred especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 disassociate-route-table --association-id rtbassoc-781d0d1a
```

- Para API obtener más información, consulte la Referencia [DisassociateRouteTable](#) de AWS CLI comandos.

## **disassociate-subnet-cidr-block**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-subnet-cidr-block`.

### AWS CLI

Para desasociar un IPv6 CIDR bloque de una subred

En este ejemplo, se desasocia un IPv6 CIDR bloque de una subred mediante el identificador de asociación del bloque. CIDR

Comando:

```
aws ec2 disassociate-subnet-cidr-block --association-id subnet-cidr-assoc-3aa54053
```

Salida:

```
{  
  "SubnetId": "subnet-5f46ec3b",
```

```

    "Ipv6CidrBlockAssociation": {
      "Ipv6CidrBlock": "2001:db8:1234:1a00::/64",
      "AssociationId": "subnet-cidr-assoc-3aa54053",
      "Ipv6CidrBlockState": {
        "State": "disassociating"
      }
    }
  }
}

```

- Para API obtener más información, consulte la Referencia [DisassociateSubnetCidrBlock](#) de AWS CLI comandos.

## disassociate-transit-gateway-multicast-domain

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-transit-gateway-multicast-domain`.

### AWS CLI

Para desasociar subredes de un dominio de multidifusión

El siguiente `disassociate-transit-gateway-multicast-domain` ejemplo desasocia una subred del dominio de multidifusión especificado.

```

aws ec2 disassociate-transit-gateway-multicast-domain \
  --transit-gateway-attachment-id tgw-attach-070e571cd1EXAMPLE \
  --subnet-id subnet-000de86e3bEXAMPLE \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE

```

Salida:

```

{
  "Associations": [
    {
      "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef7EXAMPLE",
      "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",
      "ResourceId": "vpc-7EXAMPLE",
      "ResourceType": "vpc",
      "Subnets": [
        {
          "SubnetId": "subnet-000de86e3bEXAMPLE",
          "State": "disassociating"
        }
      ]
    }
  ]
}

```

```

    }
  ]
}

```

Para obtener más información, consulte Cómo [trabajar con la multidifusión](#) en la Guía de Transit Gateways '.

- Para API obtener más información, consulte la Referencia [DisassociateTransitGatewayMulticastDomain](#) de AWS CLI comandos.

## disassociate-transit-gateway-route-table

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-transit-gateway-route-table`.

AWS CLI

Para desasociar una tabla de rutas de una pasarela de tránsito de un adjunto de recursos

El siguiente `disassociate-transit-gateway-route-table` ejemplo disocia el adjunto especificado de la tabla de rutas de la puerta de enlace de tránsito.

```

aws ec2 disassociate-transit-gateway-route-table \
  --transit-gateway-route-table-id tgw-rtb-002573ed1eEXAMPLE \
  --transit-gateway-attachment-id tgw-attach-08e0bc912cEXAMPLE

```

Salida:

```

{
  "Association": {
    "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
    "TransitGatewayAttachmentId": "tgw-attach-08e0bc912cEXAMPLE",
    "ResourceId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
    "ResourceType": "direct-connect-gateway",
    "State": "disassociating"
  }
}

```

Para obtener más información, consulte [las tablas de rutas de Transit Gateways](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [DisassociateTransitGatewayRouteTable](#) Referencia de AWS CLI comandos.

## disassociate-vpc-cidr-block

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-vpc-cidr-block`.

### AWS CLI

Para desasociar un IPv6 CIDR bloque de un VPC

En este ejemplo, se desasocia un IPv6 CIDR bloque de un VPC mediante el identificador de asociación del CIDR bloque.

Comando:

```
aws ec2 disassociate-vpc-cidr-block --association-id vpc-cidr-assoc-eca54085
```

Salida:

```
{
  "Ipv6CidrBlockAssociation": {
    "Ipv6CidrBlock": "2001:db8:1234:1a00::/56",
    "AssociationId": "vpc-cidr-assoc-eca54085",
    "Ipv6CidrBlockState": {
      "State": "disassociating"
    }
  },
  "VpcId": "vpc-a034d6c4"
}
```

Para desasociar un IPv4 CIDR bloque de un VPC

En este ejemplo se disocia un IPv4 CIDR bloque de un. VPC

Comando:

```
aws ec2 disassociate-vpc-cidr-block --association-id vpc-cidr-assoc-0287ac6b
```

Salida:



```
{
  "CidrBlockAssociation": {
    "AssociationId": "vpc-cidr-assoc-0287ac6b",
    "CidrBlock": "172.18.0.0/16",
    "CidrBlockState": {
      "State": "disassociating"
    }
  },
  "VpcId": "vpc-27621243"
}
```

- Para API obtener más información, consulte [DisassociateVpcCidrBlock](#) la Referencia de AWS CLI comandos.

## enable-address-transfer

En el siguiente ejemplo de código, se muestra cómo usar `enable-address-transfer`.

### AWS CLI

Para habilitar una transferencia de direcciones IP elásticas

El siguiente `enable-address-transfer` ejemplo permite la transferencia de direcciones IP elásticas para la dirección IP elástica especificada a la cuenta especificada.

```
aws ec2 enable-address-transfer \
  --allocation-id eipalloc-09ad461b0d03f6aaf \
  --transfer-account-id 123456789012
```

Salida:

```
{
  "AddressTransfer": {
    "PublicIp": "100.21.184.216",
    "AllocationId": "eipalloc-09ad461b0d03f6aaf",
    "TransferAccountId": "123456789012",
    "TransferOfferExpirationTimestamp": "2023-02-22T20:51:01.000Z",
    "AddressTransferStatus": "pending"
  }
}
```

Para obtener más información, consulta [Transferir direcciones IP elásticas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [EnableAddressTransfer](#) la Referencia de AWS CLI comandos.

## **enable-aws-network-performance-metric-subscription**

En el siguiente ejemplo de código, se muestra cómo usar `enable-aws-network-performance-metric-subscription`.

### AWS CLI

Para habilitar una suscripción métrica

El siguiente `enable-aws-network-performance-metric-subscription` ejemplo permite monitorear la latencia de red agregada entre las regiones de origen y destino especificadas.

```
aws ec2 enable-aws-network-performance-metric-subscription \
  --source us-east-1 \
  --destination eu-west-1 \
  --metric aggregate-latency \
  --statistic p50
```

Salida:

```
{
  "Output": true
}
```

Para obtener más información, consulte [Administrar las suscripciones](#) en la Guía del usuario de Infrastructure Performance.

- Para API obtener más información, consulte [EnableAwsNetworkPerformanceMetricSubscription](#) la Referencia de AWS CLI comandos.

## **enable-ebs-encryption-by-default**

En el siguiente ejemplo de código, se muestra cómo usar `enable-ebs-encryption-by-default`.

## AWS CLI

Para habilitar el EBS cifrado de forma predeterminada

El siguiente `enable-ebs-encryption-by-default` ejemplo habilita el EBS cifrado de forma predeterminada para su AWS cuenta en la región actual.

```
aws ec2 enable-ebs-encryption-by-default
```

Salida:

```
{
  "EbsEncryptionByDefault": true
}
```

- Para API obtener más información, consulte [EnableEbsEncryptionByDefault](#) la Referencia de AWS CLI comandos.

## `enable-fast-launch`

En el siguiente ejemplo de código, se muestra cómo usar `enable-fast-launch`.

## AWS CLI

Para iniciar el lanzamiento rápido de una imagen

En el siguiente `enable-fast-launch` ejemplo, se inicia el inicio rápido en la instancia especificada AMI y se establece en 6 el número máximo de instancias paralelas que se van a lanzar. El tipo de recurso que se va a utilizar para el aprovisionamiento previo AMI se establece en `snapshot`, que también es el valor predeterminado.

```
aws ec2 enable-fast-launch \
  --image-id ami-01234567890abcdef \
  --max-parallel-launches 6 \
  --resource-type snapshot
```

Salida:

```
{
```

```
"ImageId": "ami-01234567890abcdef",
"ResourceType": "snapshot",
"SnapshotConfiguration": {
  "TargetResourceCount": 10
},
"LaunchTemplate": {},
"MaxParallelLaunches": 6,
"OwnerId": "0123456789123",
"State": "enabling",
"StateTransitionReason": "Client.UserInitiated",
"StateTransitionTime": "2022-01-27T22:16:03.199000+00:00"
}
```

Para obtener más información sobre cómo configurar un Windows AMI para un inicio más rápido, consulte [Configurar su sistema AMI para un inicio más rápido](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [EnableFastLaunch](#) la Referencia de AWS CLI comandos.

## enable-fast-snapshot-restores

En el siguiente ejemplo de código, se muestra cómo usar `enable-fast-snapshot-restores`.

### AWS CLI

Para habilitar la restauración rápida de instantáneas

El siguiente `enable-fast-snapshot-restores` ejemplo permite la restauración rápida de instantáneas para la instantánea especificada en las zonas de disponibilidad especificadas.

```
aws ec2 enable-fast-snapshot-restores \
  --availability-zones us-east-2a us-east-2b \
  --source-snapshot-ids snap-1234567890abcdef0
```

Salida:

```
{
  "Successful": [
    {
      "SnapshotId": "snap-1234567890abcdef0"
    }
  ]
}
```

```

    "AvailabilityZone": "us-east-2a",
    "State": "enabling",
    "StateTransitionReason": "Client.UserInitiated",
    "OwnerId": "123456789012",
    "EnablingTime": "2020-01-25T23:57:49.602Z"
  },
  {
    "SnapshotId": "snap-1234567890abcdef0"
    "AvailabilityZone": "us-east-2b",
    "State": "enabling",
    "StateTransitionReason": "Client.UserInitiated",
    "OwnerId": "123456789012",
    "EnablingTime": "2020-01-25T23:57:49.596Z"
  }
],
"Unsuccessful": []
}

```

- Para API obtener más información, consulte [EnableFastSnapshotRestores](#) la Referencia de AWS CLI comandos.

## enable-image-block-public-access

En el siguiente ejemplo de código, se muestra cómo usar `enable-image-block-public-access`.

### AWS CLI

Para habilitar el bloqueo del acceso público AMIs en la región especificada

El siguiente `enable-image-block-public-access` ejemplo permite bloquear el acceso público AMIs a nivel de cuenta en la región especificada.

```

aws ec2 enable-image-block-public-access \
  --region us-east-1 \
  --image-block-public-access-state block-new-sharing

```

Salida:

```

{
  "ImageBlockPublicAccessState": "block-new-sharing"
}

```

```
}
```

Para obtener más información, consulta [Bloquear el acceso público a tu contenido AMIs](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [EnableImageBlockPublicAccess](#) la Referencia de AWS CLI comandos.

## **enable-image-deprecation**

En el siguiente ejemplo de código, se muestra cómo usar `enable-image-deprecation`.

### AWS CLI

Ejemplo 1: Para dejar en desuso un AMI

En el siguiente `enable-image-deprecation` ejemplo, se desaprueba una AMI en una fecha y hora específicas. Si especificas un valor para los segundos, Amazon EC2 redondea los segundos al minuto más cercano. Debe ser el AMI propietario para poder realizar este procedimiento.

```
aws ec2 enable-image-deprecation \  
  --image-id ami-1234567890abcdef0 \  
  --deprecate-at "2022-10-15T13:17:12.000Z"
```

Salida:

```
{  
  "RequestID": "59dbff89-35bd-4eac-99ed-be587EXAMPLE",  
  "Return": "true"  
}
```

Para obtener más información, consulte [Eliminar un AMI <https://docs.aws.amazon.com/AWS-EC2/latest/UserGuide/ami-deprecate.html>](https://docs.aws.amazon.com/AWS-EC2/latest/UserGuide/ami-deprecate.html) en la Guía del usuario de Amazon. EC2

- Para obtener API más información, consulte la Referencia de comandos. [EnableImageDeprecation](#) AWS CLI

## **enable-image**

En el siguiente ejemplo de código, se muestra cómo usar `enable-image`.

## AWS CLI

Para habilitar un AMI

El siguiente `enable-image` ejemplo habilita lo especificado AMI.

```
aws ec2 enable-image \  
  --image-id ami-1234567890abcdef0
```

Salida:

```
{  
  "Return": "true"  
}
```

Para obtener más información, consulta Cómo [deshabilitar una AMI](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [EnableImage](#) la Referencia de AWS CLI comandos.

## **enable-ipam-organization-admin-account**

En el siguiente ejemplo de código, se muestra cómo usar `enable-ipam-organization-admin-account`.

## AWS CLI

Para integrarse con AWS Organizations y delegar una cuenta de miembro como IPAM cuenta

El siguiente `enable-ipam-organization-admin-account` ejemplo se integra IPAM con AWS Organizations y delega una cuenta de miembro como IPAM cuenta.

```
aws ec2 enable-ipam-organization-admin-account \  
  --delegated-admin-account-id 320805250157
```

Salida:

```
{  
  "Success": true
```

```
}
```

Para obtener más información, consulta [Integrate IPAM with AWS Organizations](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [EnableIpamOrganizationAdminAccount](#) Referencia de AWS CLI comandos.

## **enable-reachability-analyzer-organization-sharing**

En el siguiente ejemplo de código, se muestra cómo usar `enable-reachability-analyzer-organization-sharing`.

### AWS CLI

Para habilitar el acceso confiable al Reachability Analyzer

El siguiente `enable-reachability-analyzer-organization-sharing` ejemplo habilita el acceso confiable a Reachability Analyzer.

```
aws ec2 enable-reachability-analyzer-organization-sharing
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Análisis multicuenta](#) en la Guía del usuario del Reachability Analyzer.

- Para obtener API más información, consulte la Referencia de comandos [EnableReachabilityAnalyzerOrganizationSharing](#).AWS CLI

## **enable-serial-console-access**

En el siguiente ejemplo de código, se muestra cómo usar `enable-serial-console-access`.

### AWS CLI

Para habilitar el acceso de su cuenta a la consola en serie

El siguiente `enable-serial-console-access` ejemplo permite el acceso de la cuenta a la consola serie.



```
aws ec2 enable-serial-console-access
```

Salida:

```
{
  "SerialConsoleAccessEnabled": true
}
```

Para obtener más información, consulta [EC2Serial Console](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [EnableSerialConsoleAccess](#) la Referencia de AWS CLI comandos.

## **enable-snapshot-block-public-access**

En el siguiente ejemplo de código, se muestra cómo usar `enable-snapshot-block-public-access`.

AWS CLI

Para habilitar el bloqueo del acceso público a las instantáneas

El siguiente `enable-snapshot-block-public-access` ejemplo bloquea todo el uso público de sus instantáneas.

```
aws ec2 enable-snapshot-block-public-access \
  --state block-all-sharing
```

Salida:

```
{
  "State": "block-all-sharing"
}
```

Para obtener más información, consulta [Bloquear el acceso público a las instantáneas](#) en la Guía del EBS usuario de Amazon.

- Para API obtener más información, consulte [EnableSnapshotBlockPublicAccess](#) la Referencia de AWS CLI comandos.

## enable-transit-gateway-route-table-propagation

En el siguiente ejemplo de código, se muestra cómo usar `enable-transit-gateway-route-table-propagation`.

### AWS CLI

Para habilitar un adjunto a una pasarela de tránsito para propagar las rutas a la tabla de rutas de propagación especificada

El siguiente `enable-transit-gateway-route-table-propagation` ejemplo permite que el adjunto especificado propague las rutas a la tabla de rutas de propagación especificada.

```
aws ec2 enable-transit-gateway-route-table-propagation \
  --transit-gateway-route-table-id tgw-rtb-0a823edbdeEXAMPLE \
  --transit-gateway-attachment-id tgw-attach-09b52ccdb5EXAMPLE
```

Salida:

```
{
  "Propagation": {
    "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",
    "ResourceId": "vpc-4d7de228",
    "ResourceType": "vpc",
    "TransitGatewayRouteTableId": "tgw-rtb-0a823edbdeEXAMPLE",
    "State": "disabled"
  }
}
```

Para obtener más información, consulte [las tablas de rutas de Transit Gateways](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [EnableTransitGatewayRouteTablePropagation](#) la Referencia de AWS CLI comandos.

## enable-vgw-route-propagation

En el siguiente ejemplo de código, se muestra cómo usar `enable-vgw-route-propagation`.

### AWS CLI

Para habilitar la propagación de rutas

Este ejemplo permite que la puerta de enlace privada virtual especificada propague las rutas estáticas a la tabla de rutas especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 enable-vgw-route-propagation --route-table-id rtb-22574640 --gateway-id vgw-9a4cacf3
```

- Para API obtener más información, consulte [EnableVgwRoutePropagation](#) la Referencia de AWS CLI comandos.

## **enable-volume-io**

En el siguiente ejemplo de código, se muestra cómo usar `enable-volume-io`.

AWS CLI

Para habilitar la E/S de un volumen

Este ejemplo habilita la E/S en el volumen. `vol-1234567890abcdef0`

Comando:

```
aws ec2 enable-volume-io --volume-id vol-1234567890abcdef0
```

Salida:

```
{  
  "Return": true  
}
```

- Para API obtener más información, consulte [EnableVolumeIo](#) la Referencia de AWS CLI comandos.

## **enable-vpc-classic-link-dns-support**

En el siguiente ejemplo de código, se muestra cómo usar `enable-vpc-classic-link-dns-support`.

## AWS CLI

Para habilitar el ClassicLink DNS soporte para un VPC

Este ejemplo habilita ClassicLink DNS el soporte paravpc-88888888.

Comando:

```
aws ec2 enable-vpc-classic-link-dns-support --vpc-id vpc-88888888
```

Salida:

```
{
  "Return": true
}
```

- Para API obtener más información, consulte [EnableVpcClassicLinkDnsSupport](#) la Referencia de AWS CLI comandos.

## enable-vpc-classic-link

En el siguiente ejemplo de código, se muestra cómo usar enable-vpc-classic-link.

## AWS CLI

Para habilitar un VPC para ClassicLink

Este ejemplo habilita vpc-88888888 para. ClassicLink

Comando:

```
aws ec2 enable-vpc-classic-link --vpc-id vpc-88888888
```

Salida:

```
{
  "Return": true
}
```

- Para API obtener más información, consulte la Referencia [EnableVpcClassicLink](#) de AWS CLI comandos.

## export-client-vpn-client-certificate-revocation-list

En el siguiente ejemplo de código, se muestra cómo usar `export-client-vpn-client-certificate-revocation-list`.

### AWS CLI

Para exportar una lista de revocaciones de certificados de cliente

En el siguiente `export-client-vpn-client-certificate-revocation-list` ejemplo, se exporta la lista de revocaciones de certificados de cliente para el terminal de cliente VPN especificado. En este ejemplo, el resultado se devuelve en formato de texto para facilitar su lectura.

```
aws ec2 export-client-vpn-client-certificate-revocation-list \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
  --output text
```

Salida:

```
-----BEGIN X509 CRL-----
MIICiTCcAfiCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAStC0lBTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAd
BgkqhkiG9w0BCQEWEG5vb25lQGFTYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAStC0lBTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAdBgkqhkiG9w0BCQEWEG5vb25lQGFT
YXpvbi5jb20wZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLygVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJ10ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END X509 CRL-----
STATUS      pending
```

Para obtener más información, consulte las [listas de revocación de certificados](#) de AWS cliente en la Guía del VPN administrador de clientes.

- Para API obtener más información, consulte [ExportClientVpnClientCertificateRevocationList](#) la Referencia de AWS CLI comandos.

## export-client-vpn-client-configuration

En el siguiente ejemplo de código, se muestra cómo usar `export-client-vpn-client-configuration`.

### AWS CLI

Para exportar la configuración del cliente

El siguiente `export-client-vpn-client-configuration` ejemplo exporta la configuración del cliente para el VPN punto final del cliente especificado. En este ejemplo, el resultado se devuelve en formato de texto para facilitar su lectura.

```
aws ec2 export-client-vpn-client-configuration \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \  
  --output text
```

Salida:

```
client  
dev tun  
proto udp  
remote cvpn-endpoint-123456789123abcde.prod.clientvpn.ap-south-1.amazonaws.com 443  
remote-random-hostname  
resolv-retry infinite  
nobind  
persist-key  
persist-tun  
remote-cert-tls server  
cipher AES-256-GCM  
verb 3  
<ca>  
-----BEGIN CERTIFICATE-----  
MIICiTCCAFICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAKGA1UEBhMC  
VVMxCzAJBgNVBAgTAlldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6  
b24xFDASBgNVBA5TC0lBTSBDb25zb2x1MRIwEAYDVQQDEw1UZXN0Q21sYWxhZAdB  
BgkqhkiG9w0BCQEWEG5vb251QGFTYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN  
MTIwNDI1MjA0NTIxWjCBiDELMAKGA1UEBhMCVVMxCzAJBgNVBAgTAlldBMRAwDgYD  
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC0lBTSBDb25z  
b2x1MRIwEAYDVQQDEw1UZXN0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGFT  
YXpvbi5jb20wZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ  
21uUSfwfEvySwTc2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T  
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
```

```
Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----
</ca>
reneg-sec 0
```

Para obtener más información, consulte [Client VPN Endpoints](#) en la Guía del VPN administrador de AWS clientes.

- Para API obtener más información, consulte [ExportClientVpnClientConfiguration](#) la Referencia de AWS CLI comandos.

## export-image

En el siguiente ejemplo de código, se muestra cómo usar `export-image`.

### AWS CLI

Para exportar una máquina virtual desde un AMI

El siguiente `export-image` ejemplo exporta AMI lo especificado al bucket especificado en el formato especificado.

```
aws ec2 export-image \
  --image-id ami-1234567890abcdef0 \
  --disk-image-format VMDK \
  --s3-export-location S3Bucket=my-export-bucket,S3Prefix=exports/
```

Salida:

```
{
  "DiskImageFormat": "vmdk",
  "ExportImageTaskId": "export-ami-1234567890abcdef0"
  "ImageId": "ami-1234567890abcdef0",
  "RoleName": "vmimport",
  "Progress": "0",
  "S3ExportLocation": {
    "S3Bucket": "my-export-bucket",
    "S3Prefix": "exports/"
  },
  "Status": "active",
```

```
"StatusMessage": "validating"
}
```

- Para API obtener más información, consulte [ExportImage](#) la Referencia de AWS CLI comandos.

## get-associated-ipv6-pool-cidrs

En el siguiente ejemplo de código, se muestra cómo usar `get-associated-ipv6-pool-cidrs`.

### AWS CLI

Para obtener las asociaciones de un conjunto IPv6 de direcciones

En el siguiente `get-associated-ipv6-pool-cidrs` ejemplo, se obtienen las asociaciones del conjunto de IPv6 direcciones especificado.

```
aws ec2 get-associated-ipv6-pool-cidrs \
  --pool-id ipv6pool-ec2-012345abc12345abc
```

Salida:

```
{
  "Ipv6CidrAssociations": [
    {
      "Ipv6Cidr": "2001:db8:1234:1a00::/56",
      "AssociatedResource": "vpc-111111222222333ab"
    }
  ]
}
```

- Para API obtener más información, consulte [GetAssociatedIpv6 PoolCidrs](#) en la Referencia de AWS CLI comandos.

## get-aws-network-performance-data

En el siguiente ejemplo de código, se muestra cómo usar `get-aws-network-performance-data`.

### AWS CLI

Para obtener datos de rendimiento de la red



El siguiente `get-aws-network-performance-data` ejemplo recupera datos sobre el rendimiento de la red entre las regiones especificadas en el período de tiempo especificado.

```
aws ec2 get-aws-network-performance-data \  
  --start-time 2022-10-26T12:00:00.000Z \  
  --end-time 2022-10-26T12:30:00.000Z \  
  --data-queries Id=my-query,Source=us-east-1,Destination=eu-  
west-1,Metric=aggregate-latency,Statistic=p50,Period=five-minutes
```

Salida:

```
{  
  "DataResponses": [  
    {  
      "Id": "my-query",  
      "Source": "us-east-1",  
      "Destination": "eu-west-1",  
      "Metric": "aggregate-latency",  
      "Statistic": "p50",  
      "Period": "five-minutes",  
      "MetricPoints": [  
        {  
          "StartDate": "2022-10-26T12:00:00+00:00",  
          "EndDate": "2022-10-26T12:05:00+00:00",  
          "Value": 62.44349,  
          "Status": "OK"  
        },  
        {  
          "StartDate": "2022-10-26T12:05:00+00:00",  
          "EndDate": "2022-10-26T12:10:00+00:00",  
          "Value": 62.483498,  
          "Status": "OK"  
        },  
        {  
          "StartDate": "2022-10-26T12:10:00+00:00",  
          "EndDate": "2022-10-26T12:15:00+00:00",  
          "Value": 62.51248,  
          "Status": "OK"  
        },  
        {  
          "StartDate": "2022-10-26T12:15:00+00:00",  
          "EndDate": "2022-10-26T12:20:00+00:00",  
          "Value": 62.635475,  
          "Status": "OK"  
        }  
      ]  
    }  
  ]  
}
```

```

        "Status": "OK"
    },
    {
        "StartDate": "2022-10-26T12:20:00+00:00",
        "EndDate": "2022-10-26T12:25:00+00:00",
        "Value": 62.733974,
        "Status": "OK"
    },
    {
        "StartDate": "2022-10-26T12:25:00+00:00",
        "EndDate": "2022-10-26T12:30:00+00:00",
        "Value": 62.773975,
        "Status": "OK"
    },
    {
        "StartDate": "2022-10-26T12:30:00+00:00",
        "EndDate": "2022-10-26T12:35:00+00:00",
        "Value": 62.75349,
        "Status": "OK"
    }
]
}
]
}

```

Para obtener más información, consulte [Supervisar el rendimiento de la red](#) en la Guía del usuario sobre el rendimiento de la infraestructura.

- Para API obtener más información, consulte [GetAwsNetworkPerformanceData](#) la Referencia de AWS CLI comandos.

## get-capacity-reservation-usage

En el siguiente ejemplo de código, se muestra cómo usar `get-capacity-reservation-usage`.

### AWS CLI

Para ver el uso de reservas de capacidad en todas AWS las cuentas

En el siguiente `get-capacity-reservation-usage` ejemplo, se muestra la información de uso de la reserva de capacidad especificada.

```
aws ec2 get-capacity-reservation-usage \
```

```
--capacity-reservation-id cr-1234abcd56EXAMPLE
```

Salida:

```
{
  "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
  "InstanceUsages": [
    {
      "UsedInstanceCount": 1,
      "AccountId": "123456789012"
    }
  ],
  "AvailableInstanceCount": 4,
  "TotalInstanceCount": 5,
  "State": "active",
  "InstanceType": "t2.medium"
}
```

Para obtener más información, consulte [Visualización del uso de reservas de capacidad compartida](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [GetCapacityReservationUsage](#) la Referencia de AWS CLI comandos.

## get-coip-pool-usage

En el siguiente ejemplo de código, se muestra cómo usar `get-coip-pool-usage`.

AWS CLI

Para obtener información sobre el uso del conjunto de direcciones IP propiedad del cliente

En el siguiente `get-coip-pool-usage` ejemplo, se obtienen los detalles de uso del conjunto de direcciones IP propiedad del cliente especificado.

```
aws ec2 get-coip-pool-usage \
  --pool-id ipv4pool-coip-123a45678bEXAMPLE
```

Salida:

```
{
```

```

    "CoipPoolId": "ipv4pool-coip-123a45678bEXAMPLE",
    "CoipAddressUsages": [
      {
        "CoIp": "0.0.0.0"
      },
      {
        "AllocationId": "eipalloc-123ab45c6dEXAMPLE",
        "AwsAccountId": "123456789012",
        "CoIp": "0.0.0.0"
      },
      {
        "AllocationId": "eipalloc-123ab45c6dEXAMPLE",
        "AwsAccountId": "123456789111",
        "CoIp": "0.0.0.0"
      }
    ],
    "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE"
  }

```

Para obtener más información, consulte [Direcciones IP propiedad del cliente](#) en la Guía del usuario de AWS .

- Para API obtener más información, consulte la Referencia [GetCoipPoolUsage](#) de AWS CLI comandos.

## get-console-output

En el siguiente ejemplo de código, se muestra cómo usar `get-console-output`.

### AWS CLI

Ejemplo 1: Para obtener el resultado de la consola

En el siguiente `get-console-output` ejemplo, se obtiene el resultado de la consola para la instancia de Linux especificada.

```

aws ec2 get-console-output \
  --instance-id i-1234567890abcdef0

```

Salida:

```
{
```

```

    "InstanceId": "i-1234567890abcdef0",
    "Timestamp": "2013-07-25T21:23:53.000Z",
    "Output": "...
}

```

Para obtener más información, consulta la [salida de la consola de instancias](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para obtener el último resultado de la consola

En el siguiente `get-console-output` ejemplo, se obtiene el último resultado de la consola para la instancia de Linux especificada.

```

aws ec2 get-console-output \
  --instance-id i-1234567890abcdef0 \
  --latest \
  --output text

```

Salida:

```

i-1234567890abcdef0 [ 0.000000] Command line: root=LABEL=/ console=tty1
console=ttyS0 selinux=0 nvme_core.io_timeout=4294967295
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x001: 'x87 floating point
registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x002: 'SSE registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x004: 'AVX registers'
...
Cloud-init v. 0.7.6 finished at Wed, 09 May 2018 19:01:13 +0000. Datasource
DataSourceEc2. Up 21.50 seconds
Amazon Linux AMI release 2018.03
Kernel 4.14.26-46.32.amzn1.x

```

Para obtener más información, consulta la [salida de la consola de instancias](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [GetConsoleOutput](#) la Referencia de AWS CLI comandos.

## get-console-screenshot

En el siguiente ejemplo de código, se muestra cómo usar `get-console-screenshot`.

## AWS CLI

Para recuperar una captura de pantalla de una instancia en ejecución

En el siguiente `get-console-screenshot` ejemplo, se recupera una captura de pantalla de la instancia especificada en formato.jpg. La captura de pantalla se devuelve como una cadena codificada en Base64.

```
aws ec2 get-console-screenshot \  
  --instance-id i-1234567890abcdef0
```

Salida:

```
{  
  "ImageData": "997987/8kgj49ikjhewkww0008084EXAMPLE",  
  "InstanceId": "i-1234567890abcdef0"  
}
```

- Para API obtener más información, consulte la Referencia de [GetConsoleScreenshot](#) comandos AWS CLI .

## get-default-credit-specification

En el siguiente ejemplo de código, se muestra cómo usar `get-default-credit-specification`.

## AWS CLI

Para describir la opción de crédito predeterminada

En el siguiente `get-default-credit-specification` ejemplo, se describe la opción de crédito predeterminada para las instancias T2.

```
aws ec2 get-default-credit-specification \  
  --instance-family t2
```

Salida:

```
{  
  "InstanceFamilyCreditSpecification": {  
    "InstanceFamily": "t2",
```

```
    "CpuCredits": "standard"  
  }  
}
```

- Para API obtener más información, consulte [GetDefaultCreditSpecification](#) la Referencia de AWS CLI comandos.

## get-ebs-default-kms-key-id

En el siguiente ejemplo de código, se muestra cómo usar `get-ebs-default-kms-key-id`.

### AWS CLI

Para describir su EBS cifrado predeterminado CMK

En el siguiente `get-ebs-default-kms-key-id` ejemplo, se describe el EBS cifrado predeterminado CMK de tu AWS cuenta.

```
aws ec2 get-ebs-default-kms-key-id
```

El resultado muestra el EBS cifrado predeterminado CMK, que se AWS administra CMK con el alias `alias/aws/ebs`.

```
{  
  "KmsKeyId": "alias/aws/ebs"  
}
```

El siguiente resultado muestra una configuración CMK de EBS cifrado personalizada.

```
{  
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/0ea3fef3-80a7-4778-9d8c-1c0c6EXAMPLE"  
}
```

- Para API obtener más información, consulte [GetEbsDefaultKmsKeyId](#) la Referencia de AWS CLI comandos.

## get-ebs-encryption-by-default

En el siguiente ejemplo de código, se muestra cómo usar `get-ebs-encryption-by-default`.

## AWS CLI

Para describir si el EBS cifrado está activado de forma predeterminada

El siguiente `get-ebs-encryption-by-default` ejemplo indica si el EBS cifrado está activado de forma predeterminada para su AWS cuenta en la región actual.

```
aws ec2 get-ebs-encryption-by-default
```

El siguiente resultado indica que el EBS cifrado está deshabilitado de forma predeterminada.

```
{
  "EbsEncryptionByDefault": false
}
```

El siguiente resultado indica que el EBS cifrado está activado de forma predeterminada.

```
{
  "EbsEncryptionByDefault": true
}
```

- Para API obtener más información, consulte [GetEbsEncryptionByDefault](#) la Referencia de AWS CLI comandos.

## `get-flow-logs-integration-template`

En el siguiente ejemplo de código, se muestra cómo usar `get-flow-logs-integration-template`.

## AWS CLI

Para crear una CloudFormation plantilla para automatizar la integración de los registros de VPC flujo con Amazon Athena

Los siguientes `get-flow-logs-integration-template` ejemplos crean una CloudFormation plantilla para automatizar la integración de los registros de VPC flujo con Amazon Athena.

Linux:

```
aws ec2 get-flow-logs-integration-template \
  --flow-log-id fl-1234567890abcdef0 \
```



```

--config-delivery-s3-destination-arn arn:aws:s3::DOC-EXAMPLE-BUCKET \
--integrate-services
AthenaIntegrations='[{IntegrationResultS3DestinationArn=arn:aws:s3::DOC-EXAMPLE-
BUCKET,PartitionLoadFrequency=none,PartitionStartDate=2021-07-21T00:40:00,PartitionEndDate=2
}{IntegrationResultS3DestinationArn=arn:aws:s3::DOC-EXAMPLE-
BUCKET,PartitionLoadFrequency=none,PartitionStartDate=2021-07-21T00:40:00,PartitionEndDate=2

```

Windows:

```

aws ec2 get-flow-logs-integration-template ^
--flow-log-id fl-1234567890abcdef0 ^
--config-delivery-s3-destination-arn arn:aws:s3::DOC-EXAMPLE-BUCKET ^
--integrate-
services AthenaIntegrations=[{IntegrationResultS3DestinationArn=arn:aws:s3::DOC-
EXAMPLE-
BUCKET,PartitionLoadFrequency=none,PartitionStartDate=2021-07-21T00:40:00,PartitionEndDate=2
}{IntegrationResultS3DestinationArn=arn:aws:s3::DOC-EXAMPLE-
BUCKET,PartitionLoadFrequency=none,PartitionStartDate=2021-07-21T00:40:00,PartitionEndDate=2


```

Salida:

```

{
  "Result": "https://DOC-EXAMPLE-BUCKET.s3.us-east-2.amazonaws.com/
VPCFlowLogsIntegrationTemplate_fl-1234567890abcdef0_Wed%20Jul
%2021%2000%3A57%3A56%20UTC%202021.yml"
}

```

Para obtener información sobre el uso de CloudFormation plantillas, consulte [Trabajar con AWS CloudFormation plantillas](#) en la Guía del AWS CloudFormation usuario.

Para obtener información sobre el uso de Amazon Athena y los registros de flujo, consulte [Consulta los registros de flujo con Amazon Athena](#) en la Guía del usuario de Amazon Virtual Private Cloud.

- Para API obtener más información, consulte la Referencia [GetFlowLogsIntegrationTemplate](#) de AWS CLI comandos.

## get-groups-for-capacity-reservation

En el siguiente ejemplo de código, se muestra cómo usar `get-groups-for-capacity-reservation`.

## AWS CLI

Para enumerar los grupos de recursos con una reserva de capacidad

En el siguiente `get-groups-for-capacity-reservation` ejemplo, se enumeran los grupos de recursos a los que se agregó la reserva de capacidad especificada.

```
aws ec2 get-groups-for-capacity-reservation \
  --capacity-reservation-id cr-1234abcd56EXAMPLE
```

Salida:

```
{
  "CapacityReservationsGroup": [
    {
      "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/my-
resource-group",
      "OwnerId": "123456789012"
    }
  ]
}
```

Para obtener más información, consulte Cómo [trabajar con reservas de capacidad](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [GetGroupsForCapacityReservation](#) la Referencia de AWS CLI comandos.

## `get-host-reservation-purchase-preview`

En el siguiente ejemplo de código, se muestra cómo usar `get-host-reservation-purchase-preview`.

### AWS CLI

Para obtener una vista previa de la compra de una reserva de anfitrión dedicada

En este ejemplo, se proporciona una vista previa de los costes de una reserva de anfitrión dedicado específica para el anfitrión dedicado especificado en su cuenta.

Comando:

```
aws ec2 get-host-reservation-purchase-preview --offering-id hro-03f707bf363b6b324 --  
host-id-set h-013abcd2a00cbd123
```

Salida:

```
{  
  "TotalHourlyPrice": "1.499",  
  "Purchase": [  
    {  
      "HourlyPrice": "1.499",  
      "InstanceFamily": "m4",  
      "PaymentOption": "NoUpfront",  
      "HostIdSet": [  
        "h-013abcd2a00cbd123"  
      ],  
      "UpfrontPrice": "0.000",  
      "Duration": 31536000  
    }  
  ],  
  "TotalUpfrontPrice": "0.000"  
}
```

- Para API obtener más información, consulte [GetHostReservationPurchasePreview](#) la Referencia de AWS CLI comandos.

## get-image-block-public-access-state

En el siguiente ejemplo de código, se muestra cómo usar `get-image-block-public-access-state`.

AWS CLI

Para obtener el estado del bloqueo de acceso público AMIs en la región especificada

En el siguiente `get-image-block-public-access-state` ejemplo, se obtiene el estado de bloqueo del acceso público AMIs a nivel de cuenta en la región especificada.

```
aws ec2 get-image-block-public-access-state \  
  --region us-east-1
```

Salida:

```
{
  "ImageBlockPublicAccessState": "block-new-sharing"
}
```

Para obtener más información, consulta [Bloquear el acceso público a tu contenido AMIs](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [GetImageBlockPublicAccessState](#) la Referencia de AWS CLI comandos.

## get-instance-types-from-instance-requirements

En el siguiente ejemplo de código, se muestra cómo usar `get-instance-types-from-instance-requirements`.

### AWS CLI

Para obtener una vista previa de los tipos de instancias que coinciden con los atributos especificados

En el siguiente `get-instance-types-from-instance-requirements` ejemplo, se genera primero una lista de todos los atributos posibles que se pueden especificar mediante el `--generate-cli-skeleton` parámetro y se guarda la lista en un JSON archivo. A continuación, el JSON archivo se utiliza para personalizar los atributos para obtener una vista previa de los tipos de instancias coincidentes.

Para generar todos los atributos posibles y guardar el resultado directamente en un JSON archivo, usa el siguiente comando.

```
aws ec2 get-instance-types-from-instance-requirements \
  --region us-east-1 \
  --generate-cli-skeleton input > attributes.json
```

Salida:

```
{
  "DryRun": true,
  "ArchitectureTypes": [
    "x86_64_mac"
  ],
  "VirtualizationTypes": [
```

```
    "paravirtual"
  ],
  "InstanceRequirements": {
    "VCpuCount": {
      "Min": 0,
      "Max": 0
    },
    "MemoryMiB": {
      "Min": 0,
      "Max": 0
    },
    "CpuManufacturers": [
      "intel"
    ],
    "MemoryGiBPerVCpu": {
      "Min": 0.0,
      "Max": 0.0
    },
    "ExcludedInstanceTypes": [
      ""
    ],
    "InstanceGenerations": [
      "current"
    ],
    "SpotMaxPricePercentageOverLowestPrice": 0,
    "OnDemandMaxPricePercentageOverLowestPrice": 0,
    "BareMetal": "included",
    "BurstablePerformance": "excluded",
    "RequireHibernateSupport": true,
    "NetworkInterfaceCount": {
      "Min": 0,
      "Max": 0
    },
    "LocalStorage": "required",
    "LocalStorageTypes": [
      "hdd"
    ],
    "TotalLocalStorageGB": {
      "Min": 0.0,
      "Max": 0.0
    },
    "BaselineEbsBandwidthMbps": {
      "Min": 0,
      "Max": 0
    }
  }
}
```

```

    },
    "AcceleratorTypes": [
      "inference"
    ],
    "AcceleratorCount": {
      "Min": 0,
      "Max": 0
    },
    "AcceleratorManufacturers": [
      "xilinx"
    ],
    "AcceleratorNames": [
      "t4"
    ],
    "AcceleratorTotalMemoryMiB": {
      "Min": 0,
      "Max": 0
    }
  },
  "MaxResults": 0,
  "NextToken": ""
}

```

Configure el archivo JSON. Debe proporcionar valores para `ArchitectureTypes`, `VirtualizationTypes`, `VCpuCount` y `MemoryMiB`. Puede omitir los demás atributos. Si se omite, se utilizan los valores por defecto. Para obtener una descripción de cada atributo y sus valores predeterminados, consulte `get-instance-types-from -instance-requirements < latest/reference/ec2/ -instance-requirements.html >`. <https://docs.aws.amazon.com/cli/get-instance-types-from>

Obtenga una vista previa de `attributes.json` los tipos de instancias que tienen los atributos especificados en. Especifique el nombre y la ruta JSON del archivo mediante el `--cli-input-json` parámetro. En la siguiente solicitud, el resultado tiene el formato de una tabla.

```

aws ec2 get-instance-types-from-instance-requirements \
  --cli-input-json file://attributes.json \
  --output table

```

Contenido del archivo `attributes.json`:

```
{
```

```

    "ArchitectureTypes": [
      "x86_64"
    ],
    "VirtualizationTypes": [
      "hvm"
    ],
    "InstanceRequirements": {
      "VCpuCount": {
        "Min": 4,
        "Max": 6
      },
      "MemoryMiB": {
        "Min": 2048
      },
      "InstanceGenerations": [
        "current"
      ]
    }
  }
}

```

Salida:

```

-----
|GetInstanceTypesFromInstanceRequirements|
+-----+
||           InstanceTypes           ||
|+-----+|
||           InstanceType           ||
|+-----+|
|| c4.xlarge                         ||
|| c5.xlarge                         ||
|| c5a.xlarge                       ||
|| c5ad.xlarge                      ||
|| c5d.xlarge                       ||
|| c5n.xlarge                       ||
|| d2.xlarge                        ||
...

```

Para obtener más información sobre la selección de tipos de instancia basada en atributos, consulta [Cómo funciona la selección de tipos de instancia basada en atributos en la Guía del usuario](#) de Amazon. EC2

- Para obtener API más información, consulte la Referencia de comandos [GetInstanceTypesFromInstanceRequirements](#).AWS CLI

## get-instance-uefi-data

En el siguiente ejemplo de código, se muestra cómo usar `get-instance-uefi-data`.

AWS CLI

Para recuperar UEFI datos de una instancia

En el siguiente `get-instance-uefi-data` ejemplo, se recuperan UEFI los datos de una instancia. Si la salida está vacía, la instancia no contiene UEFI datos.

```
aws ec2 get-instance-uefi-data \  
--instance-id i-0123456789example
```

Salida:

```
{  
  "InstanceId": "i-0123456789example",  
  "UefiData": "QU1aTlVFRkkf+uLXAAAAAHj5a7fZ9+3dBzxXb/.  
<snipped>  
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAD4L/J/A0Dshho="
```

Para obtener más información, consulta [UEFI Secure Boot](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [GetInstanceUefiData](#) la Referencia de AWS CLI comandos.

## get-ipam-address-history

En el siguiente ejemplo de código, se muestra cómo usar `get-ipam-address-history`.

AWS CLI

Para obtener el historial de un CIDR

En el siguiente `get-ipam-address-history` ejemplo se obtiene el historial de un CIDR.



(Linux):

```
aws ec2 get-ipam-address-history \
  --cidr 10.0.0.0/16 \
  --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 \
  --start-time 2021-12-08T01:00:00.000Z \
  --end-time 2021-12-10T01:00:00.000Z
```

(Windows):

```
aws ec2 get-ipam-address-history ^
  --cidr 10.0.0.0/16 ^
  --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 ^
  --start-time 2021-12-08T01:00:00.000Z ^
  --end-time 2021-12-10T01:00:00.000Z
```

Salida:

```
{
  "HistoryRecords": [
    {
      "ResourceOwnerId": "123456789012",
      "ResourceRegion": "us-west-1",
      "ResourceType": "vpc",
      "ResourceId": "vpc-06cbefa9ee907e1c0",
      "ResourceCidr": "10.0.0.0/16",
      "ResourceName": "Demo",
      "ResourceComplianceStatus": "unmanaged",
      "ResourceOverlapStatus": "overlapping",
      "VpcId": "vpc-06cbefa9ee907e1c0",
      "SampledStartTime": "2021-12-08T19:54:57.675000+00:00"
    },
    {
      "ResourceOwnerId": "123456789012",
      "ResourceRegion": "us-east-2",
      "ResourceType": "vpc",
      "ResourceId": "vpc-042702f474812c9ad",
      "ResourceCidr": "10.0.0.0/16",
      "ResourceName": "test",
      "ResourceComplianceStatus": "unmanaged",
      "ResourceOverlapStatus": "overlapping",
      "VpcId": "vpc-042702f474812c9ad",
      "SampledStartTime": "2021-12-08T19:54:59.019000+00:00"
    }
  ]
}
```

```
    },  
    {  
      "ResourceOwnerId": "123456789012",  
      "ResourceRegion": "us-east-2",  
      "ResourceType": "vpc",  
      "ResourceId": "vpc-042b8a44f64267d67",  
      "ResourceCidr": "10.0.0.0/16",  
      "ResourceName": "tester",  
      "ResourceComplianceStatus": "unmanaged",  
      "ResourceOverlapStatus": "overlapping",  
      "VpcId": "vpc-042b8a44f64267d67",  
      "SampledStartTime": "2021-12-08T19:54:59.019000+00:00"  
    }  
  ]  
}
```

Para obtener más información, consulta [Ver el historial de direcciones IP](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [GetIpamAddressHistory](#) la Referencia de AWS CLI comandos.

## get-ipam-discovered-accounts

En el siguiente ejemplo de código, se muestra cómo usar `get-ipam-discovered-accounts`.

### AWS CLI

Para ver las cuentas descubiertas por un IPAM

En este escenario, eres un administrador IPAM delegado que quiere ver AWS las cuentas que poseen los recursos que IPAM está descubriendo.

`--discovery-region` Es la región IPAM operativa en la que desea ver los estados de las cuentas monitoreadas. Por ejemplo, si tiene tres regiones IPAM operativas, puede que desee realizar esta solicitud tres veces para ver las marcas horarias específicas de la detección en cada una de esas regiones concretas.

En el siguiente `get-ipam-discovered-accounts` ejemplo, se enumeran las AWS cuentas que poseen los recursos que IPAM está descubriendo.

```
aws ec2 get-ipam-discovered-accounts \
```

```
--ipam-resource-discovery-id ipam-res-disco-0365d2977fc1672fe \  
--discovery-region us-east-1
```

Salida:

```
{  
  "IpamDiscoveredAccounts": [  
    {  
      "AccountId": "149977607591",  
      "DiscoveryRegion": "us-east-1",  
      "LastAttemptedDiscoveryTime": "2024-02-09T19:04:31.379000+00:00",  
      "LastSuccessfulDiscoveryTime": "2024-02-09T19:04:31.379000+00:00"  
    }  
  ]  
}
```

Para obtener más información, consulta la [sección Integración IPAM con cuentas ajenas a tu organización](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [GetIpamDiscoveredAccounts](#) la Referencia de AWS CLI comandos.

## get-ipam-discovered-public-addresses

En el siguiente ejemplo de código, se muestra cómo usar `get-ipam-discovered-public-addresses`.

AWS CLI

Para ver las direcciones IP públicas descubiertas

En este ejemplo, usted es un administrador IPAM delegado y desea ver las direcciones IP de los recursos detectados por IPAM él. Puede obtener el ID de descubrimiento de recursos con [describe-ipam-resource-discoveries](#).

El siguiente `get-ipam-discovered-public-addresses` ejemplo muestra las direcciones IP públicas descubiertas para la detección de un recurso.

```
aws ec2 get-ipam-discovered-public-addresses \  
  --ipam-resource-discovery-id ipam-res-disco-0f4ef577a9f37a162 \  
  --address-region us-east-1 \  
  --region us-east-1
```

## Salida:

```
{
  "IpamDiscoveredPublicAddresses": [
    {
      "IpamResourceDiscoveryId": "ipam-res-disco-0f4ef577a9f37a162",
      "AddressRegion": "us-east-1",
      "Address": "54.208.155.7",
      "AddressOwnerId": "320805250157",
      "AssociationStatus": "associated",
      "AddressType": "ec2-public-ip",
      "VpcId": "vpc-073b294916198ce49",
      "SubnetId": "subnet-0b6c8a8839e9a4f15",
      "NetworkInterfaceId": "eni-081c446b5284a5e06",
      "NetworkInterfaceDescription": "",
      "InstanceId": "i-07459a6fca5b35823",
      "Tags": {},
      "NetworkBorderGroup": "us-east-1c",
      "SecurityGroups": [
        {
          "GroupName": "launch-wizard-2",
          "GroupId": "sg-0a489dd6a65c244ce"
        }
      ],
      "SampleTime": "2024-04-05T15:13:59.228000+00:00"
    },
    {
      "IpamResourceDiscoveryId": "ipam-res-disco-0f4ef577a9f37a162",
      "AddressRegion": "us-east-1",
      "Address": "44.201.251.218",
      "AddressOwnerId": "470889052923",
      "AssociationStatus": "associated",
      "AddressType": "ec2-public-ip",
      "VpcId": "vpc-6c31a611",
      "SubnetId": "subnet-062f47608b99834b1",
      "NetworkInterfaceId": "eni-024845359c2c3ae9b",
      "NetworkInterfaceDescription": "",
      "InstanceId": "i-04ef786d9c4e03f41",
      "Tags": {},
      "NetworkBorderGroup": "us-east-1a",
      "SecurityGroups": [
        {
          "GroupName": "launch-wizard-32",
          "GroupId": "sg-0ed1a426e96a68374"
        }
      ]
    }
  ]
}
```

```

    }
  ],
  "SampleTime": "2024-04-05T15:13:59.145000+00:00"
}
}

```

Para obtener más información, consulta [Ver información sobre la IP pública](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [GetIpamDiscoveredPublicAddresses](#) la Referencia de AWS CLI comandos.

## get-ipam-discovered-resource-cidrs

En el siguiente ejemplo de código, se muestra cómo usar `get-ipam-discovered-resource-cidrs`.

### AWS CLI

Para ver la dirección IP CIDRs detectada por un IPAM

En este ejemplo, eres un administrador IPAM delegado que desea ver los detalles relacionados con la dirección IP de CIDRs los recursos que IPAM está descubriendo.

Para completar esta solicitud, sigue estos pasos:

La detección de recursos que elija debe estar asociada a la IPAM. `--resource-region` Es la AWS región en la que se creó el recurso.

En el siguiente `get-ipam-discovered-resource-cidrs` ejemplo, se enumeran las direcciones IP de los recursos que IPAM está descubriendo.

```

aws ec2 get-ipam-discovered-resource-cidrs \
  --ipam-resource-discovery-id ipam-res-disco-0365d2977fc1672fe \
  --resource-region us-east-1

```

Salida:

```

{
  {
    "IpamDiscoveredResourceCidrs": [

```

```
{
  "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
  "ResourceRegion": "us-east-1",
  "ResourceId": "vpc-0c974c95ca7ceef4a",
  "ResourceOwnerId": "149977607591",
  "ResourceCidr": "172.31.0.0/16",
  "ResourceType": "vpc",
  "ResourceTags": [],
  "IpUsage": 0.375,
  "VpcId": "vpc-0c974c95ca7ceef4a",
  "SampleTime": "2024-02-09T19:15:16.529000+00:00"
},
{
  "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
  "ResourceRegion": "us-east-1",
  "ResourceId": "subnet-07fe028119082a8c1",
  "ResourceOwnerId": "149977607591",
  "ResourceCidr": "172.31.0.0/20",
  "ResourceType": "subnet",
  "ResourceTags": [],
  "IpUsage": 0.0012,
  "VpcId": "vpc-0c974c95ca7ceef4a",
  "SampleTime": "2024-02-09T19:15:16.529000+00:00"
},
{
  "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
  "ResourceRegion": "us-east-1",
  "ResourceId": "subnet-0a96893763984cc4e",
  "ResourceOwnerId": "149977607591",
  "ResourceCidr": "172.31.64.0/20",
  "ResourceType": "subnet",
  "ResourceTags": [],
  "IpUsage": 0.0012,
  "VpcId": "vpc-0c974c95ca7ceef4a",
  "SampleTime": "2024-02-09T19:15:16.529000+00:00"
}
}
```

Para obtener más información, consulta [Supervisar el CIDR uso por recurso](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [GetIpamDiscoveredResourceCidrs](#) la Referencia de AWS CLI comandos.

## get-ipam-pool-allocations

En el siguiente ejemplo de código, se muestra cómo usar `get-ipam-pool-allocations`.

### AWS CLI

Para obtener lo CIDRs asignado de un IPAM grupo

En el siguiente `get-ipam-pool-allocations` ejemplo, se obtiene lo CIDRs asignado de un IPAM grupo.

(Linux):

```
aws ec2 get-ipam-pool-allocations \  
  --ipam-pool-id ipam-pool-0533048da7d823723 \  
  --filters Name=ipam-pool-allocation-id,Values=ipam-pool-alloc-0e6186d73999e47389266a5d6991e6220
```

(Windows):

```
aws ec2 get-ipam-pool-allocations ^  
  --ipam-pool-id ipam-pool-0533048da7d823723 ^  
  --filters Name=ipam-pool-allocation-id,Values=ipam-pool-alloc-0e6186d73999e47389266a5d6991e6220
```

Salida:

```
{  
  "IpamPoolAllocations": [  
    {  
      "Cidr": "10.0.0.0/16",  
      "IpamPoolAllocationId": "ipam-pool-alloc-0e6186d73999e47389266a5d6991e6220",  
      "ResourceType": "custom",  
      "ResourceOwner": "123456789012"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [GetIpamPoolAllocations](#) la Referencia de AWS CLI comandos.

## get-ipam-pool-cidrs

En el siguiente ejemplo de código, se muestra cómo usar `get-ipam-pool-cidrs`.

### AWS CLI

Para llevar lo CIDRs aprovisionado a un grupo IPAM

En el siguiente `get-ipam-pool-cidrs` ejemplo, se transfiere lo CIDRs aprovisionado a un IPAM grupo.

(Linux):

```
aws ec2 get-ipam-pool-cidrs \  
  --ipam-pool-id ipam-pool-0533048da7d823723 \  
  --filters 'Name=cidr,Values=10.*'
```

(Windows):

```
aws ec2 get-ipam-pool-cidrs ^  
  --ipam-pool-id ipam-pool-0533048da7d823723 ^  
  --filters Name=cidr,Values=10.*
```

Salida:

```
{  
  "IpamPoolCidr": {  
    "Cidr": "10.0.0.0/24",  
    "State": "provisioned"  
  }  
}
```

- Para API obtener más información, consulte [GetIpamPoolCidrs](#) la Referencia de AWS CLI comandos.

## get-ipam-resource-cidrs

En el siguiente ejemplo de código, se muestra cómo usar `get-ipam-resource-cidrs`.

### AWS CLI

Para obtener lo CIDRs asignado a un recurso



En el siguiente `get-ipam-resource-cidrs` ejemplo, se obtiene lo CIDRs asignado a un recurso.

(Linux):

```
aws ec2 get-ipam-resource-cidrs \  
  --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 \  
  --filters Name=management-state,Values=unmanaged
```

(Windows):

```
aws ec2 get-ipam-resource-cidrs ^  
  --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 ^  
  --filters Name=management-state,Values=unmanaged
```

Salida:

```
{  
  "IpamResourceCidrs": [  
    {  
      "IpamId": "ipam-08440e7a3acde3908",  
      "IpamScopeId": "ipam-scope-02fc38cd4c48e7d38",  
      "ResourceRegion": "us-east-2",  
      "ResourceOwnerId": "123456789012",  
      "ResourceId": "vpc-621b8709",  
      "ResourceName": "Default AWS VPC",  
      "ResourceCidr": "172.33.0.0/16",  
      "ResourceType": "vpc",  
      "ResourceTags": [  
        {  
          "Key": "Environment",  
          "Value": "Test"  
        },  
        {  
          "Key": "Name",  
          "Value": "Default AWS VPC"  
        }  
      ],  
      "IpUsage": 0.0039,  
      "ComplianceStatus": "unmanaged",  
      "ManagementState": "unmanaged",  
      "OverlapStatus": "nonoverlapping",  
      "VpcId": "vpc-621b8709"  
    }  
  ]  
}
```

```

    }
  ]
}

```

Para obtener más información, consulta [Supervisar el CIDR uso por recurso](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [GetIpamResourceCidrs](#) la Referencia de AWS CLI comandos.

## get-launch-template-data

En el siguiente ejemplo de código, se muestra cómo usar `get-launch-template-data`.

### AWS CLI

Para obtener datos de instancia para una plantilla de lanzamiento

En este ejemplo, se obtienen datos sobre la instancia especificada y se usa la `--query` opción para devolver el contenido `LaunchTemplateData`. Puede usar la salida como base para crear una nueva plantilla de inicialización o una versión.

Comando:

```
aws ec2 get-launch-template-data --instance-id i-0123d646e8048babc --query
'LaunchTemplateData'
```

Salida:

```
{
  "Monitoring": {},
  "ImageId": "ami-8c1be5f6",
  "BlockDeviceMappings": [
    {
      "DeviceName": "/dev/xvda",
      "Ebs": {
        "DeleteOnTermination": true
      }
    }
  ],
  "EbsOptimized": false,
  "Placement": {
```

```

    "Tenancy": "default",
    "GroupName": "",
    "AvailabilityZone": "us-east-1a"
  },
  "InstanceType": "t2.micro",
  "NetworkInterfaces": [
    {
      "Description": "",
      "NetworkInterfaceId": "eni-35306abc",
      "PrivateIpAddresses": [
        {
          "Primary": true,
          "PrivateIpAddress": "10.0.0.72"
        }
      ],
      "SubnetId": "subnet-7b16de0c",
      "Groups": [
        "sg-7c227019"
      ],
      "Ipv6Addresses": [
        {
          "Ipv6Address": "2001:db8:1234:1a00::123"
        }
      ],
      "PrivateIpAddress": "10.0.0.72"
    }
  ]
}

```

- Para API obtener más información, consulte [GetLaunchTemplateData](#) la Referencia de AWS CLI comandos.

## get-managed-prefix-list-associations

En el siguiente ejemplo de código, se muestra cómo usar `get-managed-prefix-list-associations`.

### AWS CLI

Para obtener las asociaciones de listas de prefijos

En el siguiente `get-managed-prefix-list-associations` ejemplo, se obtienen los recursos asociados a la lista de prefijos especificada.

```
aws ec2 get-managed-prefix-list-associations \
  --prefix-list-id pl-0123456abcabc1
```

Salida:

```
{
  "PrefixListAssociations": [
    {
      "ResourceId": "sg-0abc123456abc12345",
      "ResourceOwner": "123456789012"
    }
  ]
}
```

Para obtener más información, consulta [Listas de prefijos gestionadas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [GetManagedPrefixListAssociations](#) la Referencia de AWS CLI comandos.

## get-managed-prefix-list-entries

En el siguiente ejemplo de código, se muestra cómo usar `get-managed-prefix-list-entries`.

### AWS CLI

Para obtener las entradas de una lista de prefijos

A continuación `get-managed-prefix-list-entries` se obtienen las entradas de la lista de prefijos especificada.

```
aws ec2 get-managed-prefix-list-entries \
  --prefix-list-id pl-0123456abcabc1
```

Salida:

```
{
  "Entries": [
    {
      "Cidr": "10.0.0.0/16",
      "Description": "vpc-a"
    },
  ],
}
```

```
{
  "Cidr": "10.2.0.0/16",
  "Description": "vpc-b"
}
]
```

Para obtener más información, consulta [Listas de prefijos gestionadas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [GetManagedPrefixListEntries](#) la Referencia de AWS CLI comandos.

## get-network-insights-access-scope-analysis-findings

En el siguiente ejemplo de código, se muestra cómo usar `get-network-insights-access-scope-analysis-findings`.

### AWS CLI

Para obtener los resultados de Network Insights, acceda al análisis del alcance

En el siguiente `get-network-insights-access-scope-analysis-findings` ejemplo, se obtienen los resultados del análisis de alcance seleccionado en su AWS cuenta.

```
aws ec2 get-network-insights-access-scope-analysis-findings \
  --region us-east-1 \
  --network-insights-access-scope-analysis-id nisa \
  --nis-123456789111
```

Salida:

```
{
  "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789222",
  "AnalysisFindings": [
    {
      "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789222",
      "NetworkInsightsAccessScopeId": "nis-123456789111",
      "FindingComponents": [
        {
          "SequenceNumber": 1,
          "Component": {
```

```

        "Id": "eni-02e3d42d5cceca67d",
        "Arn": "arn:aws:ec2:us-east-1:936459623503:network-
interface/eni-02e3d32d9cceca17d"
    },
    "OutboundHeader": {
        "DestinationAddresses": [
            "0.0.0.0/5",
            "11.0.0.0/8",
            "12.0.0.0/6",
            "128.0.0.0/3",
            "16.0.0.0/4",
            "160.0.0.0/5",
            "168.0.0.0/6",
            "172.0.0.0/12"
            "8.0.0.0/7"
        ],
        "DestinationPortRanges": [
            {
                "From": 0,
                "To": 65535
            }
        ],
        "Protocol": "6",
        "SourceAddresses": [
            "10.0.2.253/32"
        ],
        "SourcePortRanges": [
            {
                "From": 0,
                "To": 65535
            }
        ]
    }, [etc]
]
}
]
}
}

```

Para obtener más información, consulte [Cómo empezar a utilizar Network Access Analyzer mediante AWS CLI](#) la guía Network Access Analyzer.

- Para API obtener más información, consulte la Referencia [GetNetworkInsightsAccessScopeAnalysisFindings](#) de AWS CLI comandos.

## get-network-insights-access-scope-content

En el siguiente ejemplo de código, se muestra cómo usar `get-network-insights-access-scope-content`.

### AWS CLI

Para obtener Network Insights, acceda al contenido de Scope

En el siguiente `get-network-insights-access-scope-content` ejemplo, se obtiene el contenido del ID de análisis de alcance seleccionado en su AWS cuenta.

```
aws ec2 get-network-insights-access-scope-content \  
  --region us-east-1 \  
  --network-insights-access-scope-id nis-123456789222
```

Salida:

```
{  
  "NetworkInsightsAccessScopeContent": {  
    "NetworkInsightsAccessScopeId": "nis-123456789222",  
    "MatchPaths": [  
      {  
        "Source": {  
          "ResourceStatement": {  
            "ResourceTypes": [  
              "AWS::EC2::NetworkInterface"  
            ]  
          }  
        },  
        "Destination": {  
          "ResourceStatement": {  
            "ResourceTypes": [  
              "AWS::EC2::InternetGateway"  
            ]  
          }  
        }  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [Cómo empezar a utilizar Network Access Analyzer mediante la AWS CLI](#) Guía del Network Access Analyzer.

- Para API obtener más información, consulte la Referencia [GetNetworkInsightsAccessScopeContent](#) de AWS CLI comandos.

## get-password-data

En el siguiente ejemplo de código, se muestra cómo usar `get-password-data`.

### AWS CLI

Para obtener la contraseña cifrada

En este ejemplo se obtiene la contraseña cifrada.

Comando:

```
aws ec2 get-password-data --instance-id i-1234567890abcdef0
```

Salida:

```
{
  "InstanceId": "i-1234567890abcdef0",
  "Timestamp": "2013-08-07T22:18:38.000Z",
  "PasswordData": "gSlJFq+VpcZXqy+iktXMF6NyxQ4qCrT4+ga0uN0enX1MmgXPTj7XEXAMPLE
UQ+YeFfb+L1U4C4AKv652Ux1iRB3CPTYp7WmU3TUnhsuBd+p6LVk7T2lKUm160Xbk6WPW1VYYm/TRPB1
e1DQ7PY4an/DgZT4mwcprFigzhniQgDDe01InvSDcwoUTwNs0Y1S8ouri2W4n5GNlriM3Q0AnNVe1Vz/
53TkDtxbNoU606M1gK9zUWSxqEgwbvV2j8c5rP0WCuaMWSF14ziDu4bd7q+4RSyi8NUsVWnKZ4aEZffu
DPGzKrf5yL1f3etP2L4ZR6CvG7K1hx7VK0QVN32Dajw=="
}
```

Para obtener la contraseña descifrada

En este ejemplo se obtiene la contraseña descifrada.

Comando:

```
aws ec2 get-password-data --instance-id i-1234567890abcdef0 --priv-launch-key C:
\Keys\MyKeyPair.pem
```

Salida:



```
{
  "InstanceId": "i-1234567890abcdef0",
  "Timestamp": "2013-08-30T23:18:05.000Z",
  "PasswordData": "&ViJ652e*u"
}
```

- Para API obtener más información, consulte [GetPasswordData](#) la Referencia de AWS CLI comandos.

## get-reserved-instances-exchange-quote

En el siguiente ejemplo de código, se muestra cómo usar `get-reserved-instances-exchange-quote`.

### AWS CLI

Para obtener un presupuesto para el intercambio de una instancia reservada convertible

En este ejemplo, se obtiene la información de intercambio de las instancias reservadas convertibles especificadas.

Comando:

```
aws ec2 get-reserved-instances-exchange-quote --reserved-
instance-ids 7b8750c3-397e-4da4-bbcb-a45ebexample --target-
configurations OfferingId=6fea5434-b379-434c-b07b-a7abexample
```

Salida:

```
{
  "CurrencyCode": "USD",
  "ReservedInstanceValueSet": [
    {
      "ReservedInstanceId": "7b8750c3-397e-4da4-bbcb-a45ebexample",
      "ReservationValue": {
        "RemainingUpfrontValue": "0.000000",
        "HourlyPrice": "0.027800",
        "RemainingTotalValue": "730.556200"
      }
    }
  ],
  "PaymentDue": "424.983828",
```

```

"TargetConfigurationValueSet": [
  {
    "TargetConfiguration": {
      "InstanceCount": 5,
      "OfferingId": "6fea5434-b379-434c-b07b-a7abexample"
    },
    "ReservationValue": {
      "RemainingUpfrontValue": "424.983828",
      "HourlyPrice": "0.016000",
      "RemainingTotalValue": "845.447828"
    }
  }
],
"IsValidExchange": true,
"OutputReservedInstancesWillExpireAt": "2020-10-01T13:03:39Z",
"ReservedInstanceValueRollup": {
  "RemainingUpfrontValue": "0.000000",
  "HourlyPrice": "0.027800",
  "RemainingTotalValue": "730.556200"
},
"TargetConfigurationValueRollup": {
  "RemainingUpfrontValue": "424.983828",
  "HourlyPrice": "0.016000",
  "RemainingTotalValue": "845.447828"
}
}

```

- Para API obtener más información, consulte [GetReservedInstancesExchangeQuote](#) la Referencia de AWS CLI comandos.

## get-security-groups-for-vpc

En el siguiente ejemplo de código, se muestra cómo usar `get-security-groups-for-vpc`.

### AWS CLI

Para ver los grupos de seguridad que se pueden asociar a las interfaces de red de una forma específica VPC.

En el siguiente `get-security-groups-for-vpc` ejemplo, se muestran los grupos de seguridad que se pueden asociar a las interfaces de red del VPC.

```
aws ec2 get-security-groups-for-vpc \
```

```
--vpc-id vpc-6c31a611 \  
--region us-east-1
```

Salida:

```
{  
  "SecurityGroupForVpcs": [  
    {  
      "Description": "launch-wizard-36 created 2022-08-29T15:59:35.338Z",  
      "GroupName": "launch-wizard-36",  
      "OwnerId": "470889052923",  
      "GroupId": "sg-007e0c3027ee885f5",  
      "Tags": [],  
      "PrimaryVpcId": "vpc-6c31a611"  
    },  
    {  
      "Description": "launch-wizard-18 created 2024-01-19T20:22:27.527Z",  
      "GroupName": "launch-wizard-18",  
      "OwnerId": "470889052923",  
      "GroupId": "sg-0147193bef51c9eef",  
      "Tags": [],  
      "PrimaryVpcId": "vpc-6c31a611"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [GetSecurityGroupsForVpc](#) la Referencia de AWS CLI comandos.

## get-serial-console-access-status

En el siguiente ejemplo de código, se muestra cómo usar `get-serial-console-access-status`.

AWS CLI

Para ver el estado de la cuenta, acceda a la consola serie

El siguiente `get-serial-console-access-status` ejemplo determina si el acceso a la consola en serie está habilitado para su cuenta.

```
aws ec2 get-serial-console-access-status
```

Salida:

```
{
  "SerialConsoleAccessEnabled": true
}
```

Para obtener más información, consulta [EC2Serial Console](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [GetSerialConsoleAccessStatus](#) la Referencia de AWS CLI comandos.

## **get-snapshot-block-public-access-state**

En el siguiente ejemplo de código, se muestra cómo usar `get-snapshot-block-public-access-state`.

AWS CLI

Para obtener el estado actual del bloqueo del acceso público a las instantáneas

En el siguiente `get-snapshot-block-public-access-state` ejemplo, se obtiene el estado actual del bloqueo del acceso público a las instantáneas.

```
aws ec2 get-snapshot-block-public-access-state
```

Salida:

```
{
  "State": "block-all-sharing"
}
```

Para obtener más información, consulta [Bloquear el acceso público a las instantáneas](#) en la Guía del EBS usuario de Amazon.

- Para API obtener más información, consulte [GetSnapshotBlockPublicAccessState](#) la Referencia de AWS CLI comandos.

## **get-spot-placement-scores**

En el siguiente ejemplo de código, se muestra cómo usar `get-spot-placement-scores`.

## AWS CLI

Para calcular la puntuación de colocación en el puesto según los requisitos específicos

En primer lugar `get-spot-placement-scores`, en el ejemplo siguiente se genera una lista de todos los parámetros posibles que se pueden especificar para la configuración de la puntuación de posicionamiento del spot mediante el `--generate-cli-skeleton` parámetro y se guarda la lista en un JSON archivo. A continuación, el JSON archivo se utiliza para configurar los requisitos que se van a utilizar para calcular la puntuación de posicionamiento del spot.

Generar todos los parámetros posibles que se puedan especificar para la configuración de la puntuación de posicionamiento del spot y guardar el resultado directamente en un JSON archivo.

```
aws ec2 get-spot-placement-scores \
  --region us-east-1 \
  --generate-cli-skeleton input > attributes.json
```

Salida:

```
{
  "InstanceTypes": [
    ""
  ],
  "TargetCapacity": 0,
  "TargetCapacityUnitType": "vcpu",
  "SingleAvailabilityZone": true,
  "RegionNames": [
    ""
  ],
  "InstanceRequirementsWithMetadata": {
    "ArchitectureTypes": [
      "x86_64_mac"
    ],
    "VirtualizationTypes": [
      "hvm"
    ],
    "InstanceRequirements": {
      "VCpuCount": {
        "Min": 0,
        "Max": 0
      },
      "MemoryMiB": {
        "Min": 0,
```

```
        "Max": 0
    },
    "CpuManufacturers": [
        "amd"
    ],
    "MemoryGiBPerVCpu": {
        "Min": 0.0,
        "Max": 0.0
    },
    "ExcludedInstanceTypes": [
        ""
    ],
    "InstanceGenerations": [
        "previous"
    ],
    "SpotMaxPricePercentageOverLowestPrice": 0,
    "OnDemandMaxPricePercentageOverLowestPrice": 0,
    "BareMetal": "excluded",
    "BurstablePerformance": "excluded",
    "RequireHibernateSupport": true,
    "NetworkInterfaceCount": {
        "Min": 0,
        "Max": 0
    },
    "LocalStorage": "included",
    "LocalStorageTypes": [
        "hdd"
    ],
    "TotalLocalStorageGB": {
        "Min": 0.0,
        "Max": 0.0
    },
    "BaselineEbsBandwidthMbps": {
        "Min": 0,
        "Max": 0
    },
    "AcceleratorTypes": [
        "fpga"
    ],
    "AcceleratorCount": {
        "Min": 0,
        "Max": 0
    },
    "AcceleratorManufacturers": [
```

```

        "amd"
      ],
      "AcceleratorNames": [
        "vu9p"
      ],
      "AcceleratorTotalMemoryMiB": {
        "Min": 0,
        "Max": 0
      }
    }
  },
  "DryRun": true,
  "MaxResults": 0,
  "NextToken": ""
}

```

Configure el archivo JSON. Se debe proporcionar un valor para `TargetCapacity`. Para obtener una descripción de cada parámetro y sus valores predeterminados, consulte [Calcular la puntuación de ubicación de los puntos \(AWS CLI\) <https://docs.aws.amazon.com/AWS-EC2/latest/html/UserGuide>](https://docs.aws.amazon.com/AWS-EC2/latest/html/UserGuide). `spot-placement-score calculate-sps-cli`

Calcule la puntuación de colocación de un puesto según los requisitos especificados en `attributes.json`. Especifique el nombre y la ruta JSON del archivo mediante el `--cli-input-json` parámetro.

```

aws ec2 get-spot-placement-scores \
  --region us-east-1 \
  --cli-input-json file://attributes.json

```

La salida `SingleAvailabilityZone` se establece en `false` o se omite (si se omite, el valor predeterminado es). `false` Se devuelve una lista de regiones puntuadas.

```

"Recommendation": [
  {
    "Region": "us-east-1",
    "Score": 7
  },
  {
    "Region": "us-west-1",
    "Score": 5
  },
]

```

```
...
```

La salida `SingleAvailabilityZone` está configurada en `true`. Se devuelve una lista de `SingleAvailability` zonas puntuadas.

```
"Recommendation": [
  {
    "Region": "us-east-1",
    "AvailabilityZoneId": "use1-az1"
    "Score": 8
  },
  {
    "Region": "us-east-1",
    "AvailabilityZoneId": "usw2-az3"
    "Score": 6
  },
  ...
]
```

Para obtener más información sobre el cálculo de la puntuación de posicionamiento en un puesto y, por ejemplo, sobre las configuraciones, consulte [Calcular una puntuación de posicionamiento en un puesto](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [GetSpotPlacementScores](#) la Referencia de AWS CLI comandos.

## get-subnet-cidr-reservations

En el siguiente ejemplo de código, se muestra cómo usar `get-subnet-cidr-reservations`.

### AWS CLI

Para obtener información sobre una reserva de subred CIDR

En el siguiente `get-subnet-cidr-reservations` ejemplo, se muestra información sobre la reserva de subred CIDR especificada.

```
aws ec2 get-subnet-cidr-reservations \
  --subnet-id subnet-03c51e2e6cEXAMPLE
```

Salida:

```
{
```



```

    "SubnetIpv4CidrReservations": [
      {
        "SubnetCidrReservationId": "scr-044f977c4eEXAMPLE",
        "SubnetId": "subnet-03c51e2e6cEXAMPLE",
        "Cidr": "10.1.0.16/28",
        "ReservationType": "prefix",
        "OwnerId": "123456789012"
      }
    ],
    "SubnetIpv6CidrReservations": []
  }

```

Para obtener más información, consulta [CIDRReservas de subred](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [GetSubnetCidrReservations](#) la Referencia de AWS CLI comandos.

## get-transit-gateway-attachment-propagations

En el siguiente ejemplo de código, se muestra cómo usar `get-transit-gateway-attachment-propagations`.

### AWS CLI

Para enumerar las tablas de rutas a las que el recurso adjunto especificado propaga las rutas

En el siguiente `get-transit-gateway-attachment-propagations` ejemplo, se muestra la tabla de rutas a la que el adjunto de recursos especificado propaga las rutas.

```

aws ec2 get-transit-gateway-attachment-propagations \
  --transit-gateway-attachment-id tgw-attach-09fbd47ddfEXAMPLE

```

Salida:

```

{
  "TransitGatewayAttachmentPropagations": [
    {
      "TransitGatewayRouteTableId": "tgw-rtb-0882c61b97EXAMPLE",
      "State": "enabled"
    }
  ]
}

```

```
}
```

Para obtener más información, consulte [las tablas de rutas de Transit Gateways](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [GetTransitGatewayAttachmentPropagations](#) la Referencia de AWS CLI comandos.

## get-transit-gateway-multicast-domain-associations

En el siguiente ejemplo de código, se muestra cómo usar `get-transit-gateway-multicast-domain-associations`.

### AWS CLI

Para ver la información sobre las asociaciones de dominios de multidifusión de Transit Gateway

El siguiente `get-transit-gateway-multicast-domain-associations` ejemplo devuelve las asociaciones del dominio de multidifusión especificado.

```
aws ec2 get-transit-gateway-multicast-domain-associations \  
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE
```

Salida:

```
{  
  "MulticastDomainAssociations": [  
    {  
      "TransitGatewayAttachmentId": "tgw-attach-028c1dd0f8EXAMPLE",  
      "ResourceId": "vpc-01128d2c24EXAMPLE",  
      "ResourceType": "vpc",  
      "Subnet": {  
        "SubnetId": "subnet-000de86e3bEXAMPLE",  
        "State": "associated"  
      }  
    },  
    {  
      "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",  
      "ResourceId": "vpc-7EXAMPLE",  
      "ResourceType": "vpc",  
      "Subnet": {  
        "SubnetId": "subnet-4EXAMPLE",
```

```

        "State": "associated"
    }
},
{
    "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",
    "ResourceId": "vpc-7EXAMPLE",
    "ResourceType": "vpc",
    "Subnet": {
        "SubnetId": "subnet-5EXAMPLE",
        "State": "associated"
    }
},
{
    "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",
    "ResourceId": "vpc-7EXAMPLE",
    "ResourceType": "vpc",
    "Subnet": {
        "SubnetId": "subnet-aEXAMPLE",
        "State": "associated"
    }
},
{
    "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",
    "ResourceId": "vpc-7EXAMPLE",
    "ResourceType": "vpc",
    "Subnet": {
        "SubnetId": "subnet-fEXAMPLE",
        "State": "associated"
    }
}
]
}

```

Para obtener más información, consulte [Administración de dominios de multidifusión](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [GetTransitGatewayMulticastDomainAssociations](#) de AWS CLI comandos.

## get-transit-gateway-prefix-list-references

En el siguiente ejemplo de código, se muestra cómo usar `get-transit-gateway-prefix-list-references`.

## AWS CLI

Para obtener las referencias a la lista de prefijos en una tabla de rutas de una pasarela de tránsito

El siguiente `get-transit-gateway-prefix-list-references` ejemplo obtiene las referencias de la lista de prefijos de la tabla de rutas de la puerta de enlace de tránsito especificada y filtra por el identificador de una lista de prefijos específica.

```
aws ec2 get-transit-gateway-prefix-list-references \  
  --transit-gateway-route-table-id tgw-rtb-0123456789abcd123 \  
  --filters Name=prefix-list-id,Values=pl-111112222222333
```

Salida:

```
{  
  "TransitGatewayPrefixListReferences": [  
    {  
      "TransitGatewayRouteTableId": "tgw-rtb-0123456789abcd123",  
      "PrefixListId": "pl-111112222222333",  
      "PrefixListOwnerId": "123456789012",  
      "State": "available",  
      "Blackhole": false,  
      "TransitGatewayAttachment": {  
        "TransitGatewayAttachmentId": "tgw-attach-aabbccddaabbccaab",  
        "ResourceType": "vpc",  
        "ResourceId": "vpc-112233445566aabbcc"  
      }  
    }  
  ]  
}
```

Para obtener más información, consulte las [referencias a la lista de prefijos](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [GetTransitGatewayPrefixListReferences](#) de AWS CLI comandos.

## **get-transit-gateway-route-table-associations**

En el siguiente ejemplo de código, se muestra cómo usar `get-transit-gateway-route-table-associations`.

## AWS CLI

Para obtener información sobre las asociaciones de la tabla de rutas de la puerta de enlace de tránsito especificada

El siguiente `get-transit-gateway-route-table-associations` ejemplo muestra información sobre las asociaciones de la tabla de rutas de la puerta de enlace de tránsito especificada.

```
aws ec2 get-transit-gateway-route-table-associations \  
--transit-gateway-route-table-id tgw-rtb-0a823edbdeEXAMPLE
```

Salida:

```
{  
  "Associations": [  
    {  
      "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",  
      "ResourceId": "vpc-4d7de228",  
      "ResourceType": "vpc",  
      "State": "associating"  
    }  
  ]  
}
```

Para obtener más información, consulte [las tablas de rutas de las pasarelas de tránsito](#) en la Guía de pasarelas de tránsito.

- Para API obtener más información, consulte [GetTransitGatewayRouteTableAssociations](#) la Referencia de AWS CLI comandos.

## `get-transit-gateway-route-table-propagations`

En el siguiente ejemplo de código, se muestra cómo usar `get-transit-gateway-route-table-propagations`.

## AWS CLI

Para mostrar información sobre las propagaciones de la tabla de rutas para la tabla de rutas de la puerta de enlace de tránsito especificada

El siguiente `get-transit-gateway-route-table-propagations` ejemplo devuelve las propagaciones de la tabla de rutas de la tabla de rutas especificada.

```
aws ec2 get-transit-gateway-route-table-propagations \  
--transit-gateway-route-table-id tgw-rtb-002573ed1eEXAMPLE
```

Salida:

```
{  
  "TransitGatewayRouteTablePropagations": [  
    {  
      "TransitGatewayAttachmentId": "tgw-attach-01f8100bc7EXAMPLE",  
      "ResourceId": "vpc-3EXAMPLE",  
      "ResourceType": "vpc",  
      "State": "enabled"  
    },  
    {  
      "TransitGatewayAttachmentId": "tgw-attach-08e0bc912cEXAMPLE",  
      "ResourceId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",  
      "ResourceType": "direct-connect-gateway",  
      "State": "enabled"  
    },  
    {  
      "TransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",  
      "ResourceId": "8384da05-13ce-4a91-aada-5a1baEXAMPLE",  
      "ResourceType": "direct-connect-gateway",  
      "State": "enabled"  
    }  
  ]  
}
```

Para obtener más información, consulte [las tablas de rutas de Transit Gateways](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [GetTransitGatewayRouteTablePropagations](#) la Referencia de AWS CLI comandos.

## **get-verified-access-endpoint-policy**

En el siguiente ejemplo de código, se muestra cómo usar `get-verified-access-endpoint-policy`.

## AWS CLI

Para obtener la política de acceso verificado de un punto final

En el siguiente `get-verified-access-endpoint-policy` ejemplo, se obtiene la política de acceso verificado del punto final especificado.

```
aws ec2 get-verified-access-endpoint-policy \  
--verified-access-endpoint-id vae-066fac616d4d546f2
```

Salida:

```
{  
  "PolicyEnabled": true,  
  "PolicyDocument": "permit(principal,action,resource)\nwhen  
{\n  context.identity.groups.contains(\"finance\") &&\n  context.identity.email_verified == true\n};"}
```

Para obtener más información, consulte [las políticas de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [GetVerifiedAccessEndpointPolicy](#) la Referencia de AWS CLI comandos.

## `get-verified-access-group-policy`

En el siguiente ejemplo de código, se muestra cómo usar `get-verified-access-group-policy`.

## AWS CLI

Para obtener la política de acceso verificado de un grupo

En el siguiente `get-verified-access-group-policy` ejemplo, se obtiene la política de acceso verificado del grupo especificado.

```
aws ec2 get-verified-access-group-policy \  
--verified-access-group-id vagr-0dbe967baf14b7235
```

Salida:

```
{
  "PolicyEnabled": true,
  "PolicyDocument": "permit(principal,action,resource)\nwhen
{\n  context.identity.groups.contains(\"finance\") &&\n
context.identity.email_verified == true\n};"
}
```

Para obtener más información, consulte los [grupos de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [GetVerifiedAccessGroupPolicy](#) la Referencia de AWS CLI comandos.

## get-vpn-connection-device-sample-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-vpn-connection-device-sample-configuration`.

### AWS CLI

Para descargar un ejemplo de archivo de configuración

En el siguiente `get-vpn-connection-device-sample-configuration` ejemplo, se descarga el archivo de configuración de ejemplo especificado. Para enumerar los dispositivos de puerta de enlace con un archivo de configuración de ejemplo, `get-vpn-connection-device-types` ejecute el comando.

```
aws ec2 get-vpn-connection-device-sample-configuration \
  --vpn-connection-id vpn-123456789abc01234 \
  --vpn-connection-device-type-id 5fb390ba
```

Salida:

```
{
  "VpnConnectionDeviceSampleConfiguration": "contents-of-the-sample-configuration-
file"
}
```

Para obtener más información, consulte [Descargar el archivo de configuración](#) en la Guía del VPN usuario AWS de sitio a sitio.



- Para API obtener más información, consulte la Referencia [GetVpnConnectionDeviceSampleConfiguration](#) de AWS CLI comandos.

## get-vpn-connection-device-types

En el siguiente ejemplo de código, se muestra cómo usar `get-vpn-connection-device-types`.

### AWS CLI

Para enumerar los dispositivos de puerta de enlace con un archivo de configuración de ejemplo

En el siguiente `get-vpn-connection-device-types` ejemplo, se enumeran los dispositivos de puerta de enlace de Palo Alto Networks que tienen archivos de configuración de ejemplo.

```
aws ec2 get-vpn-connection-device-types \  
  --query "VpnConnectionDeviceTypes[?Vendor=='Palo Alto Networks']"
```

Salida:

```
[  
  {  
    "VpnConnectionDeviceTypeId": "754a6372",  
    "Vendor": "Palo Alto Networks",  
    "Platform": "PA Series",  
    "Software": "PANOS 4.1.2+"  
  },  
  {  
    "VpnConnectionDeviceTypeId": "9612cbed",  
    "Vendor": "Palo Alto Networks",  
    "Platform": "PA Series",  
    "Software": "PANOS 4.1.2+ (GUI)"  
  },  
  {  
    "VpnConnectionDeviceTypeId": "5fb390ba",  
    "Vendor": "Palo Alto Networks",  
    "Platform": "PA Series",  
    "Software": "PANOS 7.0+"  
  }  
]
```

Para obtener más información, consulte [Descargar el archivo de configuración](#) en la Guía del VPNusuario AWS de sitio a sitio.

- Para API obtener más información, consulte la Referencia [GetVpnConnectionDeviceTypes](#) de AWS CLI comandos.

## **import-client-vpn-client-certificate-revocation-list**

En el siguiente ejemplo de código, se muestra cómo usar `import-client-vpn-client-certificate-revocation-list`.

### AWS CLI

Para importar una lista de revocaciones de certificados de cliente

En el siguiente `import-client-vpn-client-certificate-revocation-list` ejemplo, se importa una lista de revocaciones de certificados de cliente al VPN punto final del cliente especificando la ubicación del archivo en el equipo local.

```
aws ec2 import-client-vpn-client-certificate-revocation-list \  
  --certificate-revocation-list file:///path/to/crl.pem \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Salida:

```
{  
  "Return": true  
}
```

Para obtener más información, consulte las [listas de revocación de certificados de cliente](#) en la Guía del VPNadministrador de AWS clientes.

- Para API obtener más información, consulte [ImportClientVpnClientCertificateRevocationList](#) la Referencia de AWS CLI comandos.

## **import-image**

En el siguiente ejemplo de código, se muestra cómo usar `import-image`.

### AWS CLI

Para importar un archivo de imagen de máquina virtual como AMI

En el siguiente `import-image` ejemplo, se importa lo especificado OVA.

```
aws ec2 import-image \  
  --disk-containers Format=ova,UserBucket="{S3Bucket=my-import-bucket,S3Key=vms/my-  
server-vm.ova}"
```

Salida:

```
{  
  "ImportTaskId": "import-ami-1234567890abcdef0",  
  "Progress": "2",  
  "SnapshotDetails": [  
    {  
      "DiskImageSize": 0.0,  
      "Format": "ova",  
      "UserBucket": {  
        "S3Bucket": "my-import-bucket",  
        "S3Key": "vms/my-server-vm.ova"  
      }  
    }  
  ],  
  "Status": "active",  
  "StatusMessage": "pending"  
}
```

- Para API obtener más información, consulte [ImportImage](#) la Referencia de AWS CLI comandos.

## import-key-pair

En el siguiente ejemplo de código, se muestra cómo usar `import-key-pair`.

### AWS CLI

Para importar una clave pública

En primer lugar, genere un key pair con la herramienta que prefiera. Por ejemplo, utilice este comando `ssh-keygen`:

Comando:

```
ssh-keygen -t rsa -C "my-key" -f ~/.ssh/my-key
```

Salida:

```
Generating public/private rsa key pair.  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in /home/ec2-user/.ssh/my-key.  
Your public key has been saved in /home/ec2-user/.ssh/my-key.pub.  
...
```

Este comando de ejemplo importa la clave pública especificada.

Comando:

```
aws ec2 import-key-pair --key-name "my-key" --public-key-material fileb://~/.ssh/my-key.pub
```

Salida:

```
{  
  "KeyName": "my-key",  
  "KeyFingerprint": "1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca"  
}
```

- Para API obtener más información, consulte [ImportKeyPair](#) la Referencia de AWS CLI comandos.

## import-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `import-snapshot`.

AWS CLI

Para importar una instantánea

En el siguiente `import-snapshot` ejemplo, se importa el disco especificado como una instantánea.

```
aws ec2 import-snapshot \  
  --description "My server VMDK" \  
  --disk-container Format=VMDK,UserBucket={S3Bucket=my-import-bucket,S3Key=vms/my-server-vm.vmdk}
```

Salida:

```
{
  "Description": "My server VMDK",
  "ImportTaskId": "import-snap-1234567890abcdef0",
  "SnapshotTaskDetail": {
    "Description": "My server VMDK",
    "DiskImageSize": "0.0",
    "Format": "VMDK",
    "Progress": "3",
    "Status": "active",
    "StatusMessage": "pending"
    "UserBucket": {
      "S3Bucket": "my-import-bucket",
      "S3Key": "vms/my-server-vm.vmdk"
    }
  }
}
```

- Para API obtener más información, consulte [ImportSnapshot](#) la Referencia de AWS CLI comandos.

## list-images-in-recycle-bin

En el siguiente ejemplo de código, se muestra cómo usar `list-images-in-recycle-bin`.

### AWS CLI

Para enumerar las imágenes de la papelera de reciclaje

En el siguiente `list-images-in-recycle-bin` ejemplo, se enumeran todas las imágenes que se conservan actualmente en la papelera de reciclaje.

```
aws ec2 list-images-in-recycle-bin
```

Salida:

```
{
  "Images": [
    {
      "RecycleBinEnterTime": "2022-03-14T15:35:08.000Z",
      "Description": "Monthly AMI One",
      "RecycleBinExitTime": "2022-03-15T15:35:08.000Z",
```

```
        "Name": "AMI_01",
        "ImageId": "ami-0111222333444abcd"
    }
]
}
```

Para obtener más información, consulte [Recuperarse AMIs de la papelera de reciclaje](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte [ListImagesInRecycleBin](#) la Referencia de AWS CLI comandos.

## list-snapshots-in-recycle-bin

En el siguiente ejemplo de código, se muestra cómo usar `list-snapshots-in-recycle-bin`.

### AWS CLI

Para ver las instantáneas en la papelera de reciclaje

En el siguiente `list-snapshots-in-recycle-bin` ejemplo, se muestra información sobre las instantáneas de la papelera de reciclaje, incluido el ID de la instantánea, una descripción de la instantánea, el ID del volumen a partir del cual se creó la instantánea, la fecha y la hora en que se eliminó la instantánea y entró en la papelera de reciclaje, y la fecha y la hora en que vence el período de retención.

```
aws ec2 list-snapshots-in-recycle-bin \
  --snapshot-id snap-01234567890abcdef
```

Salida:

```
{
  "SnapshotRecycleBinInfo": [
    {
      "Description": "Monthly data backup snapshot",
      "RecycleBinEnterTime": "2022-12-01T13:00:00.000Z",
      "RecycleBinExitTime": "2022-12-15T13:00:00.000Z",
      "VolumeId": "vol-abcdef09876543210",
      "SnapshotId": "snap-01234567890abcdef"
    }
  ]
}
```

```
}
```

Para obtener más información sobre la papelera de reciclaje de AmazonEBS, consulte [Recuperar instantáneas de la papelera de reciclaje](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [ListSnapshotsInRecycleBin](#) la Referencia de AWS CLI comandos.

## lock-snapshot

En el siguiente ejemplo de código, se muestra cómo usar lock-snapshot.

### AWS CLI

Ejemplo 1: Para bloquear una instantánea en modo de gobierno

El siguiente lock-snapshot ejemplo bloquea la instantánea especificada en el modo de gobierno.

```
aws ec2 lock-snapshot \  
  --snapshot-id snap-0b5e733b4a8df6e0d \  
  --lock-mode governance \  
  --lock-duration 365
```

Salida:

```
{  
  "SnapshotId": "snap-0b5e733b4a8df6e0d",  
  "LockState": "governance",  
  "LockDuration": 365,  
  "LockCreatedOn": "2024-05-05T00:56:06.208000+00:00",  
  "LockExpiresOn": "2025-05-05T00:56:06.208000+00:00",  
  "LockDurationStartTime": "2024-05-05T00:56:06.208000+00:00"  
}
```

Para obtener más información, consulta [Snapshot Lock](#) en la Guía del EBS usuario de Amazon.

Ejemplo 2: Para bloquear una instantánea en modo de conformidad

En el siguiente lock-snapshot ejemplo, se bloquea la instantánea especificada en el modo de conformidad.

```
aws ec2 lock-snapshot \  
  --snapshot-id snap-0163a8524c5b9901f \  
  --lock-mode compliance \  
  --cool-off-period 24 \  
  --lock-duration 365
```

Salida:

```
{  
  "SnapshotId": "snap-0b5e733b4a8df6e0d",  
  "LockState": "compliance-cooloff",  
  "LockDuration": 365,  
  "CooloffPeriod": 24,  
  "CooloffPeriodExpiresOn": "2024-05-06T01:02:20.527000+00:00",  
  "LockCreatedOn": "2024-05-05T01:02:20.527000+00:00",  
  "LockExpiresOn": "2025-05-05T01:02:20.527000+00:00",  
  "LockDurationStartTime": "2024-05-05T01:02:20.527000+00:00"  
}
```

Para obtener más información, consulta [Snapshot Lock](#) en la Guía del EBS usuario de Amazon.

- Para API obtener más información, consulte [LockSnapshot](#) la Referencia de AWS CLI comandos.

## modify-address-attribute

En el siguiente ejemplo de código, se muestra cómo usar `modify-address-attribute`.

AWS CLI

Para modificar el atributo de nombre de dominio asociado a una dirección IP elástica

Los siguientes `modify-address-attribute` ejemplos modifican el atributo de nombre de dominio de una dirección IP elástica.

Linux:

```
aws ec2 modify-address-attribute \  
  --allocation-id eipalloc-abcdef01234567890 \  
  --domain-name example.com
```



Windows:

```
aws ec2 modify-address-attribute ^
  --allocation-id eipalloc-abcdef01234567890 ^
  --domain-name example.com
```

Salida:

```
{
  "Addresses": [
    {
      "PublicIp": "192.0.2.0",
      "AllocationId": "eipalloc-abcdef01234567890",
      "PtrRecord": "example.net."
      "PtrRecordUpdate": {
        "Value": "example.com.",
        "Status": "PENDING"
      }
    }
  ]
}
```

Para supervisar el cambio pendiente y ver los atributos modificados de una dirección IP elástica, consulte [describe-addresses-attribute](#) la Referencia de AWS CLI comandos.

- Para API obtener más información, consulte [ModifyAddressAttribute](#) la Referencia de AWS CLI comandos.

## modify-availability-zone-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-availability-zone-group`.

AWS CLI

Para habilitar un grupo de zonas

El siguiente `modify-availability-zone-group` ejemplo habilita el grupo de zonas especificado.

```
aws ec2 modify-availability-zone-group \
  --group-name us-west-2-lax-1 \
  --opt-in-status opted-in
```

Salida:

```
{
  "Return": true
}
```

Para obtener más información, consulte [Regiones y zonas](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [ModifyAvailabilityZoneGroup](#) la Referencia de AWS CLI comandos.

## **modify-capacity-reservation-fleet**

En el siguiente ejemplo de código, se muestra cómo usar `modify-capacity-reservation-fleet`.

AWS CLI

Ejemplo 1: Para modificar la capacidad objetivo total de una flota de reserva de capacidad

El siguiente `modify-capacity-reservation-fleet` ejemplo modifica la capacidad objetivo total de la flota de reserva de capacidad especificada. Al modificar la capacidad de destino total de una Flota de Reservas de capacidad, la flota crea automáticamente nuevas Reservas de capacidad o modifica o cancela las Reservas de capacidad existentes en la flota para cumplir la nueva capacidad de destino total. No puede intentar realizar modificaciones adicionales en una flota mientras esté en el estado `modifying`.

```
aws ec2 modify-capacity-reservation-fleet \
  --capacity-reservation-fleet-id crf-01234567890abcdef \
  --total-target-capacity 160
```

Salida:

```
{
  "Return": true
}
```

Ejemplo 2: Para modificar la fecha de finalización de una flota de reserva de capacidad

El siguiente `modify-capacity-reservation-fleet` ejemplo modifica la fecha de finalización de la flota de reserva de capacidad especificada. Al modificar la fecha de finalización de la flota, las fechas de finalización de todas las Reservas de capacidad individuales se actualizan en consecuencia. No puede intentar realizar modificaciones adicionales en una flota mientras esté en el estado `modifying`.

```
aws ec2 modify-capacity-reservation-fleet \  
  --capacity-reservation-fleet-id crf-01234567890abcdef \  
  --end-date 2022-07-04T23:59:59.000Z
```

Salida:

```
{  
  "Return": true  
}
```

Para obtener más información sobre las flotas de reserva de capacidad, consulta [Flotas de reserva de capacidad](#) en la Guía EC2 del usuario de Amazon.

- Para API obtener más información, consulte la Referencia [ModifyCapacityReservationFleet](#) de AWS CLI comandos.

## **modify-capacity-reservation**

En el siguiente ejemplo de código, se muestra cómo usar `modify-capacity-reservation`.

AWS CLI

Ejemplo 1: Para cambiar el número de instancias reservadas por una reserva de capacidad existente

El siguiente `modify-capacity-reservation` ejemplo cambia el número de instancias para las que la reserva de capacidad reserva capacidad.

```
aws ec2 modify-capacity-reservation \  
  --capacity-reservation-id cr-1234abcd56EXAMPLE \  
  --instance-count 5
```

Salida:

```
{
  "Return": true
}
```

Ejemplo 2: Para cambiar la fecha y la hora de finalización de una reserva de capacidad existente

El siguiente `modify-capacity-reservation` ejemplo modifica una reserva de capacidad existente para que finalice en la fecha y hora especificadas.

```
aws ec2 modify-capacity-reservation \
  --capacity-reservation-id cr-1234abcd56EXAMPLE \
  --end-date-type Limited \
  --end-date 2019-08-31T23:59:59Z
```

Para obtener más información, consulte [Modificación de una reserva de capacidad](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [ModifyCapacityReservation](#) la Referencia de AWS CLI comandos.

## `modify-client-vpn-endpoint`

En el siguiente ejemplo de código, se muestra cómo usar `modify-client-vpn-endpoint`.

### AWS CLI

Para modificar un VPN punto final del cliente

El siguiente `modify-client-vpn-endpoint` ejemplo habilita el registro de conexiones del cliente para el VPN punto final del cliente especificado.

```
aws ec2 modify-client-vpn-endpoint \
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
  --connection-log-options Enabled=true,CloudwatchLogGroup=ClientVPNLogs
```

Salida:

```
{
  "Return": true
}
```

Para obtener más información, consulte los [VPNpuntos finales del cliente](#) en la Guía del VPN administrador del AWS cliente.

- Para API obtener más información, consulte [ModifyClientVpnEndpoint](#) la Referencia de AWS CLI comandos.

## **modify-default-credit-specification**

En el siguiente ejemplo de código, se muestra cómo usar `modify-default-credit-specification`.

### AWS CLI

Para modificar la opción de crédito por defecto

En el siguiente `modify-default-credit-specification` ejemplo, se modifica la opción de crédito por defecto para las instancias T2.

```
aws ec2 modify-default-credit-specification \  
  --instance-family t2 \  
  --cpu-credits unlimited
```

Salida:

```
{  
  "InstanceFamilyCreditSpecification": {  
    "InstanceFamily": "t2",  
    "CpuCredits": "unlimited"  
  }  
}
```

- Para API obtener más información, consulte [ModifyDefaultCreditSpecification](#) la Referencia de AWS CLI comandos.

## **modify-ebs-default-kms-key-id**

En el siguiente ejemplo de código, se muestra cómo usar `modify-ebs-default-kms-key-id`.

### AWS CLI

Para establecer el EBS cifrado predeterminado CMK

En el siguiente `modify-ebs-default-kms-key-id` ejemplo, se establece el EBS cifrado especificado CMK como predeterminado CMK para su AWS cuenta en la región actual.

```
aws ec2 modify-ebs-default-kms-key-id \  
--kms-key-id alias/my-cmk
```

Salida:

```
{  
  "KmsKeyId": "arn:aws:kms:us-  
west-2:123456789012:key/0ea3fef3-80a7-4778-9d8c-1c0c6EXAMPLE"  
}
```

- Para API obtener más información, consulte [ModifyEbsDefaultKmsKeyId](#) la Referencia de AWS CLI comandos.

## modify-fleet

En el siguiente ejemplo de código, se muestra cómo usar `modify-fleet`.

### AWS CLI

Para escalar una EC2 flota

El siguiente `modify-fleet` ejemplo modifica la capacidad objetivo de la EC2 flota especificada. Si el valor especificado es superior a la capacidad actual, la EC2 flota lanza instancias adicionales. Si el valor especificado es inferior a la capacidad actual, la EC2 flota cancela todas las solicitudes pendientes y, si la política de cancelación lo `terminate`, la EC2 flota cancela cualquier instancia que supere la nueva capacidad objetivo.

```
aws ec2 modify-fleet \  
--fleet-ids fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE \  
--target-capacity-specification TotalTargetCapacity=5
```

Salida:

```
{  
  "Return": true  
}
```

Para obtener más información, consulte [Administración de una EC2 flota](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [ModifyFleet](#) la Referencia de AWS CLI comandos.

## **modify-fpga-image-attribute**

En el siguiente ejemplo de código, se muestra cómo usar `modify-fpga-image-attribute`.

### AWS CLI

Para modificar los atributos de una FPGA imagen de Amazon

En este ejemplo, se añaden permisos de carga 123456789012 para el identificador de cuenta especificadoAFI.

Comando:

```
aws ec2 modify-fpga-image-attribute --attribute LoadPermission --fpga-image-id afi-0d123e123bfc85abc --load-permission Add=[{UserId=123456789012}]
```

Salida:

```
{
  "FpgaImageAttribute": {
    "FpgaImageId": "afi-0d123e123bfc85abc",
    "LoadPermissions": [
      {
        "UserId": "123456789012"
      }
    ]
  }
}
```

- Para API obtener más información, consulte [ModifyFpgaImageAttribute](#) la Referencia de AWS CLI comandos.

## **modify-hosts**

En el siguiente ejemplo de código, se muestra cómo usar `modify-hosts`.

## AWS CLI

Ejemplo 1: Para habilitar la ubicación automática de un host dedicado

El siguiente `modify-hosts` ejemplo habilita la ubicación automática de un host dedicado para que acepte cualquier lanzamiento de instancia no segmentado que coincida con su configuración de tipo de instancia.

```
aws ec2 modify-hosts \  
  --host-id h-06c2f189b4EXAMPLE \  
  --auto-placement on
```

Salida:

```
{  
  "Successful": [  
    "h-06c2f189b4EXAMPLE"  
  ],  
  "Unsuccessful": []  
}
```

Ejemplo 2: Para habilitar la recuperación de un host dedicado

El siguiente `modify-hosts` ejemplo permite la recuperación del host dedicado especificado.

```
aws ec2 modify-hosts \  
  --host-id h-06c2f189b4EXAMPLE \  
  --host-recovery on
```

Salida:

```
{  
  "Successful": [  
    "h-06c2f189b4EXAMPLE"  
  ],  
  "Unsuccessful": []  
}
```

Para obtener más información, consulte [Modificación de la ubicación automática de un host dedicado](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.



- Para API obtener más información, consulte [ModifyHosts](#) la Referencia de AWS CLI comandos.

## modify-id-format

En el siguiente ejemplo de código, se muestra cómo usar `modify-id-format`.

### AWS CLI

Para habilitar el formato de ID más largo para un recurso

El siguiente `modify-id-format` ejemplo habilita el formato de ID más largo para el tipo de `instance` recurso.

```
aws ec2 modify-id-format \  
  --resource instance \  
  --use-long-ids
```

Para deshabilitar el formato de ID más largo de un recurso

El siguiente `modify-id-format` ejemplo deshabilita el formato de ID más largo para el tipo `instance` de recurso.

```
aws ec2 modify-id-format \  
  --resource instance \  
  --no-use-long-ids
```

El siguiente `modify-id-format` ejemplo habilita el formato de ID más largo para todos los tipos de recursos compatibles que estén dentro de su período de suscripción.

```
aws ec2 modify-id-format \  
  --resource all-current \  
  --use-long-ids
```

- Para API obtener más información, consulte [ModifyIdFormat](#) la Referencia de AWS CLI comandos.

## modify-identity-id-format

En el siguiente ejemplo de código, se muestra cómo usar `modify-identity-id-format`.

## AWS CLI

Para permitir que un IAM rol se utilice IDs durante más tiempo como recurso

El siguiente `modify-identity-id-format` ejemplo permite que el IAM rol `EC2Role` de su AWS cuenta utilice un formato de ID largo para el tipo de `instance` recurso.

```
aws ec2 modify-identity-id-format \  
  --principal-arn arn:aws:iam::123456789012:role/EC2Role \  
  --resource instance \  
  --use-long-ids
```

Para permitir que un IAM usuario utilice un recurso IDs durante más tiempo

El siguiente `modify-identity-id-format` ejemplo permite al IAM usuario `AdminUser` de su AWS cuenta utilizar el formato de ID más largo para el tipo de `volume` recurso.

```
aws ec2 modify-identity-id-format \  
  --principal-arn arn:aws:iam::123456789012:user/AdminUser \  
  --resource volume \  
  --use-long-ids
```

El siguiente `modify-identity-id-format` ejemplo permite al IAM usuario `AdminUser` de su AWS cuenta utilizar el formato de ID más largo para todos los tipos de recursos admitidos que estén dentro de su período de suscripción.

```
aws ec2 modify-identity-id-format \  
  --principal-arn arn:aws:iam::123456789012:user/AdminUser \  
  --resource all-current \  
  --use-long-ids
```

- Para API obtener más información, consulte [ModifyIdentityIdFormat](#) la Referencia de AWS CLI comandos.

## `modify-image-attribute`

En el siguiente ejemplo de código, se muestra cómo usar `modify-image-attribute`.

## AWS CLI

Ejemplo 1: Para hacer un AMI público

En el siguiente `modify-instance-attribute` ejemplo, se convierte en AMI público lo especificado.

```
aws ec2 modify-image-attribute \  
  --image-id ami-5731123e \  
  --launch-permission "Add=[{Group=all}]"
```

Este comando no genera ninguna salida.

Ejemplo 2: Para hacer una AMI privada

En el siguiente `modify-instance-attribute` ejemplo, se convierte en AMI privado lo especificado.

```
aws ec2 modify-image-attribute \  
  --image-id ami-5731123e \  
  --launch-permission "Remove=[{Group=all}]"
```

Este comando no genera ninguna salida.

Ejemplo 3: conceder el permiso de lanzamiento a una AWS cuenta

En el siguiente `modify-instance-attribute` ejemplo, se conceden permisos de lanzamiento a la AWS cuenta especificada.

```
aws ec2 modify-image-attribute \  
  --image-id ami-5731123e \  
  --launch-permission "Add=[{UserId=123456789012}]"
```

Este comando no genera ninguna salida.

Ejemplo 4: Para eliminar el permiso de lanzamiento de una AWS cuenta

En el siguiente `modify-instance-attribute` ejemplo, se eliminan los permisos de lanzamiento de la AWS cuenta especificada.

```
aws ec2 modify-image-attribute \  
  --image-id ami-5731123e \  
  --launch-permission "Remove=[{UserId=123456789012}]"
```

- Para API obtener más información, consulte [ModifyImageAttribute](#) la Referencia de AWS CLI comandos.

## modify-instance-attribute

En el siguiente ejemplo de código, se muestra cómo usar `modify-instance-attribute`.

### AWS CLI

Ejemplo 1: Para modificar el tipo de instancia

El siguiente `modify-instance-attribute` ejemplo modifica el tipo de instancia de la instancia especificada. La instancia debe tener el estado `stopped`.

```
aws ec2 modify-instance-attribute \  
  --instance-id i-1234567890abcdef0 \  
  --instance-type "{\"Value\": \"m1.small\"}"
```

Este comando no genera ninguna salida.

Ejemplo 2: Para habilitar una red mejorada en una instancia

El siguiente `modify-instance-attribute` ejemplo habilita una red mejorada para la instancia especificada. La instancia debe tener el estado `stopped`.

```
aws ec2 modify-instance-attribute \  
  --instance-id i-1234567890abcdef0 \  
  --sriov-net-support simple
```

Este comando no genera ninguna salida.

Ejemplo 3: Para modificar el `sourceDestCheck` atributo

En el siguiente `modify-instance-attribute` ejemplo, se establece el `sourceDestCheck` atributo de la instancia especificada en `true`. La instancia debe estar en un VPC.

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --source-dest-  
check "{\"Value\": true}"
```

Este comando no genera ninguna salida.

#### Ejemplo 4: Para modificar el deleteOnTermination atributo del volumen raíz

En el siguiente `modify-instance-attribute` ejemplo, se establece el `deleteOnTermination` atributo del volumen raíz de la instancia EBS respaldada por Amazon especificada en. `false` De forma predeterminada, este atributo es `true` para el volumen raíz.

Comando:

```
aws ec2 modify-instance-attribute \  
  --instance-id i-1234567890abcdef0 \  
  --block-device-mappings "[{"DeviceName\": \"/dev/sda1\","Ebs\":  
{"DeleteOnTermination\":false}]]"
```

Este comando no genera ninguna salida.

#### Ejemplo 5: Para modificar los datos de usuario adjuntos a una instancia

En el siguiente `modify-instance-attribute` ejemplo, se agrega el contenido del archivo `UserData.txt` como el `UserData` de la instancia especificada.

Contenido del archivo `originalUserData.txt`:

```
#!/bin/bash  
yum update -y  
service httpd start  
chkconfig httpd on
```

El contenido del archivo debe estar codificado en base64. El primer comando convierte el archivo de texto a base64 y lo guarda como un archivo nuevo.

Versión del comando para Linux/macOS:

```
base64 UserData.txt > UserData.base64.txt
```

Este comando no genera ninguna salida.

Versión del comando para Windows:

```
certutil -encode UserData.txt tmp.b64 && findstr /v /c:- tmp.b64 >  
UserData.base64.txt
```

Salida:

```
Input Length = 67
Output Length = 152
CertUtil: -encode command completed successfully.
```

Ahora puede hacer referencia a ese archivo en el siguiente CLI comando:

```
aws ec2 modify-instance-attribute \  
  --instance-id=i-09b5a14dbca622e76 \  
  --attribute userData --value file://UserData.base64.txt
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Datos de usuario y AWS CLI en la Guía del EC2 usuario](#).

- Para API obtener más información, consulte [ModifyInstanceAttribute](#) la Referencia de AWS CLI comandos.

## **modify-instance-capacity-reservation-attributes**

En el siguiente ejemplo de código, se muestra cómo usar `modify-instance-capacity-reservation-attributes`.

AWS CLI

Ejemplo 1: Para modificar la configuración de segmentación por reserva de capacidad de una instancia

En el siguiente `modify-instance-capacity-reservation-attributes` ejemplo, se modifica una instancia detenida para orientarla a una reserva de capacidad específica.

```
aws ec2 modify-instance-capacity-reservation-attributes \  
  --instance-id i-EXAMPLE8765abcd4e \  
  --capacity-reservation-specification  
  'CapacityReservationTarget={CapacityReservationId= cr-1234abcd56EXAMPLE }'
```

Salida:

```
{  
  "Return": true
```

```
}
```

Ejemplo 2: Para modificar la configuración de segmentación por reserva de capacidad de una instancia

En el siguiente `modify-instance-capacity-reservation-attributes` ejemplo, se modifica una instancia detenida que tiene como objetivo la reserva de capacidad especificada para lanzarla en cualquier reserva de capacidad que tenga atributos coincidentes (tipo de instancia, plataforma, zona de disponibilidad) y que tenga criterios de coincidencia de instancias abiertas.

```
aws ec2 modify-instance-capacity-reservation-attributes \  
  --instance-id i-EXAMPLE8765abcd4e \  
  --capacity-reservation-specification 'CapacityReservationPreference=open'
```

Salida:

```
{  
  "Return": true  
}
```

Para obtener más información, consulte [Modificación de la configuración de reserva de capacidad de una instancia](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [ModifyInstanceCapacityReservationAttributes](#) la Referencia de AWS CLI comandos.

## **modify-instance-credit-specification**

En el siguiente ejemplo de código, se muestra cómo usar `modify-instance-credit-specification`.

### AWS CLI

Para modificar la opción de crédito para el CPU uso de una instancia

En este ejemplo, se modifica la opción de crédito para el CPU uso de la instancia especificada en la región especificada a «ilimitada». Las opciones de crédito válidas son «estándar» e «ilimitado».

Comando:

```
aws ec2 modify-instance-credit-specification --instance-credit-specification "InstanceId=i-1234567890abcdef,CpuCredits=unlimited"
```

Salida:

```
{
  "SuccessfulInstanceCreditSpecifications": [
    {
      "InstanceId": "i-1234567890abcdef0"
    }
  ],
  "UnsuccessfulInstanceCreditSpecifications": []
}
```

- Para API obtener más información, consulte [ModifyInstanceCreditSpecification](#) la Referencia de AWS CLI comandos.

## modify-instance-event-start-time

En el siguiente ejemplo de código, se muestra cómo usar `modify-instance-event-start-time`.

### AWS CLI

Para modificar la hora de inicio del evento de una instancia

El siguiente `modify-instance-event-start-time` comando muestra cómo modificar la hora de inicio del evento para la instancia especificada. Especifique el ID del evento mediante el `--instance-event-id` parámetro. Especifique la nueva fecha y hora mediante el `--not-before` parámetro.

```
aws ec2 modify-instance-event-start-time --instance-id i-1234567890abcdef0 --instance-event-id instance-event-0abcdef1234567890 --not-before 2019-03-25T10:00:00.000
```

Salida:

```
"Event": {
  "InstanceEventId": "instance-event-0abcdef1234567890",
  "Code": "system-reboot",
```



```

    "Description": "scheduled reboot",
    "NotAfter": "2019-03-25T12:00:00.000Z",
    "NotBefore": "2019-03-25T10:00:00.000Z",
    "NotBeforeDeadline": "2019-04-22T21:00:00.000Z"
  }

```

Para obtener más información, consulte Trabajar con instancias programadas para el reinicio en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte [ModifyInstanceEventStartTime](#) la Referencia de AWS CLI comandos.

## modify-instance-event-window

En el siguiente ejemplo de código, se muestra cómo usar `modify-instance-event-window`.

### AWS CLI

Ejemplo 1: Para modificar el intervalo de tiempo de una ventana de eventos

El siguiente `modify-instance-event-window` ejemplo modifica el intervalo de tiempo de una ventana de eventos. Especifique el parámetro `time-range` para modificar el intervalo de tiempo. No puede especificar el parámetro `cron-expression`.

```

aws ec2 modify-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890
  --time-range StartWeekDay=monday, StartHour=2, EndWeekDay=wednesday, EndHour=8

```

Salida:

```

{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "TimeRanges": [
      {
        "StartWeekDay": "monday",
        "StartHour": 2,
        "EndWeekDay": "wednesday",
        "EndHour": 8
      }
    ],
  },
}

```

```

    "Name": "myEventWindowName",
    "AssociationTarget": {
      "InstanceIds": [
        "i-0abcdef1234567890",
        "i-0be35f9acb8ba01f0"
      ],
      "Tags": [],
      "DedicatedHostIds": []
    },
    "State": "creating",
    "Tags": [
      {
        "Key": "K1",
        "Value": "V1"
      }
    ]
  }
}

```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

Ejemplo 2: Modificar un conjunto de intervalos de tiempo para una ventana de eventos

El siguiente `modify-instance-event-window` ejemplo modifica el intervalo de tiempo de una ventana de eventos. Especifique el parámetro `time-range` para modificar el intervalo de tiempo. No puede especificar el parámetro `cron-expression`.

```

aws ec2 modify-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --time-range '["StartWeekDay": "monday", "StartHour": 2, "EndWeekDay":
wednesday, "EndHour": 8],
  ["StartWeekDay": "thursday", "StartHour": 2, "EndWeekDay": "friday",
EndHour": 8}]'

```

Salida:

```

{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "TimeRanges": [

```

```

    {
      "StartWeekDay": "monday",
      "StartHour": 2,
      "EndWeekDay": "wednesday",
      "EndHour": 8
    },
    {
      "StartWeekDay": "thursday",
      "StartHour": 2,
      "EndWeekDay": "friday",
      "EndHour": 8
    }
  ],
  "Name": "myEventWindowName",
  "AssociationTarget": {
    "InstanceIds": [
      "i-0abcdef1234567890",
      "i-0be35f9acb8ba01f0"
    ],
    "Tags": [],
    "DedicatedHostIds": []
  },
  "State": "creating",
  "Tags": [
    {
      "Key": "K1",
      "Value": "V1"
    }
  ]
}
}

```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

Ejemplo 3: Para modificar la expresión cron de una ventana de eventos

El siguiente `modify-instance-event-window` ejemplo modifica la expresión cron de una ventana de eventos. Especifique el parámetro `cron-expression` para modificar la expresión cron. No puede especificar el parámetro `time-range`.

```

aws ec2 modify-instance-event-window \
  --region us-east-1 \

```

```
--instance-event-window-id iew-0abcdef1234567890 \  
--cron-expression "* 21-23 * * 2,3"
```

Salida:

```
{  
  "InstanceEventWindow": {  
    "InstanceEventWindowId": "iew-0abcdef1234567890",  
    "Name": "myEventWindowName",  
    "CronExpression": "* 21-23 * * 2,3",  
    "AssociationTarget": {  
      "InstanceIds": [  
        "i-0abcdef1234567890",  
        "i-0be35f9acb8ba01f0"  
      ],  
      "Tags": [],  
      "DedicatedHostIds": []  
    },  
    "State": "creating",  
    "Tags": [  
      {  
        "Key": "K1",  
        "Value": "V1"  
      }  
    ]  
  }  
}
```

Para conocer las restricciones de la ventana de eventos, consulte [Consideraciones](#) en la sección Eventos programados de la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [ModifyInstanceEventWindow](#) la Referencia de AWS CLI comandos.

## modify-instance-maintenance-options

En el siguiente ejemplo de código, se muestra cómo usar `modify-instance-maintenance-options`.

AWS CLI

Ejemplo 1: Para deshabilitar el comportamiento de recuperación de una instancia

En el siguiente `modify-instance-maintenance-options` ejemplo, se deshabilita la recuperación automática simplificada para una instancia en ejecución o detenida.

```
aws ec2 modify-instance-maintenance-options \  
  --instance-id i-0abcdef1234567890 \  
  --auto-recovery disabled
```

Salida:

```
{  
  "InstanceId": "i-0abcdef1234567890",  
  "AutoRecovery": "disabled"  
}
```

Para obtener más información, consulta [Recupera tu instancia](#) en la Guía del EC2 usuario de Amazon para instancias de Linux.

Ejemplo 2: Para configurar el comportamiento de recuperación de una instancia como predeterminado

El siguiente `modify-instance-maintenance-options` ejemplo establece el comportamiento de recuperación automática como predeterminado, lo que permite una recuperación automática simplificada para los tipos de instancias compatibles.

```
aws ec2 modify-instance-maintenance-options \  
  --instance-id i-0abcdef1234567890 \  
  --auto-recovery default
```

Salida:

```
{  
  "InstanceId": "i-0abcdef1234567890",  
  "AutoRecovery": "default"  
}
```

Para obtener más información, consulta [Recupera tu instancia](#) en la Guía del EC2 usuario de Amazon para instancias de Linux.

- Para API obtener más información, consulte [ModifyInstanceMaintenanceOptions](#) la Referencia de AWS CLI comandos.

## modify-instance-metadata-options

En el siguiente ejemplo de código, se muestra cómo usar `modify-instance-metadata-options`.

### AWS CLI

Ejemplo 1: Para habilitar IMDSv2

El siguiente `modify-instance-metadata-options` ejemplo configura el uso de IMDSv2 en la instancia especificada.

```
aws ec2 modify-instance-metadata-options \  
  --instance-id i-1234567898abcdef0 \  
  --http-tokens required \  
  --http-endpoint enabled
```

Salida:

```
{  
  "InstanceId": "i-1234567898abcdef0",  
  "InstanceMetadataOptions": {  
    "State": "pending",  
    "HttpTokens": "required",  
    "HttpPutResponseHopLimit": 1,  
    "HttpEndpoint": "enabled"  
  }  
}
```

Para obtener más información, consulte [los metadatos de las instancias y los datos de usuario](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

Ejemplo 2: Para deshabilitar los metadatos de la instancia

El siguiente `modify-instance-metadata-options` ejemplo inhabilita el uso de todas las versiones de los metadatos de la instancia en la instancia especificada.

```
aws ec2 modify-instance-metadata-options \  
  --instance-id i-1234567898abcdef0 \  
  --http-endpoint disabled
```

Salida:

```
{
  "InstanceId": "i-1234567898abcdef0",
  "InstanceMetadataOptions": {
    "State": "pending",
    "HttpTokens": "required",
    "HttpPutResponseHopLimit": 1,
    "HttpEndpoint": "disabled"
  }
}
```

Para obtener más información, consulte [los metadatos de las instancias y los datos de usuario](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

Ejemplo 3: Para habilitar el IPv6 punto final de metadatos de la instancia

En el siguiente `modify-instance-metadata-options` ejemplo, se muestra cómo activar el IPv6 punto final del servicio de metadatos de la instancia.

```
aws ec2 modify-instance-metadata-options \
  --instance-id i-1234567898abcdef0 \
  --http-protocol-ipv6 enabled \
  --http-endpoint enabled
```

Salida:

```
{
  "InstanceId": "i-1234567898abcdef0",
  "InstanceMetadataOptions": {
    "State": "pending",
    "HttpTokens": "required",
    "HttpPutResponseHopLimit": 1,
    "HttpEndpoint": "enabled",
    "HttpProtocolIpv6": "enabled"
  }
}
```

De forma predeterminada, el IPv6 punto final está deshabilitado. Esto es cierto incluso si ha lanzado una instancia en una IPv6 subred exclusiva. Solo IMDS se puede acceder al IPv6 punto final para las instancias creadas en el sistema Nitro. Para obtener más información, consulte [los metadatos de las instancias y los datos de usuario](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [ModifyInstanceMetadataOptions](#) la Referencia de AWS CLI comandos.

## modify-instance-placement

En el siguiente ejemplo de código, se muestra cómo usar `modify-instance-placement`.

### AWS CLI

Ejemplo 1: Para eliminar la afinidad de una instancia con un host dedicado

El siguiente `modify-instance-placement` ejemplo elimina la afinidad de una instancia con un host dedicado y permite que se lance en cualquier host dedicado disponible en tu cuenta que admita ese tipo de instancia.

```
aws ec2 modify-instance-placement \  
  --instance-id i-0e6ddf6187EXAMPLE \  
  --affinity default
```

Salida:

```
{  
  "Return": true  
}
```

Ejemplo 2: Para establecer la afinidad entre una instancia y el host dedicado especificado

El siguiente `modify-instance-placement` ejemplo establece una relación de lanzamiento entre una instancia y un host dedicado. La instancia solo se puede ejecutar en el host dedicado especificado.

```
aws ec2 modify-instance-placement \  
  --instance-id i-0e6ddf6187EXAMPLE \  
  --affinity host \  
  --host-id i-0e6ddf6187EXAMPLE
```

Salida:

```
{  
  "Return": true  
}
```



```
}
```

Para obtener más información, consulte [Modificación de la tenencia y la afinidad de las instancias](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

Ejemplo 3: Para mover una instancia a un grupo de ubicación

En el siguiente `modify-instance-placement` ejemplo, se mueve una instancia a un grupo de ubicación, se detiene la instancia, se modifica la ubicación de la instancia y, a continuación, se reinicia la instancia.

```
aws ec2 stop-instances \  
  --instance-ids i-0123a456700123456  
  
aws ec2 modify-instance-placement \  
  --instance-id i-0123a456700123456 \  
  --group-name MySpreadGroup  
  
aws ec2 start-instances \  
  --instance-ids i-0123a456700123456
```

Para obtener más información, consulte [Cambiar el grupo de ubicación de una instancia](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

Ejemplo 4: Para eliminar una instancia de un grupo de ubicación

En el siguiente `modify-instance-placement` ejemplo, se elimina una instancia de un grupo de ubicación. Para ello, se detiene la instancia, se modifica la ubicación de la instancia y, a continuación, se reinicia la instancia. En el siguiente ejemplo, se especifica una cadena vacía («») como nombre del grupo de ubicación para indicar que la instancia no debe ubicarse en un grupo de ubicación.

Detenga la instancia:

```
aws ec2 stop-instances \  
  --instance-ids i-0123a456700123456
```

Modifique la ubicación (línea de comandos de Windows, Linux y macOS):

```
aws ec2 modify-instance-placement \  
  --instance-id i-0123a456700123456 \  
  --group-name 
```

```
--group-name ""
```

Modifique la ubicación (Windows PowerShell):

```
aws ec2 modify-instance-placement \  
  --instance-id i-0123a456700123456 \  
  --group-name ""
```

Reinicia la instancia:

```
aws ec2 start-instances \  
  --instance-ids i-0123a456700123456
```

Salida:

```
{  
  "Return": true  
}
```

Para obtener más información, consulte [Modificación de la tenencia y la afinidad de las instancias](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [ModifyInstancePlacement](#) la Referencia de AWS CLI comandos.

## modify-ipam-pool

En el siguiente ejemplo de código, se muestra cómo usar `modify-ipam-pool`.

### AWS CLI

Para modificar un IPAM grupo

En el siguiente `modify-ipam-pool` ejemplo, se modifica un IPAM grupo.

(Linux):

```
aws ec2 modify-ipam-pool \  
  --ipam-pool-id ipam-pool-0533048da7d823723 \  
  --add-allocation-resource-tags "Key=Owner,Value=Build Team" \  
  --clear-allocation-default-netmask-length \  
  --
```

```
--allocation-min-netmask-length 14
```

(Windows):

```
aws ec2 modify-ipam-pool ^
--ipam-pool-id ipam-pool-0533048da7d823723 ^
--add-allocation-resource-tags "Key=Owner,Value=Build Team" ^
--clear-allocation-default-netmask-length ^
--allocation-min-netmask-length 14
```

Salida:

```
{
  "IpamPool": {
    "OwnerId": "123456789012",
    "IpamPoolId": "ipam-pool-0533048da7d823723",
    "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0533048da7d823723",
    "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-02fc38cd4c48e7d38",
    "IpamScopeType": "private",
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
    "IpamRegion": "us-east-1",
    "Locale": "None",
    "PoolDepth": 1,
    "State": "modify-complete",
    "AutoImport": true,
    "AddressFamily": "ipv4",
    "AllocationMinNetmaskLength": 14,
    "AllocationMaxNetmaskLength": 26,
    "AllocationResourceTags": [
      {
        "Key": "Environment",
        "Value": "Preprod"
      },
      {
        "Key": "Owner",
        "Value": "Build Team"
      }
    ]
  }
}
```

Para obtener más información, consulta Cómo [editar un grupo](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [ModifyIpamPool](#) la Referencia de AWS CLI comandos.

## **modify-ipam-resource-cidr**

En el siguiente ejemplo de código, se muestra cómo usar `modify-ipam-resource-cidr`.

### AWS CLI

Para modificar lo CIDR asignado a un recurso

El siguiente `modify-ipam-resource-cidr` ejemplo modifica un recursoCIDR.

(Linux):

```
aws ec2 modify-ipam-resource-cidr \  
  --current-ipam-scope-id ipam-scope-02fc38cd4c48e7d38 \  
  --destination-ipam-scope-id ipam-scope-0da34c61fd189a141 \  
  --resource-id vpc-010e1791024eb0af9 \  
  --resource-cidr 10.0.1.0/24 \  
  --resource-region us-east-1 \  
  --monitored
```

(Windows):

```
aws ec2 modify-ipam-resource-cidr ^  
  --current-ipam-scope-id ipam-scope-02fc38cd4c48e7d38 ^  
  --destination-ipam-scope-id ipam-scope-0da34c61fd189a141 ^  
  --resource-id vpc-010e1791024eb0af9 ^  
  --resource-cidr 10.0.1.0/24 ^  
  --resource-region us-east-1 ^  
  --monitored
```

Salida:

```
{  
  "IpamResourceCidr": {  
    "IpamId": "ipam-08440e7a3acde3908",  
    "IpamScopeId": "ipam-scope-0da34c61fd189a141",
```

```
"IpamPoolId": "ipam-pool-0533048da7d823723",
"ResourceRegion": "us-east-1",
"ResourceOwnerId": "123456789012",
"ResourceId": "vpc-010e1791024eb0af9",
"ResourceCidr": "10.0.1.0/24",
"ResourceType": "vpc",
"ResourceTags": [
  {
    "Key": "Environment",
    "Value": "Preprod"
  },
  {
    "Key": "Owner",
    "Value": "Build Team"
  }
],
"IpUsage": 0.0,
"ComplianceStatus": "noncompliant",
"ManagementState": "managed",
"OverlapStatus": "overlapping",
"VpcId": "vpc-010e1791024eb0af9"
}
```

Para obtener más información sobre cómo mover recursos, consulte [Mover recursos CIDRs entre ámbitos](#) en la Guía del VPC IPAM usuario de Amazon.

Para obtener más información sobre cómo cambiar los estados de supervisión, consulte [Cambiar el estado de supervisión del recurso CIDRs](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [ModifyIpamResourceCidr](#) la Referencia de AWS CLI comandos.

## **modify-ipam-resource-discovery**

En el siguiente ejemplo de código, se muestra cómo usar `modify-ipam-resource-discovery`.

### AWS CLI

Para modificar las regiones operativas de un descubrimiento de recursos

En este ejemplo, usted es un administrador IPAM delegado que desea modificar las regiones operativas de una detección de recursos.

Para completar esta solicitud, sigue estos pasos:

No puede modificar una detección de recursos predeterminada y debe ser el propietario de la detección de recursos. Necesita el ID de detección de recursos, que puede obtener. [describe-ipam-resource-discoveries](#)

El siguiente `modify-ipam-resource-discovery` ejemplo modifica una detección de recursos no predeterminada en su cuenta. AWS

```
aws ec2 modify-ipam-resource-discovery \  
  --ipam-resource-discovery-id ipam-res-disco-0f4ef577a9f37a162 \  
  --add-operating-regions RegionName='us-west-1' \  
  --remove-operating-regions RegionName='us-east-2' \  
  --region us-east-1
```

Salida:

```
{  
  "IpamResourceDiscovery": {  
    "OwnerId": "149977607591",  
    "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",  
    "IpamResourceDiscoveryArn": "arn:aws:ec2::149977607591:ipam-resource-  
discovery/ipam-res-disco-0365d2977fc1672fe",  
    "IpamResourceDiscoveryRegion": "us-east-1",  
    "Description": "Example",  
    "OperatingRegions": [  
      {  
        "RegionName": "us-east-1"  
      },  
      {  
        "RegionName": "us-west-1"  
      }  
    ],  
    "IsDefault": false,  
    "State": "modify-in-progress"  
  }  
}
```

Para obtener más información, consulta [Cómo trabajar con descubrimientos de recursos](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [ModifyIpamResourceDiscovery](#) la Referencia de AWS CLI comandos.

## modify-ipam-scope

En el siguiente ejemplo de código, se muestra cómo usar `modify-ipam-scope`.

### AWS CLI

Para modificar la descripción de un ámbito

En este escenario, usted es un administrador IPAM delegado que desea modificar la descripción de un IPAM ámbito.

Para completar esta solicitud, necesitarás el ID del ámbito, que puedes obtener con [describe-ipam-scopes](#)él.

En el siguiente `modify-ipam-scope` ejemplo, se actualiza la descripción del ámbito.

```
aws ec2 modify-ipam-scope \  
  --ipam-scope-id ipam-scope-0d3539a30b57dcdd1 \  
  --description example \  
  --region us-east-1
```

Salida:

```
{  
  "IpamScope": {  
    "OwnerId": "320805250157",  
    "IpamScopeId": "ipam-scope-0d3539a30b57dcdd1",  
    "IpamScopeArn": "arn:aws:ec2::320805250157:ipam-scope/ipam-  
scope-0d3539a30b57dcdd1",  
    "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",  
    "IpamRegion": "us-east-1",  
    "IpamScopeType": "public",  
    "IsDefault": true,  
    "Description": "example",  
    "PoolCount": 1,  
    "State": "modify-in-progress"  
  }  
}
```

Para obtener más información sobre los ámbitos, consulta [Cómo IPAM funciona](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [ModifyIpamScope](#) la Referencia de AWS CLI comandos.

## modify-ipam

En el siguiente ejemplo de código, se muestra cómo usar `modify-ipam`.

### AWS CLI

Para modificar un IPAM

En el siguiente `modify-ipam` ejemplo se modifica y IPAM se añade una región operativa.

(Linux):

```
aws ec2 modify-ipam \  
  --ipam-id ipam-08440e7a3acde3908 \  
  --add-operating-regions RegionName=us-west-2
```

(Windows):

```
aws ec2 modify-ipam ^  
  --ipam-id ipam-08440e7a3acde3908 ^  
  --add-operating-regions RegionName=us-west-2
```

Salida:

```
{  
  "Ipam": {  
    "OwnerId": "123456789012",  
    "IpamId": "ipam-08440e7a3acde3908",  
    "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",  
    "IpamRegion": "us-east-1",  
    "PublicDefaultScopeId": "ipam-scope-0b9eed026396dbc16",  
    "PrivateDefaultScopeId": "ipam-scope-02fc38cd4c48e7d38",  
    "ScopeCount": 3,  
    "OperatingRegions": [  
      {  
        "RegionName": "us-east-1"  
      },  
      {
```



```

        "RegionName": "us-east-2"
      },
      {
        "RegionName": "us-west-1"
      },
      {
        "RegionName": "us-west-2"
      }
    ],
    "State": "modify-in-progress"
  }
}

```

- Para API obtener más información, consulte [ModifyIpam](#) la Referencia de AWS CLI comandos.

## modify-launch-template

En el siguiente ejemplo de código, se muestra cómo usar `modify-launch-template`.

### AWS CLI

Para cambiar la versión por defecto de la plantilla de lanzamiento

En este ejemplo, se especifica la versión 2 de la plantilla de lanzamiento especificada como versión predeterminada.

Comando:

```
aws ec2 modify-launch-template --launch-template-id lt-0abcd290751193123 --default-version 2
```

Salida:

```

{
  "LaunchTemplate": {
    "LatestVersionNumber": 2,
    "LaunchTemplateId": "lt-0abcd290751193123",
    "LaunchTemplateName": "WebServers",
    "DefaultVersionNumber": 2,
    "CreatedBy": "arn:aws:iam::123456789012:root",
    "CreateTime": "2017-12-01T13:35:46.000Z"
  }
}

```

```
}
```

- Para API obtener más información, consulte [ModifyLaunchTemplate](#) la Referencia de AWS CLI comandos.

## modify-managed-prefix-list

En el siguiente ejemplo de código, se muestra cómo usar `modify-managed-prefix-list`.

### AWS CLI

Para modificar una lista de prefijos

El siguiente `modify-managed-prefix-list` ejemplo agrega una entrada a la lista de prefijos especificada.

```
aws ec2 modify-managed-prefix-list \  
  --prefix-list-id pl-0123456abcabcabc1 \  
  --add-entries Cidr=10.1.0.0/16,Description=vpc-c \  
  --current-version 1
```

Salida:

```
{  
  "PrefixList": {  
    "PrefixListId": "pl-0123456abcabcabc1",  
    "AddressFamily": "IPv4",  
    "State": "modify-in-progress",  
    "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/  
pl-0123456abcabcabc1",  
    "PrefixListName": "vpc-cidrs",  
    "MaxEntries": 10,  
    "Version": 1,  
    "OwnerId": "123456789012"  
  }  
}
```

Para obtener más información, consulta [Listas de prefijos gestionadas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [ModifyManagedPrefixList](#) la Referencia de AWS CLI comandos.

## modify-network-interface-attribute

En el siguiente ejemplo de código, se muestra cómo usar `modify-network-interface-attribute`.

### AWS CLI

Para modificar el atributo de adjunto de una interfaz de red

Este comando de ejemplo modifica el `attachment` atributo de la interfaz de red especificada.

Comando:

```
aws ec2 modify-network-interface-attribute --network-interface-id eni-686ea200 --attachment AttachmentId=eni-attach-43348162,DeleteOnTermination=false
```

Para modificar el atributo de descripción de una interfaz de red

Este comando de ejemplo modifica el `description` atributo de la interfaz de red especificada.

Comando:

```
aws ec2 modify-network-interface-attribute --network-interface-id eni-686ea200 --description "My description"
```

Para modificar el `groupSet` atributo de una interfaz de red

Este comando de ejemplo modifica el `groupSet` atributo de la interfaz de red especificada.

Comando:

```
aws ec2 modify-network-interface-attribute --network-interface-id eni-686ea200 --groups sg-903004f8 sg-1a2b3c4d
```

Para modificar el `sourceDestCheck` atributo de una interfaz de red

Este comando de ejemplo modifica el `sourceDestCheck` atributo de la interfaz de red especificada.

Comando:

```
aws ec2 modify-network-interface-attribute --network-interface-id eni-686ea200 --no-source-dest-check
```

- Para API obtener más información, consulte [ModifyNetworkInterfaceAttribute](#) la Referencia de AWS CLI comandos.

## modify-private-dns-name-options

En el siguiente ejemplo de código, se muestra cómo usar `modify-private-dns-name-options`.

### AWS CLI

Para modificar las opciones (por ejemplo, los nombres de host)

En el siguiente `modify-private-dns-name-options` ejemplo, se deshabilita la opción de responder a DNS consultas (por ejemplo, nombres de host) con registros A. DNS

```
aws ec2 modify-private-dns-name-options \  
  --instance-id i-1234567890abcdef0 \  
  --no-enable-resource-name-dns-a-record
```

Salida:

```
{  
  "Return": true  
}
```

Para obtener más información, consulta los [tipos de nombres de host de las EC2 instancias de Amazon](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [ModifyPrivateDnsNameOptions](#) la Referencia de AWS CLI comandos.

## modify-reserved-instances

En el siguiente ejemplo de código, se muestra cómo usar `modify-reserved-instances`.

### AWS CLI

Para modificar las instancias reservadas

Este comando de ejemplo mueve una instancia reservada a otra zona de disponibilidad en la misma región.

Comando:

```
aws ec2 modify-reserved-instances --reserved-instances-ids b847fa93-e282-4f55-b59a-1342f5bd7c02 --target-configurations AvailabilityZone=us-west-1c,Platform=EC2-Classic,InstanceCount=10
```

Salida:

```
{  
  "ReservedInstancesModificationId": "rimod-d3ed4335-b1d3-4de6-ab31-0f13aaf46687"  
}
```

Para modificar la plataforma de red de las instancias reservadas

Este comando de ejemplo convierte las instancias EC2 reservadas clásicas en EC2 -VPC.

Comando:

```
aws ec2 modify-reserved-instances --reserved-instances-ids f127bd27-edb7-44c9-a0eb-0d7e09259af0 --target-configurations AvailabilityZone=us-west-1c,Platform=EC2-VPC,InstanceCount=5
```

Salida:

```
{  
  "ReservedInstancesModificationId": "rimod-82fa9020-668f-4fb6-945d-61537009d291"  
}
```

Para obtener más información, consulte [Modificación de instancias reservadas](#) en la Guía del EC2 usuario de Amazon.

Para modificar el tamaño de las instancias reservadas

Este comando de ejemplo modifica una instancia reservada que tiene 10 instancias m1.small Linux/ en us-west-1c, de modo que 8 UNIX instancias m1.small se convierten en 2 instancias m1.large y las 2 m1.small restantes se convierten en 1 m1.medium en la misma zona de disponibilidad. Comando:

```
aws ec2 modify-reserved-instances --reserved-instances-ids 1ba8e2e3-3556-4264-949e-63ee671405a9 --target-configurations AvailabilityZone=us-west-1c,Platform=EC2-Classic,InstanceCount=2,InstanceType=m1.large AvailabilityZone=us-west-1c,Platform=EC2-Classic,InstanceCount=1,InstanceType=m1.medium
```

Salida:

```
{
  "ReservedInstancesModificationId": "rimod-acc5f240-080d-4717-b3e3-1c6b11fa00b6"
}
```

Para obtener más información, consulta [Modificación del tamaño de instancia de tus reservas](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [ModifyReservedInstances](#) la Referencia de AWS CLI comandos.

## modify-security-group-rules

En el siguiente ejemplo de código, se muestra cómo usar `modify-security-group-rules`.

### AWS CLI

Para modificar las reglas de un grupo de seguridad para actualizar la descripción de la regla, el protocolo IP y el rango de CidrIpv4 direcciones

El siguiente `modify-security-group-rules` ejemplo actualiza la descripción, el protocolo IP y el IPV4 CIDR rango de una regla de grupo de seguridad especificada. Utilice el `security-group-rules` parámetro para introducir las actualizaciones de las reglas del grupo de seguridad especificado. `-1` especifica todos los protocolos.

```
aws ec2 modify-security-group-rules \
  --group-id sg-1234567890abcdef0 \
  --security-group-rules SecurityGroupId=sg-abcdef01234567890,SecurityGroupRule='{Description=test,IpProtocol=-1,CidrIpv4=0.0.0.0/0}'
```

Salida:

```
{
```

```
"Return": true
}
```

Para obtener más información sobre las reglas de los grupos de [seguridad, consulte Reglas de grupos](#) de seguridad en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [ModifySecurityGroupRules](#) la Referencia de AWS CLI comandos.

## modify-snapshot-attribute

En el siguiente ejemplo de código, se muestra cómo usar `modify-snapshot-attribute`.

### AWS CLI

Ejemplo 1: Para modificar un atributo de instantánea

En el siguiente `modify-snapshot-attribute` ejemplo, se actualiza el `createVolumePermission` atributo de la instantánea especificada y se eliminan los permisos de volumen del usuario especificado.

```
aws ec2 modify-snapshot-attribute \  
  --snapshot-id snap-1234567890abcdef0 \  
  --attribute createVolumePermission \  
  --operation-type remove \  
  --user-ids 123456789012
```

Ejemplo 2: Hacer pública una instantánea

En el siguiente `modify-snapshot-attribute` ejemplo, se hace pública la instantánea especificada.

```
aws ec2 modify-snapshot-attribute \  
  --snapshot-id snap-1234567890abcdef0 \  
  --attribute createVolumePermission \  
  --operation-type add \  
  --group-names all
```

- Para API obtener más información, consulte [ModifySnapshotAttribute](#) la Referencia de AWS CLI comandos.

## modify-snapshot-tier

En el siguiente ejemplo de código, se muestra cómo usar `modify-snapshot-tier`.

### AWS CLI

Ejemplo 1: Para archivar una instantánea

En el siguiente `modify-snapshot-tier` ejemplo, se archiva la instantánea especificada.

```
aws ec2 modify-snapshot-tier \  
  --snapshot-id snap-01234567890abcdef \  
  --storage-tier archive
```

Salida:

```
{  
  "SnapshotId": "snap-01234567890abcdef",  
  "TieringStartTime": "2021-09-15T16:44:37.574Z"  
}
```

El parámetro de `TieringStartTime` respuesta indica la fecha y la hora en que se inició el proceso de archivado, en formato de UTC hora (YYYY-MM DDTHH -:MM:).SSZ

Para obtener más información sobre el archivado de instantáneas, consulte [Archivar EBS instantáneas de Amazon](#) en la Guía EC2 del usuario de Amazon.

- Para API obtener más información, consulte la Referencia [ModifySnapshotTier](#) de AWS CLI comandos.

## modify-spot-fleet-request

En el siguiente ejemplo de código, se muestra cómo usar `modify-spot-fleet-request`.

### AWS CLI

Para modificar una solicitud de flota de Spot

Este comando de ejemplo actualiza la capacidad objetivo de la solicitud de flota de Spot especificada.

Comando:



```
aws ec2 modify-spot-fleet-request --target-capacity 20 --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Salida:

```
{
  "Return": true
}
```

Este comando de ejemplo reduce la capacidad objetivo de la solicitud de flota de spot especificada sin cerrar, por lo tanto, ninguna instancia de spot.

Comando:

```
aws ec2 modify-spot-fleet-request --target-capacity 10 --excess-capacity-termination-policy NoTermination --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Salida:

```
{
  "Return": true
}
```

- Para API obtener más información, consulte [ModifySpotFleetRequest](#) la Referencia de AWS CLI comandos.

## modify-subnet-attribute

En el siguiente ejemplo de código, se muestra cómo usar `modify-subnet-attribute`.

AWS CLI

Para cambiar el comportamiento del IPv4 direccionamiento público de una subred

En este ejemplo, se modifica la subnet-1a2b3c4d para especificar que se asigne una dirección pública a todas las instancias lanzadas en esta subred. IPv4 Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 modify-subnet-attribute --subnet-id subnet-1a2b3c4d --map-public-ip-on-launch
```

Para cambiar el comportamiento de IPv6 direccionamiento de una subred

En este ejemplo, se modifica la subnet-1a2b3c4d para especificar que a todas las instancias lanzadas en esta subred se les asigne una dirección del rango de la subred. IPv6

Comando:

```
aws ec2 modify-subnet-attribute --subnet-id subnet-1a2b3c4d --assign-ipv6-address-on-creation
```

Para obtener más información, consulte Direcciones IP en la guía del usuario de Your in the Virtual Private Cloud. VPC AWS

- Para API obtener más información, consulte [ModifySubnetAttribute](#) la Referencia de AWS CLI comandos.

## modify-traffic-mirror-filter-network-services

En el siguiente ejemplo de código, se muestra cómo usar modify-traffic-mirror-filter-network-services.

AWS CLI

Para añadir servicios de red a un filtro de Traffic Mirror

En el siguiente modify-traffic-mirror-filter-network-services ejemplo, se añaden los servicios DNS de red de Amazon al filtro especificado.

```
aws ec2 modify-traffic-mirror-filter-network-services \  
  --traffic-mirror-filter-id tmf-04812ff784EXAMPLE \  
  --add-network-service amazon-dns
```

Salida:

```
{  
  "TrafficMirrorFilter": {  
    "Tags": [  

```

```
    {
      "Key": "Name",
      "Value": "Production"
    }
  ],
  "EgressFilterRules": [],
  "NetworkServices": [
    "amazon-dns"
  ],
  "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
  "IngressFilterRules": [
    {
      "SourceCidrBlock": "0.0.0.0/0",
      "RuleNumber": 1,
      "DestinationCidrBlock": "0.0.0.0/0",
      "Description": "TCP Rule",
      "Protocol": 6,
      "TrafficDirection": "ingress",
      "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
      "RuleAction": "accept",
      "TrafficMirrorFilterRuleId": "tmf-04812ff784EXAMPLE"
    }
  ]
}
```

Para obtener más información, consulte [Modificar los servicios de red del filtro espejo](#) de tráfico en la AWS Guía de duplicación de tráfico.

- Para API obtener más información, consulte la Referencia [ModifyTrafficMirrorFilterNetworkServices](#) de AWS CLI comandos.

## **modify-traffic-mirror-filter-rule**

En el siguiente ejemplo de código, se muestra cómo usar `modify-traffic-mirror-filter-rule`.

### AWS CLI

Para modificar una regla de filtrado de réplicas de tráfico

En el siguiente `modify-traffic-mirror-filter-rule` ejemplo, se modifica la descripción de la regla de filtrado de reflejo de tráfico especificada.

```
aws ec2 modify-traffic-mirror-filter-rule \  
  --traffic-mirror-filter-rule-id tmfr-0ca76e0e08EXAMPLE \  
  --description "TCP Rule"
```

Salida:

```
{  
  "TrafficMirrorFilterRule": {  
    "TrafficMirrorFilterRuleId": "tmfr-0ca76e0e08EXAMPLE",  
    "TrafficMirrorFilterId": "tmf-0293f26e86EXAMPLE",  
    "TrafficDirection": "ingress",  
    "RuleNumber": 100,  
    "RuleAction": "accept",  
    "Protocol": 6,  
    "DestinationCidrBlock": "10.0.0.0/24",  
    "SourceCidrBlock": "10.0.0.0/24",  
    "Description": "TCP Rule"  
  }  
}
```

Para obtener más información, consulte [Modificar las reglas del filtro de espejo de tráfico](#) en la Guía de duplicación de AWS tráfico.

- Para API obtener más información, consulte la Referencia [ModifyTrafficMirrorFilterRule](#) de AWS CLI comandos.

## modify-traffic-mirror-session

En el siguiente ejemplo de código, se muestra cómo usar `modify-traffic-mirror-session`.

### AWS CLI

Para modificar una sesión duplicada de tráfico

En el siguiente `modify-traffic-mirror-session` ejemplo, se cambia la descripción de la sesión de duplicación de tráfico y la cantidad de paquetes que se van a duplicar.

```
aws ec2 modify-traffic-mirror-session \  
  --description "Change packet length" \  
  --traffic-mirror-session-id tms-08a33b1214EXAMPLE \  
  --remove-fields "packet-length"
```

Salida:

```
{
  "TrafficMirrorSession": {
    "TrafficMirrorSessionId": "tms-08a33b1214EXAMPLE",
    "TrafficMirrorTargetId": "tmt-07f75d8feeEXAMPLE",
    "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
    "NetworkInterfaceId": "eni-070203f901EXAMPLE",
    "OwnerId": "111122223333",
    "SessionNumber": 1,
    "VirtualNetworkId": 7159709,
    "Description": "Change packet length",
    "Tags": []
  }
}
```

Para obtener más información, consulte [Modificar la sesión de duplicación de tráfico](#) en la Guía de duplicación de tráfico.

- Para API obtener más información, consulte la Referencia [ModifyTrafficMirrorSession](#) de AWS CLI comandos.

## modify-transit-gateway-prefix-list-reference

En el siguiente ejemplo de código, se muestra cómo usar `modify-transit-gateway-prefix-list-reference`.

AWS CLI

Para modificar una referencia a una lista de prefijos

El siguiente `modify-transit-gateway-prefix-list-reference` ejemplo modifica la referencia a la lista de prefijos en la tabla de rutas especificada cambiando el adjunto al que se enruta el tráfico.

```
aws ec2 modify-transit-gateway-prefix-list-reference \
  --transit-gateway-route-table-id tgw-rtb-0123456789abcd123 \
  --prefix-list-id pl-11111122222222333 \
  --transit-gateway-attachment-id tgw-attach-aabbccddaabbccaab
```

Salida:

```
{
  "TransitGatewayPrefixListReference": {
    "TransitGatewayRouteTableId": "tgw-rtb-0123456789abcd123",
    "PrefixListId": "pl-1111112222222333",
    "PrefixListOwnerId": "123456789012",
    "State": "modifying",
    "Blackhole": false,
    "TransitGatewayAttachment": {
      "TransitGatewayAttachmentId": "tgw-attach-aabbccddaabbccaab",
      "ResourceType": "vpc",
      "ResourceId": "vpc-112233445566aabbcc"
    }
  }
}
```

Para obtener más información, consulte las [referencias a las listas de prefijos](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [ModifyTransitGatewayPrefixListReference](#) de AWS CLI comandos.

## modify-transit-gateway-vpc-attachment

En el siguiente ejemplo de código, se muestra cómo usar `modify-transit-gateway-vpc-attachment`.

### AWS CLI

Para modificar un VPC adjunto a una pasarela de tránsito

El siguiente `modify-transit-gateway-vpc-attachment` ejemplo agrega una subred al VPC adjunto de la puerta de enlace de tránsito especificado.

```
aws ec2 modify-transit-gateway-vpc-attachment \
  --transit-gateway-attachment-id tgw-attach-09fbd47ddfEXAMPLE \
  --add-subnet-ids subnet-0e51f45802EXAMPLE
```

Salida:

```
{
  "TransitGatewayVpcAttachment": {
    "TransitGatewayAttachmentId": "tgw-attach-09fbd47ddfEXAMPLE",
```

```

    "TransitGatewayId": "tgw-0560315ccfEXAMPLE",
    "VpcId": "vpc-5eccc927",
    "VpcOwnerId": "111122223333",
    "State": "modifying",
    "SubnetIds": [
      "subnet-0e51f45802EXAMPLE",
      "subnet-1EXAMPLE"
    ],
    "CreationTime": "2019-08-08T16:47:38.000Z",
    "Options": {
      "DnsSupport": "enable",
      "Ipv6Support": "disable"
    }
  }
}

```

Para obtener más información, consulte los [adjuntos de Transit Gateways a una VPC](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [ModifyTransitGatewayVpcAttachment](#) la Referencia de AWS CLI comandos.

## modify-transit-gateway

En el siguiente ejemplo de código, se muestra cómo usar `modify-transit-gateway`.

### AWS CLI

Para modificar una puerta de enlace de tránsito

El siguiente `modify-transit-gateway` ejemplo modifica la puerta de enlace de tránsito especificada al habilitar la ECMP compatibilidad con los VPN archivos adjuntos.

```

aws ec2 modify-transit-gateway \
  --transit-gateway-id tgw-11111222222aaaaa \
  --options VpnEcmpSupport=enable

```

Salida:

```

{
  "TransitGateway": {
    "TransitGatewayId": "tgw-11111222222aaaaa",

```

```
"TransitGatewayArn": "64512",
"State": "modifying",
"OwnerId": "123456789012",
"CreationTime": "2020-04-30T08:41:37.000Z",
"Options": {
  "AmazonSideAsn": 64512,
  "AutoAcceptSharedAttachments": "disable",
  "DefaultRouteTableAssociation": "enable",
  "AssociationDefaultRouteTableId": "tgw-rtb-0123456789abcd123",
  "DefaultRouteTablePropagation": "enable",
  "PropagationDefaultRouteTableId": "tgw-rtb-0123456789abcd123",
  "VpnEcmpSupport": "enable",
  "DnsSupport": "enable"
}
}
```

Para obtener más información, consulte [Pasarelas de tránsito en la Guía](#) de pasarelas de tránsito.

- Para API obtener más información, consulte la Referencia [ModifyTransitGateway](#) de AWS CLI comandos.

## modify-verified-access-endpoint-policy

En el siguiente ejemplo de código, se muestra cómo usar `modify-verified-access-endpoint-policy`.

### AWS CLI

Para configurar la política de acceso verificado para un punto final

En el siguiente `modify-verified-access-endpoint-policy` ejemplo, se agrega la política de acceso verificado especificada al punto final de acceso verificado especificado.

```
aws ec2 modify-verified-access-endpoint-policy \
  --verified-access-endpoint-id vae-066fac616d4d546f2 \
  --policy-enabled \
  --policy-document file://policy.txt
```

Contenidos de `policy.txt`:

```
permit(principal,action,resource)
```



```
when {
  context.identity.groups.contains("finance") &&
  context.identity.email.verified == true
};
```

Salida:

```
{
  "PolicyEnabled": true,
  "PolicyDocument": "permit(principal,action,resource)\nwhen
{\n  context.identity.groups.contains(\"finance\") &&\n
context.identity.email_verified == true\n};"
}
```

Para obtener más información, consulte [las políticas de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [ModifyVerifiedAccessEndpointPolicy](#) la Referencia de AWS CLI comandos.

## modify-verified-access-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `modify-verified-access-endpoint`.

### AWS CLI

Para modificar la configuración de un punto final de acceso verificado

En el siguiente `modify-verified-access-endpoint` ejemplo, se agrega la descripción especificada al punto final de Verified Access especificado.

```
aws ec2 modify-verified-access-endpoint \
  --verified-access-endpoint-id vae-066fac616d4d546f2 \
  --description "Testing Verified Access"
```

Salida:

```
{
  "VerifiedAccessEndpoint": {
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
```

```

    "VerifiedAccessEndpointId": "vae-066fac616d4d546f2",
    "ApplicationDomain": "example.com",
    "EndpointType": "network-interface",
    "AttachmentType": "vpc",
    "DomainCertificateArn": "arn:aws:acm:us-east-2:123456789012:certificate/
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE",
    "EndpointDomain": "my-ava-
app.edge-00c3372d53b1540bb.vai-0ce000c0b7643abea.prod.verified-access.us-
east-2.amazonaws.com",
    "SecurityGroupIds": [
        "sg-004915970c4c8f13a"
    ],
    "NetworkInterfaceOptions": {
        "NetworkInterfaceId": "eni-0aec70418c8d87a0f",
        "Protocol": "https",
        "Port": 443
    },
    "Status": {
        "Code": "updating"
    },
    "Description": "Testing Verified Access",
    "CreationTime": "2023-08-25T20:54:43",
    "LastUpdatedTime": "2023-08-25T22:46:32"
}
}

```

Para obtener más información, consulte los [puntos finales de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [ModifyVerifiedAccessEndpoint](#) la Referencia de AWS CLI comandos.

## **modify-verified-access-group-policy**

En el siguiente ejemplo de código, se muestra cómo usar `modify-verified-access-group-policy`.

### AWS CLI

Para configurar una política de acceso verificado para un grupo

En el siguiente `modify-verified-access-group-policy` ejemplo, se agrega la política de acceso verificado especificada al grupo de acceso verificado especificado.

```
aws ec2 modify-verified-access-group-policy \  
  --verified-access-group-id vagr-0dbe967baf14b7235 \  
  --policy-enabled \  
  --policy-document file://policy.txt
```

Contenidos de `policy.txt`:

```
permit(principal,action,resource)  
when {  
  context.identity.groups.contains("finance") &&  
  context.identity.email.verified == true  
};
```

Salida:

```
{  
  "PolicyEnabled": true,  
  "PolicyDocument": "permit(principal,action,resource)\nwhen  
{\n  context.identity.groups.contains(\"finance\") &&\n  context.identity.email_verified == true\n};"  
}
```

Para obtener más información, consulte los [grupos de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [ModifyVerifiedAccessGroupPolicy](#) la Referencia de AWS CLI comandos.

## **modify-verified-access-group**

En el siguiente ejemplo de código, se muestra cómo usar `modify-verified-access-group`.

### AWS CLI

Para modificar la configuración de un grupo de acceso verificado

En el siguiente `modify-verified-access-group` ejemplo, se agrega la descripción especificada al grupo de acceso verificado especificado.

```
aws ec2 modify-verified-access-group \  
  --verified-access-group-id vagr-0dbe967baf14b7235 \  
  --policy-enabled \  
  --policy-document file://policy.txt
```

```
--verified-access-group-id vagr-0dbe967baf14b7235 \  
--description "Testing Verified Access"
```

Salida:

```
{  
  "VerifiedAccessGroup": {  
    "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",  
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",  
    "Description": "Testing Verified Access",  
    "Owner": "123456789012",  
    "VerifiedAccessGroupArn": "arn:aws:ec2:us-east-2:123456789012:verified-  
access-group/vagr-0dbe967baf14b7235",  
    "CreationTime": "2023-08-25T19:55:19",  
    "LastUpdatedTime": "2023-08-25T22:17:25"  
  }  
}
```

Para obtener más información, consulte los [grupos de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [ModifyVerifiedAccessGroup](#) la Referencia de AWS CLI comandos.

## modify-verified-access-instance-logging-configuration

En el siguiente ejemplo de código, se muestra cómo usar `modify-verified-access-instance-logging-configuration`.

AWS CLI

Para habilitar el registro de una instancia de acceso verificado

El siguiente `modify-verified-access-instance-logging-configuration` ejemplo habilita el registro de acceso para la instancia de acceso verificado especificada. Los registros se entregarán al grupo de CloudWatch registros especificado.

```
aws ec2 modify-verified-access-instance-logging-configuration \  
--verified-access-instance-id vai-0ce000c0b7643abea \  
--access-logs CloudWatchLogs={Enabled=true,LogGroup=my-log-group}
```

**Salida:**

```
{
  "LoggingConfiguration": {
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
    "AccessLogs": {
      "S3": {
        "Enabled": false
      },
      "CloudWatchLogs": {
        "Enabled": true,
        "DeliveryStatus": {
          "Code": "success"
        },
        "LogGroup": "my-log-group"
      },
      "KinesisDataFirehose": {
        "Enabled": false
      },
      "LogVersion": "ocsf-1.0.0-rc.2",
      "IncludeTrustContext": false
    }
  }
}
```

Para obtener más información, consulte [los registros de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [ModifyVerifiedAccessInstanceLoggingConfiguration](#) la Referencia de AWS CLI comandos.

**modify-verified-access-instance**

En el siguiente ejemplo de código, se muestra cómo usar `modify-verified-access-instance`.

**AWS CLI**

Para modificar la configuración de una instancia de acceso verificado

En el siguiente `modify-verified-access-instance` ejemplo, se agrega la descripción especificada a la instancia de acceso verificado especificada.

```
aws ec2 modify-verified-access-instance \
```

```
--verified-access-instance-id vai-0ce000c0b7643abea \  
--description "Testing Verified Access"
```

Salida:

```
{  
  "VerifiedAccessInstance": {  
    "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",  
    "Description": "Testing Verified Access",  
    "VerifiedAccessTrustProviders": [  
      {  
        "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",  
        "TrustProviderType": "user",  
        "UserTrustProviderType": "iam-identity-center"  
      }  
    ],  
    "CreationTime": "2023-08-25T18:27:56",  
    "LastUpdatedTime": "2023-08-25T22:41:04"  
  }  
}
```

Para obtener más información, consulte las [instancias de acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [ModifyVerifiedAccessInstance](#) la Referencia de AWS CLI comandos.

## modify-verified-access-trust-provider

En el siguiente ejemplo de código, se muestra cómo usar `modify-verified-access-trust-provider`.

AWS CLI

Para modificar la configuración de un proveedor de confianza de acceso verificado

En el siguiente `modify-verified-access-trust-provider` ejemplo, se agrega la descripción especificada al proveedor de confianza de Verified Access especificado.

```
aws ec2 modify-verified-access-trust-provider \  
--verified-access-trust-provider-id vatp-0bb32de759a3e19e7 \  
--description "Testing Verified Access"
```

```
--description "Testing Verified Access"
```

Salida:

```
{
  "VerifiedAccessTrustProvider": {
    "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
    "Description": "Testing Verified Access",
    "TrustProviderType": "user",
    "UserTrustProviderType": "iam-identity-center",
    "PolicyReferenceName": "idc",
    "CreationTime": "2023-08-25T19:00:38",
    "LastUpdatedTime": "2023-08-25T19:18:21"
  }
}
```

Para obtener más información, consulte [Proveedores de confianza para el acceso verificado](#) en la Guía del usuario de acceso AWS verificado.

- Para API obtener más información, consulte [ModifyVerifiedAccessTrustProvider](#) la Referencia de AWS CLI comandos.

## modify-volume-attribute

En el siguiente ejemplo de código, se muestra cómo usar `modify-volume-attribute`.

AWS CLI

Para modificar un atributo de volumen

En este ejemplo se establece el `autoEnableIo` atributo del volumen con el `vol-1234567890abcdef0` identificador en `true`. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 modify-volume-attribute --volume-id vol-1234567890abcdef0 --auto-enable-io
```

- Para API obtener más información, consulte [ModifyVolumeAttribute](#) la Referencia de AWS CLI comandos.

## modify-volume

En el siguiente ejemplo de código, se muestra cómo usar `modify-volume`.

### AWS CLI

Ejemplo 1: modificar un volumen cambiando su tamaño

En el siguiente `modify-volume` ejemplo, se cambia el tamaño del volumen especificado a 150 GB.

Comando:

```
aws ec2 modify-volume --size 150 --volume-id vol-1234567890abcdef0
```

Salida:

```
{
  "VolumeModification": {
    "TargetSize": 150,
    "TargetVolumeType": "io1",
    "ModificationState": "modifying",
    "VolumeId": " vol-1234567890abcdef0",
    "TargetIops": 100,
    "StartTime": "2019-05-17T11:27:19.000Z",
    "Progress": 0,
    "OriginalVolumeType": "io1",
    "OriginalIops": 100,
    "OriginalSize": 100
  }
}
```

Ejemplo 2: modificar un volumen cambiando su tipo, tamaño y valor IOPS

El siguiente `modify-volume` ejemplo cambia el tipo de volumen a Provisionado IOPSSSD, establece la IOPS velocidad objetivo en 10000 y establece el tamaño del volumen en 350 GB.

```
aws ec2 modify-volume \
  --volume-type io1 \
  --iops 10000 \
  --size 350 \
```



```
--volume-id vol-1234567890abcdef0
```

Salida:

```
{
  "VolumeModification": {
    "TargetSize": 350,
    "TargetVolumeType": "io1",
    "ModificationState": "modifying",
    "VolumeId": "vol-0721c1a9d08c93bf6",
    "TargetIops": 10000,
    "StartTime": "2019-05-17T11:38:57.000Z",
    "Progress": 0,
    "OriginalVolumeType": "gp2",
    "OriginalIops": 150,
    "OriginalSize": 50
  }
}
```

- Para API obtener más información, consulte la Referencia [ModifyVolume](#) de AWS CLI comandos.

## modify-vpc-attribute

En el siguiente ejemplo de código, se muestra cómo usar `modify-vpc-attribute`.

### AWS CLI

Para modificar el `enableDnsSupport` atributo

En este ejemplo se modifica el `enableDnsSupport` atributo. Este atributo indica si DNS la resolución está habilitada para VPC. Si este atributo lo establece en `true`, el DNS servidor de Amazon resuelve DNS los nombres de host de tus instancias en sus direcciones IP correspondientes; de lo contrario, no lo hace. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 modify-vpc-attribute --vpc-id vpc-a01106c2 --enable-dns-support '{"Value\n":false}'
```

Para modificar el atributo `enableDnsHostnames`

En este ejemplo se modifica el `enableDnsHostnames` atributo. Este atributo indica si las instancias se lanzaron en los nombres de DNS host de VPC get. Si este atributo lo establece en `true`, las instancias están en el comando VPC get DNS hostnames; de lo contrario, no lo hacen. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 modify-vpc-attribute --vpc-id vpc-a01106c2 --enable-dns-hostnames "{\"Value\":false}"
```

- Para API obtener más información, consulte [ModifyVpcAttribute](#) la Referencia de AWS CLI comandos.

## **modify-vpc-endpoint-connection-notification**

En el siguiente ejemplo de código, se muestra cómo usar `modify-vpc-endpoint-connection-notification`.

AWS CLI

Para modificar una notificación de conexión de punto final

En este ejemplo, se cambia el SNS tema de la notificación de conexión de punto final especificada.

Comando:

```
aws ec2 modify-vpc-endpoint-connection-notification --connection-notification-id vpce-nfn-008776de7e03f5abc --connection-events Accept Reject --connection-notification-arn arn:aws:sns:us-east-2:123456789012:mytopic
```

Salida:

```
{
  "ReturnValue": true
}
```

- Para API obtener más información, consulte [ModifyVpcEndpointConnectionNotification](#) la Referencia de AWS CLI comandos.

## modify-vpc-endpoint-service-configuration

En el siguiente ejemplo de código, se muestra cómo usar `modify-vpc-endpoint-service-configuration`.

### AWS CLI

Para modificar la configuración de un servicio de punto final

En este ejemplo, se cambia el requisito de aceptación del servicio de punto final especificado.

Comando:

```
aws ec2 modify-vpc-endpoint-service-configuration --service-id vpce-svc-09222513e6e77dc86 --no-acceptance-required
```

Salida:

```
{
  "ReturnValue": true
}
```

- Para API obtener más información, consulte [ModifyVpcEndpointServiceConfiguration](#) la Referencia de AWS CLI comandos.

## modify-vpc-endpoint-service-payer-responsibility

En el siguiente ejemplo de código, se muestra cómo usar `modify-vpc-endpoint-service-payer-responsibility`.

### AWS CLI

Para modificar la responsabilidad del pagador

El siguiente `modify-vpc-endpoint-service-payer-responsibility` ejemplo modifica la responsabilidad del pagador del servicio de punto final especificado.

```
aws ec2 modify-vpc-endpoint-service-payer-responsibility \
  --service-id vpce-svc-071afff70666e61e0 \
  --payer-responsibility ServiceOwner
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte la Referencia [ModifyVpcEndpointServicePayerResponsibility](#) de AWS CLI comandos.

## modify-vpc-endpoint-service-permissions

En el siguiente ejemplo de código, se muestra cómo usar `modify-vpc-endpoint-service-permissions`.

### AWS CLI

Para modificar los permisos de Endpoint Service

En este ejemplo, se agrega el permiso para que una AWS cuenta se conecte al servicio de punto final especificado.

Comando:

```
aws ec2 modify-vpc-endpoint-service-permissions --service-id vpce-svc-03d5ebb7d9579a2b3 --add-allowed-principals '["arn:aws:iam::123456789012:root"]'
```

Salida:

```
{
  "ReturnValue": true
}
```

En este ejemplo, se agrega permiso para que un IAM usuario específico (`admin`) se conecte al servicio de punto final especificado.

Comando:

```
aws ec2 modify-vpc-endpoint-service-permissions --service-id vpce-svc-03d5ebb7d9579a2b3 --add-allowed-principals '["arn:aws:iam::123456789012:user/admin"]'
```

- Para API obtener más información, consulte [ModifyVpcEndpointServicePermissions](#) la Referencia de AWS CLI comandos.

## modify-vpc-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `modify-vpc-endpoint`.

## AWS CLI

Para modificar un punto final de puerta de enlace

En este ejemplo, se modifica el punto final de la puerta de enlace `vpce-1a2b3c4d` asociando la tabla `rtb-aaa222bb` de rutas al punto final y restableciendo el documento de política.

Comando:

```
aws ec2 modify-vpc-endpoint --vpc-endpoint-id vpce-1a2b3c4d --add-route-table-ids rtb-aaa222bb --reset-policy
```

Salida:

```
{
  "Return": true
}
```

Para modificar un punto final de interfaz

En este ejemplo, se modifica el punto final de la interfaz `vpce-0fe5b17a0707d6fa5` añadiendo una subred `subnet-d6fcaa8d` al punto final.

Comando:

```
aws ec2 modify-vpc-endpoint --vpc-endpoint-id vpce-0fe5b17a0707d6fa5 --add-subnet-id subnet-d6fcaa8d
```

Salida:

```
{
  "Return": true
}
```

- Para API obtener más información, consulte la Referencia [ModifyVpcEndpoint](#) de AWS CLI comandos.

### **modify-vpc-peering-connection-options**

En el siguiente ejemplo de código, se muestra cómo usar `modify-vpc-peering-connection-options`.

## AWS CLI

Para habilitar la comunicación a través de una conexión VPC entre pares desde su conexión local ClassicLink

En este ejemplo, para la conexión entre pares `pcx-aaaabbbb`, el propietario del solicitante VPC modifica las opciones de conexión entre VPC pares para permitir que una ClassicLink conexión local se comuniquen con el par. VPC

Comando:

```
aws ec2 modify-vpc-peering-connection-options --vpc-peering-connection-id pcx-aaaabbbb --requester-peering-connection-options AllowEgressFromLocalClassicLinkToRemoteVpc=true
```

Salida:

```
{
  "RequesterPeeringConnectionOptions": {
    "AllowEgressFromLocalClassicLinkToRemoteVpc": true
  }
}
```

Para habilitar la comunicación a través de una conexión de VPC pares desde la conexión local a una remota VPC ClassicLink

En este ejemplo, el propietario del aceptador VPC modifica las opciones de conexión entre pares VPC para permitir que la conexión local se VPC comuniquen con la ClassicLink conexión entre pares. VPC

Comando:

```
aws ec2 modify-vpc-peering-connection-options --vpc-peering-connection-id pcx-aaaabbbb --accepter-peering-connection-options AllowEgressFromLocalVpcToRemoteClassicLink=true
```

Salida:

```
{
  "AcceptorPeeringConnectionOptions": {
    "AllowEgressFromLocalVpcToRemoteClassicLink": true
  }
}
```

```
}
```

Para habilitar el soporte de DNS resolución para la conexión de emparejamiento VPC

En este ejemplo, el propietario del solicitante VPC modifica las opciones de conexión entre VPC pares `pcx-aaaabbbb` para permitir que la conexión local VPC resuelva los DNS nombres de host públicos en direcciones IP privadas cuando se consultan desde instancias del par. VPC

Comando:

```
aws ec2 modify-vpc-peering-connection-options --vpc-peering-connection-id pcx-aaaabbbb --requester-peering-connection-options AllowDnsResolutionFromRemoteVpc=true
```

Salida:

```
{
  "RequesterPeeringConnectionOptions": {
    "AllowDnsResolutionFromRemoteVpc": true
  }
}
```

- Para obtener API más información, consulte la Referencia de comandos.  
[ModifyVpcPeeringConnectionOptions](#) AWS CLI

## modify-vpc-tenancy

En el siguiente ejemplo de código, se muestra cómo usar `modify-vpc-tenancy`.

AWS CLI

Para modificar el arrendamiento de un VPC

En este ejemplo se modifica la tenencia de to. VPC `vpc-1a2b3c4d` `default`

Comando:

```
aws ec2 modify-vpc-tenancy --vpc-id vpc-1a2b3c4d --instance-tenancy default
```

Salida:

```
{
  "Return": true
}
```

```
}
```

- Para API obtener más información, consulte la Referencia [ModifyVpcTenancy](#) de AWS CLI comandos.

## modify-vpn-connection-options

En el siguiente ejemplo de código, se muestra cómo usar `modify-vpn-connection-options`.

### AWS CLI

Para modificar las opciones de VPN conexión

En el siguiente `modify-vpn-connection-options` ejemplo, se modifica la conexión local IPv4 CIDR en la pasarela del cliente de la VPN conexión especificada.

```
aws ec2 modify-vpn-connection-options \  
  --vpn-connection-id vpn-1122334455aabbccd \  
  --local-ipv4-network-cidr 10.0.0.0/16
```

Salida:

```
{  
  "VpnConnections": [  
    {  
      "CustomerGatewayConfiguration": "...configuration information...",  
      "CustomerGatewayId": "cgw-01234567abcde1234",  
      "Category": "VPN",  
      "State": "modifying",  
      "Type": "ipsec.1",  
      "VpnConnectionId": "vpn-1122334455aabbccd",  
      "TransitGatewayId": "tgw-00112233445566aab",  
      "Options": {  
        "EnableAcceleration": false,  
        "StaticRoutesOnly": true,  
        "LocalIpv4NetworkCidr": "10.0.0.0/16",  
        "RemoteIpv4NetworkCidr": "0.0.0.0/0",  
        "TunnelInsideIpVersion": "ipv4"  
      },  
      "Routes": [],  
      "Tags": [  
        {
```



```

        "Key": "Name",
        "Value": "CanadaVPN"
    }
],
"VgwTelemetry": [
    {
        "AcceptedRouteCount": 0,
        "LastStatusChange": "2020-07-29T10:35:11.000Z",
        "OutsideIpAddress": "203.0.113.3",
        "Status": "DOWN",
        "StatusMessage": ""
    },
    {
        "AcceptedRouteCount": 0,
        "LastStatusChange": "2020-09-02T09:09:33.000Z",
        "OutsideIpAddress": "203.0.113.5",
        "Status": "UP",
        "StatusMessage": ""
    }
]
}
]
}

```

Para obtener más información, consulte [Modificación de las opciones de VPN conexión de sitio a sitio](#) en la Guía del usuario de sitio a AWS sitio VPN.

- Para obtener API más información, consulte la Referencia de comandos [ModifyVpnConnectionOptions](#).AWS CLI

## modify-vpn-connection

En el siguiente ejemplo de código, se muestra cómo usar `modify-vpn-connection`.

### AWS CLI

Para modificar una VPN conexión

El siguiente `modify-vpn-connection` ejemplo cambia la puerta de enlace de destino para VPN conectarla `vpn-12345678901234567` a una puerta de enlace privada `virtualvgw-11223344556677889`:

```
aws ec2 modify-vpn-connection \
```

```
--vpn-connection-id vpn-12345678901234567 \  
--vpn-gateway-id vgw-11223344556677889
```

Salida:

```
{  
  "VpnConnection": {  
    "CustomerGatewayConfiguration": "...configuration information...",  
    "CustomerGatewayId": "cgw-aabbccdee1122334",  
    "Category": "VPN",  
    "State": "modifying",  
    "Type": "ipsec.1",  
    "VpnConnectionId": "vpn-12345678901234567",  
    "VpnGatewayId": "vgw-11223344556677889",  
    "Options": {  
      "StaticRoutesOnly": false  
    },  
    "VgwTelemetry": [  
      {  
        "AcceptedRouteCount": 0,  
        "LastStatusChange": "2019-07-17T07:34:00.000Z",  
        "OutsideIpAddress": "18.210.3.222",  
        "Status": "DOWN",  
        "StatusMessage": "IPSEC IS DOWN"  
      },  
      {  
        "AcceptedRouteCount": 0,  
        "LastStatusChange": "2019-07-20T21:20:16.000Z",  
        "OutsideIpAddress": "34.193.129.33",  
        "Status": "DOWN",  
        "StatusMessage": "IPSEC IS DOWN"  
      }  
    ]  
  }  
}
```

- Para API obtener más información, consulte [ModifyVpnConnection](#) la Referencia de AWS CLI comandos.

## modify-vpn-tunnel-certificate

En el siguiente ejemplo de código, se muestra cómo usar `modify-vpn-tunnel-certificate`.

## AWS CLI

Para rotar un certificado de VPN túnel

En el siguiente `modify-vpn-tunnel-certificate` ejemplo, se rota el certificado del túnel especificado para una conexión VPN

```
aws ec2 modify-vpn-tunnel-certificate \  
  --vpn-tunnel-outside-ip-address 203.0.113.17 \  
  --vpn-connection-id vpn-12345678901234567
```

Salida:

```
{  
  "VpnConnection": {  
    "CustomerGatewayConfiguration": "...configuration information...",  
    "CustomerGatewayId": "cgw-aabbccdde1122334",  
    "Category": "VPN",  
    "State": "modifying",  
    "Type": "ipsec.1",  
    "VpnConnectionId": "vpn-12345678901234567",  
    "VpnGatewayId": "vgw-11223344556677889",  
    "Options": {  
      "StaticRoutesOnly": false  
    },  
    "VgwTelemetry": [  
      {  
        "AcceptedRouteCount": 0,  
        "LastStatusChange": "2019-09-11T17:27:14.000Z",  
        "OutsideIpAddress": "203.0.113.17",  
        "Status": "DOWN",  
        "StatusMessage": "IPSEC IS DOWN",  
        "CertificateArn": "arn:aws:acm:us-east-1:123456789101:certificate/  
c544d8ce-20b8-4fff-98b0-example"  
      },  
      {  
        "AcceptedRouteCount": 0,  
        "LastStatusChange": "2019-09-11T17:26:47.000Z",  
        "OutsideIpAddress": "203.0.114.18",  
        "Status": "DOWN",  
        "StatusMessage": "IPSEC IS DOWN",  
        "CertificateArn": "arn:aws:acm:us-  
east-1:123456789101:certificate/5ab64566-761b-4ad3-b259-example"      }  
    ]  
  }  
}
```

```

    }
  ]
}
}

```

- Para API obtener más información, consulte [ModifyVpnTunnelCertificate](#) la Referencia de AWS CLI comandos.

## modify-`vpn-tunnel-options`

En el siguiente ejemplo de código, se muestra cómo usar `modify-vpn-tunnel-options`.

### AWS CLI

Para modificar las opciones de túnel de una VPN conexión

En el siguiente `modify-vpn-tunnel-options` ejemplo, se actualizan los grupos Diffie-Hellman permitidos para el túnel y la conexión especificados. VPN

```

aws ec2 modify-vpn-tunnel-options \
  --vpn-connection-id vpn-12345678901234567 \
  --vpn-tunnel-outside-ip-address 203.0.113.17 \
  --tunnel-options Phase1DHGroupNumbers=[{Value=14},{Value=15},{Value=16},
{Value=17},{Value=18}],Phase2DHGroupNumbers=[{Value=14},{Value=15},{Value=16},
{Value=17},{Value=18}]

```

Salida:

```

{
  "VpnConnection": {
    "CustomerGatewayConfiguration": "...configuration information...",
    "CustomerGatewayId": "cgw-aabbccdde1122334",
    "Category": "VPN",
    "State": "available",
    "Type": "ipsec.1",
    "VpnConnectionId": "vpn-12345678901234567",
    "VpnGatewayId": "vgw-11223344556677889",
    "Options": {
      "StaticRoutesOnly": false,
      "TunnelOptions": [
        {
          "OutsideIpAddress": "203.0.113.17",

```

```
        "Phase1DHGroupNumbers": [
            {
                "Value": 14
            },
            {
                "Value": 15
            },
            {
                "Value": 16
            },
            {
                "Value": 17
            },
            {
                "Value": 18
            }
        ],
        "Phase2DHGroupNumbers": [
            {
                "Value": 14
            },
            {
                "Value": 15
            },
            {
                "Value": 16
            },
            {
                "Value": 17
            },
            {
                "Value": 18
            }
        ]
    },
    {
        "OutsideIpAddress": "203.0.114.19"
    }
]
},
"VgwTelemetry": [
    {
        "AcceptedRouteCount": 0,
        "LastStatusChange": "2019-09-10T21:56:54.000Z",
```

```

        "OutsideIpAddress": "203.0.113.17",
        "Status": "DOWN",
        "StatusMessage": "IPSEC IS DOWN"
    },
    {
        "AcceptedRouteCount": 0,
        "LastStatusChange": "2019-09-10T21:56:43.000Z",
        "OutsideIpAddress": "203.0.114.19",
        "Status": "DOWN",
        "StatusMessage": "IPSEC IS DOWN"
    }
]
}
}

```

- Para obtener API más información, consulte la Referencia de comandos [ModifyVpnTunnelOptions](#).AWS CLI

## monitor-instances

En el siguiente ejemplo de código, se muestra cómo usar `monitor-instances`.

AWS CLI

Habilitar el monitoreo detallado para una instancia

Este comando de ejemplo habilita el monitoreo detallado de la instancia especificada.

Comando:

```
aws ec2 monitor-instances --instance-ids i-1234567890abcdef0
```

Salida:

```

{
  "InstanceMonitorings": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "Monitoring": {
        "State": "pending"
      }
    }
  ]
}

```

```
]
}
```

- Para API obtener más información, consulte [MonitorInstances](#) la Referencia de AWS CLI comandos.

## move-address-to-vpc

En el siguiente ejemplo de código, se muestra cómo usar `move-address-to-vpc`.

### AWS CLI

Para mover una dirección a EC2 - VPC

En este ejemplo, se mueve la dirección IP elástica 54.123.4.56 a la EC2 plataforma -. VPC

Comando:

```
aws ec2 move-address-to-vpc --public-ip 54.123.4.56
```

Salida:

```
{
  "Status": "MoveInProgress"
}
```

- Para API obtener más información, consulte la Referencia de [MoveAddressToVpc](#) comandos AWS CLI .

## move-byoip-cidr-to-ipam

En el siguiente ejemplo de código, se muestra cómo usar `move-byoip-cidr-to-ipam`.

### AWS CLI

Para transferir un BYOIP CIDR a IPAM

El siguiente `move-byoip-cidr-to-ipam` ejemplo transfiere un BYOIP CIDR a IPAM.

(Linux):

```
aws ec2 move-byoip-cidr-to-ipam \
  --region us-west-2 \
  --ipam-pool-id ipam-pool-0a03d430ca3f5c035 \
  --ipam-pool-owner 111111111111 \
  --cidr 130.137.249.0/24
```

(Windows):

```
aws ec2 move-byoip-cidr-to-ipam ^
  --region us-west-2 ^
  --ipam-pool-id ipam-pool-0a03d430ca3f5c035 ^
  --ipam-pool-owner 111111111111 ^
  --cidr 130.137.249.0/24
```

Salida:

```
{
  "ByoipCidr": {
    "Cidr": "130.137.249.0/24",
    "State": "pending-transfer"
  }
}
```

Para obtener más información, consulta el [Tutorial: Transferir un objeto existente BYOIP IPv4 CIDR a IPAM](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [MoveByoipCidrToIpam](#) la Referencia de AWS CLI comandos.

## network-insights-access-scope

En el siguiente ejemplo de código, se muestra cómo usar `network-insights-access-scope`.

AWS CLI

Para crear ámbitos de acceso a Network Insights

En el siguiente `create-network-insights-access-scope` ejemplo, se crea un ámbito de acceso a Network Insights en su AWS cuenta.

```
aws ec2 create-network-insights-access-scope \
```



```
--cli-input-json file://access-scope-file.json
```

Contenidos de `access-scope-file.json`:

```
{
  {
    "MatchPaths": [
      {
        "Source": {
          "ResourceStatement": {
            "Resources": [
              "vpc-abcd12e3"
            ]
          }
        }
      }
    ],
    "ExcludePaths": [
      {
        "Source": {
          "ResourceStatement": {
            "ResourceTypes": [
              "AWS::EC2::InternetGateway"
            ]
          }
        }
      }
    ]
  }
}
```

Salida:

```
{
  "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789111"
}{
  "NetworkInsightsAccessScope": {
    "NetworkInsightsAccessScopeId": "nis-123456789222",
    "NetworkInsightsAccessScopeArn": "arn:aws:ec2:us-east-1:123456789222:network-insights-access-scope/nis-123456789222",
    "CreateDate": "2022-01-25T19:20:28.796000+00:00",
    "UpdateDate": "2022-01-25T19:20:28.797000+00:00"
  },
}
```

```

"NetworkInsightsAccessScopeContent": {
  "NetworkInsightsAccessScopeId": "nis-04c0c0fbca737c404",
  "MatchPaths": [
    {
      "Source": {
        "ResourceStatement": {
          "Resources": [
            "vpc-abcd12e3"
          ]
        }
      }
    }
  ],
  "ExcludePaths": [
    {
      "Source": {
        "ResourceStatement": {
          "ResourceTypes": [
            "AWS::EC2::InternetGateway"
          ]
        }
      }
    }
  ]
}
}

```

Para obtener más información, consulte [Cómo empezar a utilizar Network Access Analyzer mediante AWS CLI](#) la guía Network Access Analyzer.

- Para API obtener más información, consulte la Referencia [NetworkInsightsAccessScope](#) de AWS CLI comandos.

## provision-byoip-cidr

En el siguiente ejemplo de código, se muestra cómo usar `provision-byoip-cidr`.

### AWS CLI

Para aprovisionar un rango de direcciones

El siguiente `provision-byoip-cidr` ejemplo proporciona un rango de direcciones IP públicas para su uso con AWS.

```
aws ec2 provision-byoip-cidr \  
  --cidr 203.0.113.25/24 \  
  --cidr-authorization-context Message="$text_message",Signature="$signed_message"
```

Salida:

```
{  
  "ByoipCidr": {  
    "Cidr": "203.0.113.25/24",  
    "State": "pending-provision"  
  }  
}
```

Para obtener más información sobre la creación de las cadenas de mensajes para el contexto de autorización, consulte [Bring Your Own IP Addresses](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [ProvisionByoipCidr](#) la Referencia de AWS CLI comandos.

## provision-ipam-pool-cidr

En el siguiente ejemplo de código, se muestra cómo usar `provision-ipam-pool-cidr`.

AWS CLI

Para CIDR aprovisionar un IPAM grupo

El siguiente `provision-ipam-pool-cidr` ejemplo aprovisiona CIDR a un IPAM grupo.

(Linux):

```
aws ec2 provision-ipam-pool-cidr \  
  --ipam-pool-id ipam-pool-0533048da7d823723 \  
  --cidr 10.0.0.0/24
```

(Windows):

```
aws ec2 provision-ipam-pool-cidr ^  
  --ipam-pool-id ipam-pool-0533048da7d823723 ^  
  --cidr 10.0.0.0/24
```

Salida:

```
{
  "IpamPoolCidr": {
    "Cidr": "10.0.0.0/24",
    "State": "pending-provision"
  }
}
```

Para obtener más información, consulte [Aprovisionar CIDRs a un grupo](#) en la Guía del VPC IPAM usuario de Amazon.

- Para API obtener más información, consulte [ProvisionIpamPoolCidr](#) la Referencia de AWS CLI comandos.

## purchase-host-reservation

En el siguiente ejemplo de código, se muestra cómo usar `purchase-host-reservation`.

AWS CLI

Para comprar una reserva de host dedicada

En este ejemplo, se compra la oferta de reserva de anfitrión dedicado especificada para el host dedicado especificado de su cuenta.

Comando:

```
aws ec2 purchase-host-reservation --offering-id hro-03f707bf363b6b324 --host-id-set h-013abcd2a00cbd123
```

Salida:

```
{
  "TotalHourlyPrice": "1.499",
  "Purchase": [
    {
      "HourlyPrice": "1.499",
      "InstanceFamily": "m4",
      "PaymentOption": "NoUpfront",
      "HostIdSet": [
        "h-013abcd2a00cbd123"
      ]
    }
  ]
}
```

```
    ],
    "HostReservationId": "hr-0d418a3a4ffc669ae",
    "UpfrontPrice": "0.000",
    "Duration": 31536000
  }
],
"TotalUpfrontPrice": "0.000"
}
```

- Para API obtener más información, consulte [PurchaseHostReservation](#) la Referencia de AWS CLI comandos.

## **purchase-reserved-instances-offering**

En el siguiente ejemplo de código, se muestra cómo usar `purchase-reserved-instances-offering`.

### AWS CLI

Para comprar una oferta de instancias reservadas

Este comando de ejemplo ilustra la compra de una oferta de instancias reservadas, especificando un identificador de oferta y un número de instancias.

Comando:

```
aws ec2 purchase-reserved-instances-offering --reserved-instances-offering-id ec06327e-dd07-46ee-9398-75b5fexample --instance-count 3
```

Salida:

```
{
  "ReservedInstancesId": "af9f760e-6f91-4559-85f7-4980eexample"
}
```

- Para API obtener más información, consulte [PurchaseReservedInstancesOffering](#) la Referencia de AWS CLI comandos.

## **purchase-scheduled-instances**

En el siguiente ejemplo de código, se muestra cómo usar `purchase-scheduled-instances`.

## AWS CLI

Para comprar una instancia programada

En este ejemplo, se compra una instancia programada.

Comando:

```
aws ec2 purchase-scheduled-instances --purchase-requests file://purchase-request.json
```

Purchase-request.json:

```
[
  {
    "PurchaseToken": "eyJ2IjoiMSIsInMiOjEsImMiOi...",
    "InstanceCount": 1
  }
]
```

Salida:

```
{
  "ScheduledInstanceSet": [
    {
      "AvailabilityZone": "us-west-2b",
      "ScheduledInstanceId": "sci-1234-1234-1234-1234-123456789012",
      "HourlyPrice": "0.095",
      "CreateDate": "2016-01-25T21:43:38.612Z",
      "Recurrence": {
        "OccurrenceDaySet": [
          1
        ],
        "Interval": 1,
        "Frequency": "Weekly",
        "OccurrenceRelativeToEnd": false,
        "OccurrenceUnit": ""
      },
      "Platform": "Linux/UNIX",
      "TermEndDate": "2017-01-31T09:00:00Z",
      "InstanceCount": 1,
      "SlotDurationInHours": 32,
    }
  ]
}
```

```
    "TermStartDate": "2016-01-31T09:00:00Z",
    "NetworkPlatform": "EC2-VPC",
    "TotalScheduledInstanceHours": 1696,
    "NextSlotStartTime": "2016-01-31T09:00:00Z",
    "InstanceType": "c4.large"
  }
]
```

- Para obtener API más información, consulte la Referencia de comandos.

[PurchaseScheduledInstances](#) AWS CLI

## reboot-instances

En el siguiente ejemplo de código, se muestra cómo usar `reboot-instances`.

### AWS CLI

Para reiniciar una EC2 instancia de Amazon

En este ejemplo, se reinicia la instancia especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 reboot-instances --instance-ids i-1234567890abcdef5
```

Para obtener más información, consulte Reiniciar su instancia en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte [RebootInstances](#) la Referencia de AWS CLI comandos.

## register-image

En el siguiente ejemplo de código, se muestra cómo usar `register-image`.

### AWS CLI

Ejemplo 1: Para registrar y AMI utilizar un archivo de manifiesto

El siguiente `register-image` ejemplo registra y AMI utiliza el archivo de manifiesto especificado en Amazon S3.

```
aws ec2 register-image \  
  --name my-image \  
  --image-location my-s3-bucket/myimage/image.manifest.xml
```

Salida:

```
{  
  "ImageId": "ami-1234567890EXAMPLE"  
}
```

Para obtener más información, consulte [Amazon Machine Images \(AMI\)](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para registrar y AMI utilizar una instantánea de un dispositivo raíz

El siguiente `register-image` ejemplo registra y AMI utiliza la instantánea especificada de un volumen EBS raíz como dispositivo `/dev/xvda`. El mapeo del dispositivo de bloques también incluye un EBS volumen vacío de 100 GiB como dispositivo. `/dev/xvdf`

```
aws ec2 register-image \  
  --name my-image \  
  --root-device-name /dev/xvda \  
  --block-device-mappings DeviceName=/dev/  
xvda,Ebs={SnapshotId=snap-0db2cf683925d191f} DeviceName=/dev/  
xvdf,Ebs={VolumeSize=100}
```

Salida:

```
{  
  "ImageId": "ami-1a2b3c4d5eEXAMPLE"  
}
```

Para obtener más información, consulte [Amazon Machine Images \(AMI\)](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [RegisterImage](#) la Referencia de AWS CLI comandos.



## register-instance-event-notification-attributes

En el siguiente ejemplo de código, se muestra cómo usar `register-instance-event-notification-attributes`.

### AWS CLI

Ejemplo 1: Para incluir todas las etiquetas en las notificaciones de eventos

El siguiente `register-instance-event-notification-attributes` ejemplo incluye todas las etiquetas de las notificaciones de eventos.

```
aws ec2 register-instance-event-notification-attributes \  
  --instance-tag-attribute IncludeAllTagsOfInstance=true
```

Salida:

```
{  
  "InstanceTagAttribute": {  
    "InstanceTagKeys": [],  
    "IncludeAllTagsOfInstance": true  
  }  
}
```

Para obtener más información, consulte [Eventos programados para sus instancias](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

Ejemplo 2: Para incluir etiquetas específicas en las notificaciones de eventos

El siguiente `register-instance-event-notification-attributes` ejemplo incluye las etiquetas especificadas en las notificaciones de eventos. Si es así, no puede especificar `IncludeAllTagsOfInstance`

```
aws ec2 register-instance-event-notification-attributes \  
  --instance-tag-attribute InstanceTagKeys="tag-key1","tag-key2"
```

Salida:

```
{  
  "InstanceTagAttribute": {  
    "InstanceTagKeys": [  

```

```

        "tag-key1",
        "tag-key2"
    ],
    "IncludeAllTagsOfInstance": false
}
}

```

Para obtener más información, consulte [Eventos programados para sus instancias](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

- Para API obtener más información, consulte [RegisterInstanceEventNotificationAttributes](#) la Referencia de AWS CLI comandos.

## register-transit-gateway-multicast-group-sources

En el siguiente ejemplo de código, se muestra cómo usar `register-transit-gateway-multicast-group-sources`.

### AWS CLI

Para registrar una fuente en un grupo de multidifusión de Transit Gateway.

El siguiente `register-transit-gateway-multicast-group-sources` ejemplo registra el origen del grupo de interfaz de red especificado con un grupo de multidifusión.

```

aws ec2 register-transit-gateway-multicast-group-sources \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
  --group-ip-address 224.0.1.0 \
  --network-interface-ids eni-07f290fc3c090cbae

```

Salida:

```

{
  "RegisteredMulticastGroupSources": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
    "RegisteredNetworkInterfaceIds": [
      "eni-07f290fc3c090cbae"
    ],
    "GroupIpAddress": "224.0.1.0"
  }
}

```

Para obtener más información, consulte [Registrar fuentes con un grupo de multidifusión](#) en la Guía del usuario de AWS Transit Gateways.

- Para API obtener más información, consulte la Referencia [RegisterTransitGatewayMulticastGroupSources](#) de AWS CLI comandos.

## **register-transit-gateway-multicast-group-members**

En el siguiente ejemplo de código, se muestra cómo usar `register-transit-gateway-multicast-group-members`.

### AWS CLI

Para ver la información sobre las asociaciones de dominios de multidifusión de Transit Gateway

El siguiente `register-transit-gateway-multicast-group-members` ejemplo devuelve las asociaciones del dominio de multidifusión especificado.

```
aws ec2 register-transit-gateway-multicast-group-members \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
  --group-ip-address 224.0.1.0 \
  --network-interface-ids eni-0e246d32695012e81
```

Salida:

```
{
  "RegisteredMulticastGroupMembers": {
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
    "RegisteredNetworkInterfaceIds": [
      "eni-0e246d32695012e81"
    ],
    "GroupIpAddress": "224.0.1.0"
  }
}
```

Para obtener más información, consulte [Administrar dominios de multidifusión](#) en la Guía del usuario de Transit Gateways.

- Para API obtener más información, consulte la Referencia [RegisterTransitGatewayMulticastGroupMembers](#) de AWS CLI comandos.

## register-transit-gateway-multicast-group-sources

En el siguiente ejemplo de código, se muestra cómo usar `register-transit-gateway-multicast-group-sources`.

### AWS CLI

Para registrar una fuente en un grupo de multidifusión de Transit Gateway.

El siguiente `register-transit-gateway-multicast-group-sources` ejemplo registra el origen del grupo de interfaz de red especificado con un grupo de multidifusión.

```
aws ec2 register-transit-gateway-multicast-group-sources \  
  --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \  
  --group-ip-address 224.0.1.0 \  
  --network-interface-ids eni-07f290fc3c090cbae
```

Salida:

```
{  
  "RegisteredMulticastGroupSources": {  
    "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",  
    "RegisteredNetworkInterfaceIds": [  
      "eni-07f290fc3c090cbae"  
    ],  
    "GroupIpAddress": "224.0.1.0"  
  }  
}
```

Para obtener más información, consulte [Administración de dominios de multidifusión](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [RegisterTransitGatewayMulticastGroupSources](#) de AWS CLI comandos.

## reject-transit-gateway-peering-attachment

En el siguiente ejemplo de código, se muestra cómo usar `reject-transit-gateway-peering-attachment`.

## AWS CLI

Para rechazar un adjunto de peering de Transit Gateway

El siguiente `reject-transit-gateway-peering-attachment` ejemplo rechaza la solicitud de adjunto de emparejamiento de la pasarela de tránsito especificada. El `--region` parámetro especifica la región en la que se encuentra la puerta de enlace de tránsito receptora.

```
aws ec2 reject-transit-gateway-peering-attachment \  
  --transit-gateway-attachment-id tgw-attach-4455667788aabbccd \  
  --region us-east-2
```

Salida:

```
{  
  "TransitGatewayPeeringAttachment": {  
    "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",  
    "RequesterTgwInfo": {  
      "TransitGatewayId": "tgw-123abc05e04123abc",  
      "OwnerId": "123456789012",  
      "Region": "us-west-2"  
    },  
    "AcceptorTgwInfo": {  
      "TransitGatewayId": "tgw-11223344aabbcc112",  
      "OwnerId": "123456789012",  
      "Region": "us-east-2"  
    },  
    "State": "rejecting",  
    "CreationTime": "2019-12-09T11:50:31.000Z"  
  }  
}
```

Para obtener más información, consulte los [adjuntos de peering de Transit Gateway](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [RejectTransitGatewayPeeringAttachment](#) de AWS CLI comandos.

## `reject-transit-gateway-vpc-attachment`

En el siguiente ejemplo de código, se muestra cómo usar `reject-transit-gateway-vpc-attachment`.

## AWS CLI

Para rechazar un VPC adjunto a una pasarela de tránsito

El siguiente `reject-transit-gateway-vpc-attachment` ejemplo rechaza el VPC adjunto de la pasarela de tránsito especificado.

```
aws ec2 reject-transit-gateway-vpc-attachment \  
  --transit-gateway-attachment-id tgw-attach-0a34fe6b4fEXAMPLE
```

Salida:

```
{  
  "TransitGatewayVpcAttachment": {  
    "TransitGatewayAttachmentId": "tgw-attach-0a34fe6b4fEXAMPLE",  
    "TransitGatewayId": "tgw-0262a0e521EXAMPLE",  
    "VpcId": "vpc-07e8ffd50fEXAMPLE",  
    "VpcOwnerId": "111122223333",  
    "State": "pending",  
    "SubnetIds": [  
      "subnet-0752213d59EXAMPLE"  
    ],  
    "CreationTime": "2019-07-10T17:33:46.000Z",  
    "Options": {  
      "DnsSupport": "enable",  
      "Ipv6Support": "disable"  
    }  
  }  
}
```

Para obtener más información, consulte los [adjuntos de Transit Gateways a a VPC](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [RejectTransitGatewayVpcAttachment](#) la Referencia de AWS CLI comandos.

## `reject-transit-gateway-vpc-attachments`

En el siguiente ejemplo de código, se muestra cómo usar `reject-transit-gateway-vpc-attachments`.

## AWS CLI

Para rechazar un VPC adjunto a una pasarela de tránsito

El siguiente `reject-transit-gateway-vpc-attachment` ejemplo rechaza el VPC adjunto de la pasarela de tránsito especificado.

```
aws ec2 reject-transit-gateway-vpc-attachment \  
  --transit-gateway-attachment-id tgw-attach-0a34fe6b4fEXAMPLE
```

Salida:

```
{  
  "TransitGatewayVpcAttachment": {  
    "TransitGatewayAttachmentId": "tgw-attach-0a34fe6b4fEXAMPLE",  
    "TransitGatewayId": "tgw-0262a0e521EXAMPLE",  
    "VpcId": "vpc-07e8ffd50fEXAMPLE",  
    "VpcOwnerId": "111122223333",  
    "State": "pending",  
    "SubnetIds": [  
      "subnet-0752213d59EXAMPLE"  
    ],  
    "CreationTime": "2019-07-10T17:33:46.000Z",  
    "Options": {  
      "DnsSupport": "enable",  
      "Ipv6Support": "disable"  
    }  
  }  
}
```

Para obtener más información, consulte los [adjuntos de Transit Gateways a a VPC](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [RejectTransitGatewayVpcAttachments](#) la Referencia de AWS CLI comandos.

## `reject-vpc-endpoint-connections`

En el siguiente ejemplo de código, se muestra cómo usar `reject-vpc-endpoint-connections`.

## AWS CLI

Para rechazar una solicitud de conexión de punto final de interfaz

En este ejemplo, se rechaza la solicitud de conexión de punto final especificada para el servicio de punto final especificado.

Comando:

```
aws ec2 reject-vpc-endpoint-connections --service-id vpce-svc-03d5ebb7d9579a2b3 --vpc-endpoint-ids vpce-0c1308d7312217abc
```

Salida:

```
{
  "Unsuccessful": []
}
```

- Para API obtener más información, consulte [RejectVpcEndpointConnections](#) la Referencia de AWS CLI comandos.

## **reject-vpc-peering-connection**

En el siguiente ejemplo de código, se muestra cómo usar `reject-vpc-peering-connection`.

## AWS CLI

Para rechazar una conexión VPC entre pares

En este ejemplo, se rechaza la solicitud de conexión entre VPC pares especificada.

Comando:

```
aws ec2 reject-vpc-peering-connection --vpc-peering-connection-id pcx-1a2b3c4d
```

Salida:

```
{
  "Return": true
}
```



```
}
```

- Para API obtener más información, consulte [RejectVpcPeeringConnection](#) la Referencia de AWS CLI comandos.

## release-address

En el siguiente ejemplo de código, se muestra cómo usar `release-address`.

### AWS CLI

Para publicar direcciones IP elásticas para EC2 -Classic

En este ejemplo, se publica una dirección IP elástica para usarla con instancias de EC2 -Classic. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 release-address --public-ip 198.51.100.0
```

Para liberar una dirección IP elástica para EC2: VPC

En este ejemplo, se publica una dirección IP elástica para usarla con instancias de un VPC. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 release-address --allocation-id eipalloc-64d5890a
```

- Para API obtener más información, consulte [ReleaseAddress](#) la Referencia de AWS CLI comandos.

## release-hosts

En el siguiente ejemplo de código, se muestra cómo usar `release-hosts`.

### AWS CLI

Para liberar un host dedicado de su cuenta

Para liberar un host dedicado de tu cuenta. Las instancias que se encuentran en el host deben detenerse o cancelarse antes de que se pueda liberar el host.

Comando:

```
aws ec2 release-hosts --host-id=h-0029d6e3cacf1b3da
```

Salida:

```
{
  "Successful": [
    "h-0029d6e3cacf1b3da"
  ],
  "Unsuccessful": []
}
```

- Para API obtener más información, consulte [ReleaseHosts](#) la Referencia de AWS CLI comandos.

## release-ipam-pool-allocation

En el siguiente ejemplo de código, se muestra cómo usar `release-ipam-pool-allocation`.

### AWS CLI

Para publicar una asignación IPAM de grupos

En este ejemplo, usted es un administrador IPAM delegado que intentó eliminar un IPAM grupo, pero recibió un error que indica que no puede eliminar el grupo mientras el grupo tenga asignaciones. Está utilizando este comando para liberar una asignación de grupo.

Tenga en cuenta lo siguiente:

Solo puede usar este comando para asignaciones personalizadas. Para eliminar una asignación de un recurso sin eliminar el recurso, defina su estado supervisado en falso mediante [modify-ipam-resource-cidr](#). Para completar esta solicitud, necesitará el ID del IPAM grupo, que puede obtener. [describe-ipam-pools](#) También necesitará el identificador de asignación, que puede obtener con [get-ipam-pool-allocations](#) él. Si no desea eliminar las asignaciones una por una, puede utilizar el `--cascade` option al eliminar un IPAM grupo para liberar automáticamente

cualquier asignación del grupo antes de eliminarla. Hay varios requisitos previos antes de ejecutar este comando. Para obtener más información, consulte [Publicar una asignación](#) en la Guía del VPC IPAM usuario de Amazon. `--region` En la que ejecute este comando debe ser la configuración regional del IPAM grupo en el que se encuentra la asignación.

En el siguiente `release-ipam-pool-allocation` ejemplo, se publica una asignación IPAM de grupos.

```
aws ec2 release-ipam-pool-allocation \  
  --ipam-pool-id ipam-pool-07bdd12d7c94e4693 \  
  --cidr 10.0.0.0/23 \  
  --ipam-pool-allocation-id ipam-pool-alloc-0e66a1f730da54791b99465b79e7d1e89 \  
  --region us-west-1
```

Salida:

```
{  
  "Success": true  
}
```

Una vez que publiques una asignación, es posible que desees ejecutarla [delete-ipam-pool](#).

- Para API obtener más información, consulte [ReleaseIpamPoolAllocation](#) la Referencia de AWS CLI comandos.

## **replace-iam-instance-profile-association**

En el siguiente ejemplo de código, se muestra cómo usar `replace-iam-instance-profile-association`.

AWS CLI

Para reemplazar un perfil de IAM instancia por una instancia

En este ejemplo, se reemplaza el perfil de IAM instancia representado por la `iip-assoc-060bae234aac2e7fa` asociación por el perfil de IAM instancia denominado `AdminRole`.

```
aws ec2 replace-iam-instance-profile-association \  
  --iam-instance-profile Name=AdminRole \  
  --association-id iip-assoc-060bae234aac2e7fa
```

Salida:

```
{
  "IamInstanceProfileAssociation": {
    "InstanceId": "i-087711ddaf98f9489",
    "State": "associating",
    "AssociationId": "iip-assoc-0b215292fab192820",
    "IamInstanceProfile": {
      "Id": "AIPAJLNLDX3AMYZNWYYAY",
      "Arn": "arn:aws:iam::123456789012:instance-profile/AdminRole"
    }
  }
}
```

- Para API obtener más información, consulte [ReplacelamInstanceProfileAssociation](#) la Referencia de AWS CLI comandos.

## replace-network-acl-association

En el siguiente ejemplo de código, se muestra cómo usar `replace-network-acl-association`.

AWS CLI

Para reemplazar la red ACL asociada a una subred

En este ejemplo, se asocia la red ACL especificada a la subred de la asociación de red ACL especificada.

Comando:

```
aws ec2 replace-network-acl-association --association-id aclassoc-e5b95c8c --
network-acl-id acl-5fb85d36
```

Salida:

```
{
  "NewAssociationId": "aclassoc-3999875b"
}
```

- Para API obtener más información, consulte [ReplaceNetworkAclAssociation](#) la Referencia de AWS CLI comandos.

## replace-network-acl-entry

En el siguiente ejemplo de código, se muestra cómo usar `replace-network-acl-entry`.

### AWS CLI

Para reemplazar una ACL entrada de red

En este ejemplo se reemplaza una entrada de la red especificada ACL. La nueva regla 100 permite la entrada de tráfico desde el 203.0.113.12/24 por el UDP puerto 53 () DNS a cualquier subred asociada.

Comando:

```
aws ec2 replace-network-acl-entry --network-acl-id acl-5fb85d36 --ingress --rule-number 100 --protocol udp --port-range From=53,To=53 --cidr-block 203.0.113.12/24 --rule-action allow
```

- Para API obtener más información, consulte la Referencia de comandos.

[ReplaceNetworkAclEntry](#) AWS CLI

## replace-route-table-association

En el siguiente ejemplo de código, se muestra cómo usar `replace-route-table-association`.

### AWS CLI

Para reemplazar la tabla de rutas asociada a una subred

En este ejemplo, se asocia la tabla de enrutamiento especificada a la subred de la asociación de tabla de enrutamiento especificada.

Comando:

```
aws ec2 replace-route-table-association --association-id rtbassoc-781d0d1a --route-table-id rtb-22574640
```

Salida:

```
{
```

```
"NewAssociationId": "rtbassoc-3a1f0f58"  
}
```

- Para API obtener más información, consulte [ReplaceRouteTableAssociation](#) la Referencia de AWS CLI comandos.

## replace-route

En el siguiente ejemplo de código, se muestra cómo usar `replace-route`.

### AWS CLI

Para reemplazar una ruta

Este ejemplo reemplaza la ruta especificada en la tabla de rutas especificada. La nueva ruta coincide con la especificada CIDR y envía el tráfico a la puerta de enlace privada virtual especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 replace-route --route-table-id rtb-22574640 --destination-cidr-  
block 10.0.0.0/16 --gateway-id vgw-9a4cacf3
```

- Para API obtener más información, consulte [ReplaceRoute](#) la Referencia de AWS CLI comandos.

## replace-transit-gateway-route

En el siguiente ejemplo de código, se muestra cómo usar `replace-transit-gateway-route`.

### AWS CLI

Para reemplazar la ruta especificada en la tabla de rutas de la puerta de enlace de tránsito especificada

El siguiente `replace-transit-gateway-route` ejemplo reemplaza la ruta de la tabla de rutas de la puerta de enlace de tránsito especificada.

```
aws ec2 replace-transit-gateway-route \  

```

```
--destination-cidr-block 10.0.2.0/24 \  
--transit-gateway-attachment-id tgw-attach-09b52ccdb5EXAMPLE \  
--transit-gateway-route-table-id tgw-rtb-0a823edbdeEXAMPLE
```

Salida:

```
{  
  "Route": {  
    "DestinationCidrBlock": "10.0.2.0/24",  
    "TransitGatewayAttachments": [  
      {  
        "ResourceId": "vpc-4EXAMPLE",  
        "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",  
        "ResourceType": "vpc"  
      }  
    ],  
    "Type": "static",  
    "State": "active"  
  }  
}
```

Para obtener más información, consulte [las tablas de rutas de Transit Gateways](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [ReplaceTransitGatewayRoute](#) la Referencia de AWS CLI comandos.

## report-instance-status

En el siguiente ejemplo de código, se muestra cómo usar `report-instance-status`.

### AWS CLI

Para informar sobre el estado de una instancia

Este comando de ejemplo informa sobre el estado de la instancia especificada.

Comando:

```
aws ec2 report-instance-status --instances i-1234567890abcdef0 --status impaired --  
reason-codes unresponsive
```

- Para API obtener más información, consulte [ReportInstanceStatus](#) la Referencia de AWS CLI comandos.

## request-spot-fleet

En el siguiente ejemplo de código, se muestra cómo usar `request-spot-fleet`.

### AWS CLI

Para solicitar una flota puntual en la subred con el precio más bajo

Este comando de ejemplo crea una solicitud de flota puntual con dos especificaciones de lanzamiento que solo difieren según la subred. La flota de Spot lanza las instancias en la subred especificada con el precio más bajo. Si las instancias se lanzan de forma predeterminadaVPC, reciben una dirección IP pública de forma predeterminada. Si las instancias se lanzan de forma no predeterminadaVPC, no reciben una dirección IP pública de forma predeterminada.

Tenga en cuenta que no puede especificar subredes diferentes de la misma zona de disponibilidad en una solicitud de flota puntual.

Comando:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

Config.json:

```
{
  "SpotPrice": "0.04",
  "TargetCapacity": 2,
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role",
  "LaunchSpecifications": [
    {
      "ImageId": "ami-1a2b3c4d",
      "KeyName": "my-key-pair",
      "SecurityGroups": [
        {
          "GroupId": "sg-1a2b3c4d"
        }
      ],
      "InstanceType": "m3.medium",
      "SubnetId": "subnet-1a2b3c4d, subnet-3c4d5e6f",
    }
  ]
}
```



```
        "IamInstanceProfile": {
            "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
        }
    ]
}
```

Salida:

```
{
  "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
}
```

Para solicitar una flota puntual en la zona de disponibilidad con el precio más bajo

Este comando de ejemplo crea una solicitud de flota puntual con dos especificaciones de lanzamiento que solo difieren según la zona de disponibilidad. La flota de Spot lanza las instancias en la zona de disponibilidad especificada con el precio más bajo. Si su cuenta VPC solo admite EC2 -, Amazon EC2 lanza las instancias puntuales en la subred predeterminada de la zona de disponibilidad. Si tu cuenta admite EC2 -Classic, Amazon EC2 lanza las instancias en EC2 -Classic en la zona de disponibilidad.

Comando:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

Config.json:

```
{
  "SpotPrice": "0.04",
  "TargetCapacity": 2,
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role",
  "LaunchSpecifications": [
    {
      "ImageId": "ami-1a2b3c4d",
      "KeyName": "my-key-pair",
      "SecurityGroups": [
        {
          "GroupId": "sg-1a2b3c4d"
        }
      ],
    }
  ],
}
```

```

    "InstanceType": "m3.medium",
    "Placement": {
      "AvailabilityZone": "us-west-2a, us-west-2b"
    },
    "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
    }
  }
]
}

```

Para lanzar instancias puntuales en una subred y asignarles direcciones IP públicas

Este comando de ejemplo asigna direcciones públicas a las instancias lanzadas de forma no predeterminada. VPC Tenga en cuenta que al especificar una interfaz de red, debe incluir el ID de subred y el ID del grupo de seguridad mediante la interfaz de red.

Comando:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

Config.json:

```

{
  "SpotPrice": "0.04",
  "TargetCapacity": 2,
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role",
  "LaunchSpecifications": [
    {
      "ImageId": "ami-1a2b3c4d",
      "KeyName": "my-key-pair",
      "InstanceType": "m3.medium",
      "NetworkInterfaces": [
        {
          "DeviceIndex": 0,
          "SubnetId": "subnet-1a2b3c4d",
          "Groups": [ "sg-1a2b3c4d" ],
          "AssociatePublicIpAddress": true
        }
      ],
      "IamInstanceProfile": {
        "Arn": "arn:aws:iam::880185128111:instance-profile/my-iam-role"
      }
    }
  ]
}

```

```

    }
  }
]
}

```

Para solicitar una flota puntual mediante la estrategia de asignación diversificada

Este comando de ejemplo crea una solicitud de flota puntual que lanza 30 instancias mediante la estrategia de asignación diversificada. Las especificaciones de lanzamiento varían según el tipo de instancia. La flota de Spot distribuye las instancias entre las especificaciones de lanzamiento, de forma que hay 10 instancias de cada tipo.

Comando:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

Config.json:

```

{
  "SpotPrice": "0.70",
  "TargetCapacity": 30,
  "AllocationStrategy": "diversified",
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role",
  "LaunchSpecifications": [
    {
      "ImageId": "ami-1a2b3c4d",
      "InstanceType": "c4.2xlarge",
      "SubnetId": "subnet-1a2b3c4d"
    },
    {
      "ImageId": "ami-1a2b3c4d",
      "InstanceType": "m3.2xlarge",
      "SubnetId": "subnet-1a2b3c4d"
    },
    {
      "ImageId": "ami-1a2b3c4d",
      "InstanceType": "r3.2xlarge",
      "SubnetId": "subnet-1a2b3c4d"
    }
  ]
}

```

Para obtener más información, consulte [Spot Fleet Requests](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte [RequestSpotFleet](#) la Referencia de AWS CLI comandos.

## request-spot-instances

En el siguiente ejemplo de código, se muestra cómo usar `request-spot-instances`.

### AWS CLI

Para solicitar instancias puntuales

Este comando de ejemplo crea una solicitud única de instancia puntual para cinco instancias en la zona de disponibilidad especificada. Si su cuenta VPC solo admite EC2 -, Amazon EC2 lanza las instancias en la subred predeterminada de la zona de disponibilidad especificada. Si su cuenta admite EC2 -Classic, Amazon EC2 lanza las instancias en EC2 -Classic en la zona de disponibilidad especificada.

Comando:

```
aws ec2 request-spot-instances --spot-price "0.03" --instance-count 5 --type "one-time" --launch-specification file://specification.json
```

Especificación.json:

```
{
  "ImageId": "ami-1a2b3c4d",
  "KeyName": "my-key-pair",
  "SecurityGroupIds": [ "sg-1a2b3c4d" ],
  "InstanceType": "m3.medium",
  "Placement": {
    "AvailabilityZone": "us-west-2a"
  },
  "IamInstanceProfile": {
    "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

Salida:

```
{
  "SpotInstanceRequests": [
    {
      "Status": {
        "UpdateTime": "2014-03-25T20:54:21.000Z",
        "Code": "pending-evaluation",
        "Message": "Your Spot request has been submitted for review, and is
pending evaluation."
      },
      "ProductDescription": "Linux/UNIX",
      "SpotInstanceRequestId": "sir-df6f405d",
      "State": "open",
      "LaunchSpecification": {
        "Placement": {
          "AvailabilityZone": "us-west-2a"
        },
        "ImageId": "ami-1a2b3c4d",
        "KeyName": "my-key-pair",
        "SecurityGroups": [
          {
            "GroupName": "my-security-group",
            "GroupId": "sg-1a2b3c4d"
          }
        ],
        "Monitoring": {
          "Enabled": false
        },
        "IamInstanceProfile": {
          "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
        },
        "InstanceType": "m3.medium"
      },
      "Type": "one-time",
      "CreateTime": "2014-03-25T20:54:20.000Z",
      "SpotPrice": "0.050000"
    },
    ...
  ]
}
```

Este comando de ejemplo crea una solicitud única de instancia puntual para cinco instancias de la subred especificada. Amazon EC2 lanza las instancias en la subred especificada. Si no VPC es la predeterminadaVPC, las instancias no reciben una dirección IP pública de forma predeterminada.

## Comando:

```
aws ec2 request-spot-instances --spot-price "0.050" --instance-count 5 --type "one-time" --launch-specification file://specification.json
```

## Especificación.json:

```
{
  "ImageId": "ami-1a2b3c4d",
  "SecurityGroupIds": [ "sg-1a2b3c4d" ],
  "InstanceType": "m3.medium",
  "SubnetId": "subnet-1a2b3c4d",
  "IamInstanceProfile": {
    "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

## Salida:

```
{
  "SpotInstanceRequests": [
    {
      "Status": {
        "UpdateTime": "2014-03-25T22:21:58.000Z",
        "Code": "pending-evaluation",
        "Message": "Your Spot request has been submitted for review, and is pending evaluation."
      },
      "ProductDescription": "Linux/UNIX",
      "SpotInstanceRequestId": "sir-df6f405d",
      "State": "open",
      "LaunchSpecification": {
        "Placement": {
          "AvailabilityZone": "us-west-2a"
        }
        "ImageId": "ami-1a2b3c4d"
        "SecurityGroups": [
          {
            "GroupName": "my-security-group",
            "GroupID": "sg-1a2b3c4d"
          }
        ]
      }
    }
  ]
}
```

```

        "SubnetId": "subnet-1a2b3c4d",
        "Monitoring": {
            "Enabled": false
        },
        "IamInstanceProfile": {
            "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
        },
        "InstanceType": "m3.medium",
    },
    "Type": "one-time",
    "CreateTime": "2014-03-25T22:21:58.000Z",
    "SpotPrice": "0.050000"
},
...
]
}

```

En este ejemplo, se asigna una dirección IP pública a las instancias puntuales que se lanzan de forma no predeterminada. VPC Tenga en cuenta que al especificar una interfaz de red, debe incluir el ID de subred y el ID del grupo de seguridad mediante la interfaz de red.

Comando:

```
aws ec2 request-spot-instances --spot-price "0.050" --instance-count 1 --type "one-time" --launch-specification file://specification.json
```

Especificación.json:

```

{
  "ImageId": "ami-1a2b3c4d",
  "KeyName": "my-key-pair",
  "InstanceType": "m3.medium",
  "NetworkInterfaces": [
    {
      "DeviceIndex": 0,
      "SubnetId": "subnet-1a2b3c4d",
      "Groups": [ "sg-1a2b3c4d" ],
      "AssociatePublicIpAddress": true
    }
  ],
  "IamInstanceProfile": {
    "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}

```

```
}  
}
```

- Para API obtener más información, consulte la Referencia de [RequestSpotInstances](#) comandos AWS CLI .

## reset-address-attribute

En el siguiente ejemplo de código, se muestra cómo usar `reset-address-attribute`.

### AWS CLI

Para restablecer el atributo de nombre de dominio asociado a una dirección IP elástica

Los siguientes `reset-address-attribute` ejemplos restablecen el atributo de nombre de dominio de una dirección IP elástica.

Linux:

```
aws ec2 reset-address-attribute \  
  --allocation-id eipalloc-abcdef01234567890 \  
  --attribute domain-name
```

Windows:

```
aws ec2 reset-address-attribute ^  
  --allocation-id eipalloc-abcdef01234567890 ^  
  --attribute domain-name
```

Salida:

```
{  
  "Addresses": [  
    {  
      "PublicIp": "192.0.2.0",  
      "AllocationId": "eipalloc-abcdef01234567890",  
      "PtrRecord": "example.com."  
      "PtrRecordUpdate": {  
        "Value": "example.net.",  
        "Status": "PENDING"  
      }  
    }  
  ]  
}
```



```
    }  
  ]  
}
```

Para supervisar el cambio pendiente, consulte [describe-addresses-attribute](#) la Referencia de AWS CLI comandos.

- Para API obtener más información, consulte [ResetAddressAttribute](#) la Referencia de AWS CLI comandos.

## **reset-ebs-default-kms-key-id**

En el siguiente ejemplo de código, se muestra cómo usar `reset-ebs-default-kms-key-id`.

### AWS CLI

Para restablecer el EBS cifrado predeterminado CMK

En el siguiente `reset-ebs-default-kms-key-id` ejemplo, se restablece el EBS cifrado predeterminado CMK de tu AWS cuenta en la región actual.

```
aws ec2 reset-ebs-default-kms-key-id
```

Salida:

```
{  
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/8c5b2c63-b9bc-45a3-a87a-5513eEXAMPLE"  
}
```

- Para API obtener más información, consulte [ResetEbsDefaultKmsKeyId](#) la Referencia de AWS CLI comandos.

## **reset-fpga-image-attribute**

En el siguiente ejemplo de código, se muestra cómo usar `reset-fpga-image-attribute`.

### AWS CLI

Para restablecer los atributos de una FPGA imagen de Amazon

En este ejemplo, se restablecen los permisos de carga de lo especificado AFI.

Comando:

```
aws ec2 reset-fpga-image-attribute --fpga-image-id afi-0d123e123bfc85abc --  
attribute LoadPermission
```

Salida:

```
{  
  "Return": true  
}
```

- Para API obtener más información, consulte [ResetFpgaImageAttribute](#) la Referencia de AWS CLI comandos.

## **reset-image-attribute**

En el siguiente ejemplo de código, se muestra cómo usar `reset-image-attribute`.

AWS CLI

Para restablecer el `launchPermission` atributo

En este ejemplo, se restablece el `launchPermission` atributo del especificado AMI a su valor predeterminado. De forma predeterminada, AMIs son privados. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 reset-image-attribute --image-id ami-5731123e --attribute launchPermission
```

- Para API obtener más información, consulte [ResetImageAttribute](#) la Referencia de AWS CLI comandos.

## **reset-instance-attribute**

En el siguiente ejemplo de código, se muestra cómo usar `reset-instance-attribute`.

## AWS CLI

Para restablecer el `sourceDestCheck` atributo

En este ejemplo se restablece el `sourceDestCheck` atributo de la instancia especificada. La instancia debe estar en un VPC. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 reset-instance-attribute --instance-id i-1234567890abcdef0 --  
attribute sourceDestCheck
```

Para restablecer el atributo del núcleo

Este ejemplo restablece el `kernel` atributo de la instancia especificada. La instancia debe tener el estado `stopped`. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 reset-instance-attribute --instance-id i-1234567890abcdef0 --  
attribute kernel
```

Para restablecer el atributo `ramdisk`

En este ejemplo se restablece el `ramdisk` atributo de la instancia especificada. La instancia debe tener el estado `stopped`. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 reset-instance-attribute --instance-id i-1234567890abcdef0 --  
attribute ramdisk
```

- Para API obtener más información, consulte [ResetInstanceAttribute](#) la Referencia de AWS CLI comandos.

## **reset-network-interface-attribute**

En el siguiente ejemplo de código, se muestra cómo usar `reset-network-interface-attribute`.

## AWS CLI

Para restablecer un atributo de la interfaz de red

El siguiente `reset-network-interface-attribute` ejemplo restablece el valor del atributo de comprobación de origen/destino en. `true`

```
aws ec2 reset-network-interface-attribute \  
  --network-interface-id eni-686ea200 \  
  --source-dest-check
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte la Referencia de [ResetNetworkInterfaceAttribute](#) comandos AWS CLI .

## `reset-snapshot-attribute`

En el siguiente ejemplo de código, se muestra cómo usar `reset-snapshot-attribute`.

## AWS CLI

Para restablecer un atributo de instantánea

En este ejemplo, se restablecen los permisos de creación de volúmenes para la instantánea `snap-1234567890abcdef0`. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 reset-snapshot-attribute --snapshot-id snap-1234567890abcdef0 --  
attribute createVolumePermission
```

- Para API obtener más información, consulte [ResetSnapshotAttribute](#) la Referencia de AWS CLI comandos.

## `restore-address-to-classic`

En el siguiente ejemplo de código, se muestra cómo usar `restore-address-to-classic`.

## AWS CLI

Para restaurar una dirección a EC2 -Classic

En este ejemplo, se restaura la dirección IP elástica 198.51.100.0 en la plataforma -Classic. EC2

Comando:

```
aws ec2 restore-address-to-classic --public-ip 198.51.100.0
```

Salida:

```
{
  "Status": "MoveInProgress",
  "PublicIp": "198.51.100.0"
}
```

- Para obtener API más información, consulte la Referencia de comandos [RestoreAddressToClassic](#).AWS CLI

## restore-image-from-recycle-bin

En el siguiente ejemplo de código, se muestra cómo usar `restore-image-from-recycle-bin`.

### AWS CLI

Para restaurar una imagen de la papelera de reciclaje

El siguiente `restore-image-from-recycle-bin` ejemplo restaura el AMI `ami-0111222333444abcd` de la papelera de reciclaje.

```
aws ec2 restore-image-from-recycle-bin \
  --image-id ami-0111222333444abcd
```

Salida:

```
{
  "Return": true
}
```

Para obtener más información, consulte [Recuperarse AMLs de la papelera de reciclaje](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte [RestoreImageFromRecycleBin](#) la Referencia de AWS CLI comandos.

## restore-managed-prefix-list-version

En el siguiente ejemplo de código, se muestra cómo usar `restore-managed-prefix-list-version`.

### AWS CLI

US-WEST-2\*\*Para restaurar una versión de lista de prefijos\*\*

`restore-managed-prefix-list-version`A continuación, se restauran las entradas de la versión 1 de la lista de prefijos especificada.

```
aws ec2 restore-managed-prefix-list-version \
  --prefix-list-id pl-0123456abcabc1 \
  --current-version 2 \
  --previous-version 1
```

Salida:

```
{
  "PrefixList": {
    "PrefixListId": "pl-0123456abcabc1",
    "AddressFamily": "IPv4",
    "State": "restore-in-progress",
    "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-0123456abcabc1",
    "PrefixListName": "vpc-cidrs",
    "MaxEntries": 10,
    "Version": 2,
    "OwnerId": "123456789012"
  }
}
```

Para obtener más información, consulta [Listas de prefijos gestionadas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [RestoreManagedPrefixListVersion](#) la Referencia de AWS CLI comandos.

## **restore-snapshot-from-recycle-bin**

En el siguiente ejemplo de código, se muestra cómo usar `restore-snapshot-from-recycle-bin`.

### AWS CLI

Para restaurar las instantáneas de la papelera de reciclaje

El siguiente `restore-snapshot-from-recycle-bin` ejemplo restaura una instantánea de la papelera de reciclaje. Cuando restaura una instantánea desde la papelera de reciclaje, la instantánea está disponible inmediatamente para su uso y se quita de la papelera de reciclaje. Puede utilizar una instantánea restaurada de la misma manera en que utiliza cualquier otra instantánea de la cuenta.

```
aws ec2 restore-snapshot-from-recycle-bin \  
  --snapshot-id snap-01234567890abcdef
```

Este comando no genera ninguna salida.

Para obtener más información sobre la papelera de reciclaje de AmazonEBS, consulte [Recuperar instantáneas de la papelera de reciclaje](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [RestoreSnapshotFromRecycleBin](#) la Referencia de AWS CLI comandos.

## **restore-snapshot-tier**

En el siguiente ejemplo de código, se muestra cómo usar `restore-snapshot-tier`.

### AWS CLI

Ejemplo 1: Para restaurar permanentemente una instantánea archivada

El siguiente `restore-snapshot-tier` ejemplo restaura permanentemente la instantánea especificada. Especifique la `permanent-restore` opción `--snapshot-id` e inclúyala.

```
aws ec2 restore-snapshot-tier \  
  --snapshot-id snap-01234567890abcdef
```

```
--snapshot-id snap-01234567890abcdef \  
--permanent-restore
```

Salida:

```
{  
  "SnapshotId": "snap-01234567890abcdef",  
  "IsPermanentRestore": true  
}
```

Para obtener más información sobre el archivado de instantáneas, consulte Archivar EBS instantáneas de Amazon <<https://docs.aws.amazon.com/AWS EC2/latest/ /snapshot-archive.html> UserGuide > en la Guía del usuario de Amazon. EC2

Ejemplo 2: Para restaurar temporalmente una instantánea archivada

El siguiente `restore-snapshot-tier` ejemplo restaura temporalmente la instantánea especificada. Omita la opción `--permanent-restore`. Especifique `--snapshot-id` y, `paratemporary-restore-days`, especifique el número de días durante los que se va a restaurar la instantánea. `temporary-restore-days` debe especificarse en días. El rango permitido es 1 de 180. Si no especifica ningún valor, el valor predeterminado es 1 día.

```
aws ec2 restore-snapshot-tier \  
  --snapshot-id snap-01234567890abcdef \  
  --temporary-restore-days 5
```

Salida:

```
{  
  "SnapshotId": "snap-01234567890abcdef",  
  "RestoreDuration": 5,  
  "IsPermanentRestore": false  
}
```

Para obtener más información sobre el archivado de instantáneas, consulte Archivar EBS instantáneas de Amazon <<https://docs.aws.amazon.com/AWS EC2/latest/ /snapshot-archive.html> UserGuide > en la Guía del usuario de Amazon. EC2

Ejemplo 3: Para modificar el período de restauración



En el siguiente `restore-snapshot-tier` ejemplo, se cambia el período de restauración de la instantánea especificada a 10 días.

```
aws ec2 restore-snapshot-tier \  
  --snapshot-id snap-01234567890abcdef \  
  --temporary-restore-days 10
```

Salida:

```
{  
  "SnapshotId": "snap-01234567890abcdef",  
  "RestoreDuration": 10,  
  "IsPermanentRestore": false  
}
```

Para obtener más información sobre el archivado de instantáneas, consulte Archivar EBS instantáneas de Amazon <<https://docs.aws.amazon.com/AWS-EC2/latest/userguide/snapshot-archive.html>> en la Guía del usuario de Amazon. EC2

Ejemplo 4: Para modificar el tipo de restauración

En el siguiente `restore-snapshot-tier` ejemplo, se cambia el tipo de restauración de la instantánea especificada de temporal a permanente.

```
aws ec2 restore-snapshot-tier \  
  --snapshot-id snap-01234567890abcdef \  
  --permanent-restore
```

Salida:

```
{  
  "SnapshotId": "snap-01234567890abcdef",  
  "IsPermanentRestore": true  
}
```

Para obtener más información sobre el archivado de instantáneas, consulte Archivar EBS instantáneas de Amazon <<https://docs.aws.amazon.com/AWS-EC2/latest/userguide/snapshot-archive.html>> en la Guía del usuario de Amazon. EC2

- Para obtener API más información, consulte la Referencia de comandos.

[RestoreSnapshotTier](#) AWS CLI

## revoke-client-vpn-ingress

En el siguiente ejemplo de código, se muestra cómo usar `revoke-client-vpn-ingress`.

### AWS CLI

Para revocar una regla de autorización para un punto final del cliente VPN

El siguiente `revoke-client-vpn-ingress` ejemplo revoca una regla de acceso a Internet (`0.0.0.0/0`) para todos los grupos.

```
aws ec2 revoke-client-vpn-ingress \  
  --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \  
  --target-network-cidr 0.0.0.0/0 --revoke-all-groups
```

Salida:

```
{  
  "Status": {  
    "Code": "revoking"  
  }  
}
```

Para obtener más información, consulte [las reglas de autorización](#) en la Guía del VPN administrador de AWS clientes.

- Para API obtener más información, consulte [RevokeClientVpnIngress](#) la Referencia de AWS CLI comandos.

## revoke-security-group-egress

En el siguiente ejemplo de código, se muestra cómo usar `revoke-security-group-egress`.

### AWS CLI

Ejemplo 1: Para eliminar la regla que permite el tráfico saliente a un rango de direcciones específico

El siguiente comando de `revoke-security-group-egress` ejemplo elimina la regla que concede acceso a los rangos de direcciones especificados en el TCP puerto 80.

```
aws ec2 revoke-security-group-egress \  
  --security-group-id sg-12345678 --target-cidr 0.0.0.0/0 --target-ports 80
```

```
--group-id sg-026c12253ce15eff7 \  
--ip-  
permissions ["IpProtocol=tcp,FromPort=80,ToPort=80,IpRanges=[{CidrIp=10.0.0.0/16}]"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Grupos de seguridad](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Eliminar la regla que permite el tráfico saliente a un grupo de seguridad específico

El siguiente comando de `revoke-security-group-egress` ejemplo elimina la regla que concede acceso al grupo de seguridad especificado en el TCP puerto 80.

```
aws ec2 revoke-security-group-egress \  
--group-id sg-026c12253ce15eff7 \  
--ip-permissions '["IpProtocol": "tcp", "FromPort": 443, "ToPort":  
443,"UserIdGroupPairs": [{"GroupId": "sg-06df23a01ff2df86d"}]']
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Grupos de seguridad](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [RevokeSecurityGroupEgress](#) la Referencia de AWS CLI comandos.

## **revoke-security-group-ingress**

En el siguiente ejemplo de código, se muestra cómo usar `revoke-security-group-ingress`.

### AWS CLI

Ejemplo 1: Para eliminar una regla de un grupo de seguridad

El siguiente `revoke-security-group-ingress` ejemplo elimina el acceso al TCP puerto 22 al rango de `203.0.113.0/24` direcciones del grupo de seguridad especificado de forma predeterminada VPC.

```
aws ec2 revoke-security-group-ingress \  
--group-name mySecurityGroup
```

```
--protocol tcp \  
--port 22 \  
--cidr 203.0.113.0/24
```

Este comando no produce ningún resultado si se ejecuta correctamente.

Para obtener más información, consulte [Grupos de seguridad](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para eliminar una regla mediante el conjunto de permisos de IP

En el siguiente `revoke-security-group-ingress` ejemplo, se utiliza el `ip-permissions` parámetro para eliminar una regla de entrada que permite el ICMP mensaje `Destination Unreachable: Fragmentation Needed and Don't Fragment was Set` (tipo 3, código 4).

```
aws ec2 revoke-security-group-ingress \  
  --group-id sg-026c12253ce15eff7 \  
  --ip-  
permissions IpProtocol=icmp,FromPort=3,ToPort=4,IpRanges=[{CidrIp=0.0.0.0/0}]
```

Este comando no produce ningún resultado si se ejecuta correctamente.

Para obtener más información, consulte [Grupos de seguridad](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [RevokeSecurityGroupIngress](#) la Referencia de AWS CLI comandos.

## run-instances

En el siguiente ejemplo de código, se muestra cómo usar `run-instances`.

### AWS CLI

Ejemplo 1: Lanzar una instancia en una subred predeterminada

En el siguiente `run-instances` ejemplo, se lanza una única instancia de este tipo `t2.micro` en la subred predeterminada de la región actual y se asocia a la subred predeterminada de la región predeterminadaVPC. El par de claves es opcional si no tienes pensado conectarte a la instancia mediante SSH (Linux) o RDP (Windows).

```
aws ec2 run-instances \  
  --image-id ami-0abcdef1234567890 \  
  --instance-type t2.micro \  
  --key-name MyKeyPair
```

Salida:

```
{  
  "Instances": [  
    {  
      "AmiLaunchIndex": 0,  
      "ImageId": "ami-0abcdef1234567890",  
      "InstanceId": "i-1231231230abcdef0",  
      "InstanceType": "t2.micro",  
      "KeyName": "MyKeyPair",  
      "LaunchTime": "2018-05-10T08:05:20.000Z",  
      "Monitoring": {  
        "State": "disabled"  
      },  
      "Placement": {  
        "AvailabilityZone": "us-east-2a",  
        "GroupName": "",  
        "Tenancy": "default"  
      },  
      "PrivateDnsName": "ip-10-0-0-157.us-east-2.compute.internal",  
      "PrivateIpAddress": "10.0.0.157",  
      "ProductCodes": [],  
      "PublicDnsName": "",  
      "State": {  
        "Code": 0,  
        "Name": "pending"  
      },  
      "StateTransitionReason": "",  
      "SubnetId": "subnet-04a636d18e83cfacb",  
      "VpcId": "vpc-1234567890abcdef0",  
      "Architecture": "x86_64",  
      "BlockDeviceMappings": [],  
      "ClientToken": "",  
      "EbsOptimized": false,  
      "Hypervisor": "xen",  
      "NetworkInterfaces": [  
        {  
          "Attachment": {
```

```
        "AttachTime": "2018-05-10T08:05:20.000Z",
        "AttachmentId": "eni-attach-0e325c07e928a0405",
        "DeleteOnTermination": true,
        "DeviceIndex": 0,
        "Status": "attaching"
    },
    "Description": "",
    "Groups": [
        {
            "GroupName": "MySecurityGroup",
            "GroupId": "sg-0598c7d356eba48d7"
        }
    ],
    "Ipv6Addresses": [],
    "MacAddress": "0a:ab:58:e0:67:e2",
    "NetworkInterfaceId": "eni-0c0a29997760baee7",
    "OwnerId": "123456789012",
    "PrivateDnsName": "ip-10-0-0-157.us-east-2.compute.internal",
    "PrivateIpAddress": "10.0.0.157",
    "PrivateIpAddresses": [
        {
            "Primary": true,
            "PrivateDnsName": "ip-10-0-0-157.us-
east-2.compute.internal",
            "PrivateIpAddress": "10.0.0.157"
        }
    ],
    "SourceDestCheck": true,
    "Status": "in-use",
    "SubnetId": "subnet-04a636d18e83cfacb",
    "VpcId": "vpc-1234567890abcdef0",
    "InterfaceType": "interface"
    }
],
"RootDeviceName": "/dev/xvda",
"RootDeviceType": "ebs",
"SecurityGroups": [
    {
        "GroupName": "MySecurityGroup",
        "GroupId": "sg-0598c7d356eba48d7"
    }
],
"SourceDestCheck": true,
"StateReason": {
```

```

        "Code": "pending",
        "Message": "pending"
    },
    "Tags": [],
    "VirtualizationType": "hvm",
    "CpuOptions": {
        "CoreCount": 1,
        "ThreadsPerCore": 1
    },
    "CapacityReservationSpecification": {
        "CapacityReservationPreference": "open"
    },
    "MetadataOptions": {
        "State": "pending",
        "HttpTokens": "optional",
        "HttpPutResponseHopLimit": 1,
        "HttpEndpoint": "enabled"
    }
}
],
"OwnerId": "123456789012",
"ReservationId": "r-02a3f596d91211712"
}

```

Ejemplo 2: Lanzar una instancia en una subred no predeterminada y agregar una dirección IP pública

En el siguiente ejemplo de `run-instances`, se solicita una dirección IP pública para una instancia que se va a lanzar en una subred no predeterminada. La instancia se asocia a los grupos de seguridad especificados.

```

aws ec2 run-instances \
  --image-id ami-0abcdef1234567890 \
  --instance-type t2.micro \
  --subnet-id subnet-08fc749671b2d077c \
  --security-group-ids sg-0b0384b66d7d692f9 \
  --associate-public-ip-address \
  --key-name MyKeyPair

```

Para ver un ejemplo del resultado de `run-instances`, consulte el ejemplo 1.

Ejemplo 3: Lanzar una instancia con volúmenes adicionales

En el siguiente ejemplo de `run-instances`, se usa una asignación de dispositivos de bloques, especificada en `mapping.json`, para asociar volúmenes adicionales en el momento del lanzamiento. Un mapeo de dispositivos de bloques puede especificar EBS volúmenes, volúmenes de almacenes de instancias o tanto EBS volúmenes como volúmenes de almacenes de instancias.

```
aws ec2 run-instances \  
  --image-id ami-0abcdef1234567890 \  
  --instance-type t2.micro \  
  --subnet-id subnet-08fc749671b2d077c \  
  --security-group-ids sg-0b0384b66d7d692f9 \  
  --key-name MyKeyPair \  
  --block-device-mappings file://mapping.json
```

Contenido de `mapping.json`. En este ejemplo se agrega `/dev/sdh` un EBS volumen vacío con un tamaño de 100 GiB.

```
[  
  {  
    "DeviceName": "/dev/sdh",  
    "Ebs": {  
      "VolumeSize": 100  
    }  
  }  
]
```

Contenido de `mapping.json`. En este ejemplo, se le agrega a `ephemeral1` un volumen de almacén de instancias.

```
[  
  {  
    "DeviceName": "/dev/sdc",  
    "VirtualName": "ephemeral1"  
  }  
]
```

Para ver un ejemplo del resultado de `run-instances`, consulte el ejemplo 1.

Para obtener más información sobre las asignaciones de dispositivos de bloques, consulta la sección [Asignación de dispositivos de bloques](#) en la Guía EC2 del usuario de Amazon.



#### Ejemplo 4: Lanzar una instancia y agregar etiquetas al crearla

En el siguiente ejemplo de `run-instances`, se agrega una etiqueta con una clave de `webserver` y un valor de `production` a la instancia. El comando también aplica una etiqueta con una clave `cost-center` y un valor de `cc123` a cualquier EBS volumen que se cree (en este caso, el volumen raíz).

```
aws ec2 run-instances \  
  --image-id ami-0abcdef1234567890 \  
  --instance-type t2.micro \  
  --count 1 \  
  --subnet-id subnet-08fc749671b2d077c \  
  --key-name MyKeyPair \  
  --security-group-ids sg-0b0384b66d7d692f9 \  
  --tag-specifications  
  'ResourceType=instance,Tags=[{Key=webserver,Value=production}]'  
  'ResourceType=volume,Tags=[{Key=cost-center,Value=cc123}]'
```

Para ver un ejemplo del resultado de `run-instances`, consulte el ejemplo 1.

#### Ejemplo 5: Lanzar una instancia con datos de usuario

En el siguiente ejemplo de `run-instances`, se transfieren los datos del usuario a un archivo denominado `my_script.txt` que contiene un script de configuración para la instancia. El script se ejecuta en el momento del lanzamiento.

```
aws ec2 run-instances \  
  --image-id ami-0abcdef1234567890 \  
  --instance-type t2.micro \  
  --count 1 \  
  --subnet-id subnet-08fc749671b2d077c \  
  --key-name MyKeyPair \  
  --security-group-ids sg-0b0384b66d7d692f9 \  
  --user-data file://my_script.txt
```

Para ver un ejemplo del resultado de `run-instances`, consulte el ejemplo 1.

Para obtener más información sobre los datos de usuarios de instancias, consulta [Cómo trabajar con datos de usuarios de instancias](#) en la Guía del EC2 usuario de Amazon.

#### Ejemplo 6: Lanzar una instancia de rendimiento ampliable

En el siguiente ejemplo de `run-instances`, se lanza una instancia `t2.micro` con la opción de crédito `unlimited`. Al lanzar una instancia T2, si no especifica `--credit-specification`, la opción de crédito predeterminada es `standard`. Al lanzar una instancia T3, la opción de crédito predeterminada es `unlimited`.

```
aws ec2 run-instances \  
  --image-id ami-0abcdef1234567890 \  
  --instance-type t2.micro \  
  --count 1 \  
  --subnet-id subnet-08fc749671b2d077c \  
  --key-name MyKeyPair \  
  --security-group-ids sg-0b0384b66d7d692f9 \  
  --credit-specification CpuCredits=unlimited
```

Para ver un ejemplo del resultado de `run-instances`, consulte el ejemplo 1.

Para obtener más información sobre las instancias de rendimiento con ráfagas, consulte Instancias de [rendimiento con ráfagas](#) en la Guía EC2 del usuario de Amazon.

- Para API obtener más información, consulte la Referencia [RunInstances](#) de AWS CLI comandos.

## run-scheduled-instances

En el siguiente ejemplo de código, se muestra cómo usar `run-scheduled-instances`.

### AWS CLI

Para lanzar una instancia programada

En este ejemplo, se lanza la instancia programada especificada en un VPC.

Comando:

```
aws ec2 run-scheduled-instances --scheduled-instance-  
id sci-1234-1234-1234-1234-123456789012 --instance-count 1 --launch-  
specification file://launch-specification.json
```

Launch-Specification.json:

```
{  
  "ImageId": "ami-12345678",
```

```
"KeyName": "my-key-pair",
"InstanceType": "c4.large",
"NetworkInterfaces": [
  {
    "DeviceIndex": 0,
    "SubnetId": "subnet-12345678",
    "AssociatePublicIpAddress": true,
    "Groups": ["sg-12345678"]
  }
],
"IamInstanceProfile": {
  "Name": "my-iam-role"
}
}
```

Salida:

```
{
  "InstanceIdSet": [
    "i-1234567890abcdef0"
  ]
}
```

En este ejemplo, se lanza la instancia programada especificada en `-Classic`. EC2

Comando:

```
aws ec2 run-scheduled-instances --scheduled-instance-
id sci-1234-1234-1234-1234-123456789012 --instance-count 1 --launch-
specification file://launch-specification.json
```

Launch-Specification.json:

```
{
  "ImageId": "ami-12345678",
  "KeyName": "my-key-pair",
  "SecurityGroupIds": ["sg-12345678"],
  "InstanceType": "c4.large",
  "Placement": {
    "AvailabilityZone": "us-west-2b"
  }
  "IamInstanceProfile": {
```

```

    "Name": "my-iam-role"
  }
}

```

Salida:

```

{
  "InstanceIdSet": [
    "i-1234567890abcdef0"
  ]
}

```

- Para obtener API más información, consulte la Referencia de comandos.  
[RunScheduledInstances](#) AWS CLI

## search-local-gateway-routes

En el siguiente ejemplo de código, se muestra cómo usar `search-local-gateway-routes`.

AWS CLI

Para buscar rutas en una tabla de rutas de una puerta de enlace local

El siguiente `search-local-gateway-routes` ejemplo busca rutas estáticas en la tabla de rutas de la puerta de enlace local especificada.

```

aws ec2 search-local-gateway-routes \
  --local-gateway-route-table-id lgw-rtb-059615ef7dEXAMPLE \
  --filters "Name=type,Values=static"

```

Salida:

```

{
  "Route": {
    "DestinationCidrBlock": "0.0.0.0/0",
    "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
    "Type": "static",
    "State": "deleted",
    "LocalGatewayRouteTableId": "lgw-rtb-059615ef7EXAMPLE"
  }
}

```

- Para API obtener más información, consulte [SearchLocalGatewayRoutes](#) la Referencia de AWS CLI comandos.

## search-transit-gateway-multicast-groups

En el siguiente ejemplo de código, se muestra cómo usar `search-transit-gateway-multicast-groups`.

### AWS CLI

Para buscar uno o más grupos de multidifusión de Transit Gateway y devolver la información de pertenencia al grupo

El siguiente `search-transit-gateway-multicast-groups` ejemplo devuelve la pertenencia al grupo de multidifusión especificado.

```
aws ec2 search-transit-gateway-multicast-groups \
  --transit-gateway-multicast-domain-id tgw-mcast-domain-000fb24d04EXAMPLE
```

Salida:

```
{
  "MulticastGroups": [
    {
      "GroupIpAddress": "224.0.1.0",
      "TransitGatewayAttachmentId": "tgw-attach-0372e72386EXAMPLE",
      "SubnetId": "subnet-0187aff814EXAMPLE",
      "ResourceId": "vpc-0065acced4EXAMPLE",
      "ResourceType": "vpc",
      "NetworkInterfaceId": "eni-03847706f6EXAMPLE",
      "GroupMember": false,
      "GroupSource": true,
      "SourceType": "static"
    }
  ]
}
```

Para obtener más información, consulte [Administración de grupos de multidifusión](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte la Referencia [SearchTransitGatewayMulticastGroups](#) de AWS CLI comandos.

## search-transit-gateway-routes

En el siguiente ejemplo de código, se muestra cómo usar `search-transit-gateway-routes`.

### AWS CLI

Para buscar rutas en la tabla de rutas de la pasarela de tránsito especificada

El siguiente `search-transit-gateway-routes` ejemplo devuelve todas las rutas que son del tipo de `static` la tabla de rutas especificada.

```
aws ec2 search-transit-gateway-routes \
  --transit-gateway-route-table-id tgw-rtb-0a823edbdeEXAMPLE \
  --filters "Name=type,Values=static"
```

Salida:

```
{
  "Routes": [
    {
      "DestinationCidrBlock": "10.0.2.0/24",
      "TransitGatewayAttachments": [
        {
          "ResourceId": "vpc-4EXAMPLE",
          "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",
          "ResourceType": "vpc"
        }
      ],
      "Type": "static",
      "State": "active"
    },
    {
      "DestinationCidrBlock": "10.1.0.0/24",
      "TransitGatewayAttachments": [
        {
          "ResourceId": "vpc-4EXAMPLE",
          "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",
          "ResourceType": "vpc"
        }
      ],
      "Type": "static",
      "State": "active"
    }
  ]
}
```

```
  ],  
  "AdditionalRoutesAvailable": false  
}
```

Para obtener más información, consulte [las tablas de rutas de Transit Gateways](#) en la Guía de Transit Gateways.

- Para API obtener más información, consulte [SearchTransitGatewayRoutes](#) la Referencia de AWS CLI comandos.

## send-diagnostic-interrupt

En el siguiente ejemplo de código, se muestra cómo usar `send-diagnostic-interrupt`.

### AWS CLI

Para enviar una interrupción de diagnóstico

En el siguiente `send-diagnostic-interrupt` ejemplo, se envía una interrupción de diagnóstico a la instancia especificada.

```
aws ec2 send-diagnostic-interrupt \  
  --instance-id i-1234567890abcdef0
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SendDiagnosticInterrupt](#) la Referencia de AWS CLI comandos.

## start-instances

En el siguiente ejemplo de código, se muestra cómo usar `start-instances`.

### AWS CLI

Para iniciar una EC2 instancia de Amazon

En este ejemplo, se inicia la instancia EBS respaldada por Amazon especificada.

Comando:

```
aws ec2 start-instances --instance-ids i-1234567890abcdef0
```

Salida:

```
{
  "StartingInstances": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "CurrentState": {
        "Code": 0,
        "Name": "pending"
      },
      "PreviousState": {
        "Code": 80,
        "Name": "stopped"
      }
    }
  ]
}
```

Para obtener más información, consulte Detener e iniciar la instancia en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para API obtener más información, consulte [StartInstances](#) la Referencia de AWS CLI comandos.

## start-network-insights-access-scope-analysis

En el siguiente ejemplo de código, se muestra cómo usar `start-network-insights-access-scope-analysis`.

AWS CLI

Para iniciar un análisis del alcance del acceso a Network Insights

En el siguiente `start-network-insights-access-scope-analysis` ejemplo, se inicia el análisis del alcance en su AWS cuenta.

```
aws ec2 start-network-insights-access-scope-analysis \
  --region us-east-1 \
  --network-insights-access-scope-id nis-123456789111
```

Salida:



```
{
  "NetworkInsightsAccessScopeAnalysis": {
    "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789222",
    "NetworkInsightsAccessScopeAnalysisArn": "arn:aws:ec2:us-
east-1:123456789012:network-insights-access-scope-analysis/nisa-123456789222",
    "NetworkInsightsAccessScopeId": "nis-123456789111",
    "Status": "running",
    "StartDate": "2022-01-26T00:47:06.814000+00:00"
  }
}
```

Para obtener más información, consulte [Cómo empezar a utilizar Network Access Analyzer mediante AWS CLI](#) la guía Network Access Analyzer.

- Para API obtener más información, consulte la Referencia [StartNetworkInsightsAccessScopeAnalysis](#) de AWS CLI comandos.

## start-network-insights-analysis

En el siguiente ejemplo de código, se muestra cómo usar `start-network-insights-analysis`.

### AWS CLI

Para analizar una ruta

El siguiente `start-network-insights-analysis` ejemplo analiza la ruta entre el origen y el destino. Para ver los resultados del análisis de la ruta, utilice el `describe-network-insights-analyses` comando.

```
aws ec2 start-network-insights-analysis \
  --network-insights-path-id nip-0b26f224f1d131fa8
```

Salida:

```
{
  "NetworkInsightsAnalysis": {
    "NetworkInsightsAnalysisId": "nia-02207aa13eb480c7a",
    "NetworkInsightsAnalysisArn": "arn:aws:ec2:us-east-1:123456789012:network-
insights-analysis/nia-02207aa13eb480c7a",
    "NetworkInsightsPathId": "nip-0b26f224f1d131fa8",
    "StartDate": "2021-01-20T22:58:37.495Z",
    "Status": "running"
  }
}
```

```
}  
}
```

Para obtener más información, consulte [Cómo empezar a utilizar la Guía del AWS CLI Reachability Analyzer](#).

- Para API obtener más información, consulte la Referencia de [StartNetworkInsightsAnalysis](#) comandos AWS CLI .

## **start-vpc-endpoint-service-private-dns-verification**

En el siguiente ejemplo de código, se muestra cómo usar `start-vpc-endpoint-service-private-dns-verification`.

### AWS CLI

Para iniciar el proceso DNS de verificación

El siguiente `start-vpc-endpoint-service-private-dns-verification` ejemplo inicia el proceso de DNS verificación para el servicio de punto final especificado.

```
aws ec2 start-vpc-endpoint-service-private-dns-verification \  
  --service-id vpce-svc-071afff70666e61e0
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar DNS nombres](#) en la Guía del AWS PrivateLink usuario.

- Para API obtener más información, consulte [StartVpcEndpointServicePrivateDnsVerification](#) la Referencia de AWS CLI comandos.

## **stop-instances**

En el siguiente ejemplo de código, se muestra cómo usar `stop-instances`.

### AWS CLI

Ejemplo 1: Para detener una EC2 instancia de Amazon

El siguiente `stop-instances` ejemplo detiene la instancia EBS respaldada por Amazon especificada.

```
aws ec2 stop-instances \  
  --instance-ids i-1234567890abcdef0
```

Salida:

```
{  
  "StoppingInstances": [  
    {  
      "InstanceId": "i-1234567890abcdef0",  
      "CurrentState": {  
        "Code": 64,  
        "Name": "stopping"  
      },  
      "PreviousState": {  
        "Code": 16,  
        "Name": "running"  
      }  
    }  
  ]  
}
```

Para obtener más información, consulte [Detener e iniciar la instancia](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

Ejemplo 2: Para hibernar una instancia de Amazon EC2

En el siguiente `stop-instances` ejemplo, se hiberna la instancia EBS respaldada por Amazon si la instancia está habilitada para la hibernación y cumple los requisitos previos de hibernación. Luego de que la instancia sea puesta en hibernación, la instancia se detiene.

```
aws ec2 stop-instances \  
  --instance-ids i-1234567890abcdef0 \  
  --hibernate
```

Salida:

```
{  
  "StoppingInstances": [  
    {  
      "CurrentState": {  
        "Code": 64,
```

```

        "Name": "stopping"
      },
      "InstanceId": "i-1234567890abcdef0",
      "PreviousState": {
        "Code": 16,
        "Name": "running"
      }
    }
  ]
}

```

Para obtener más información, consulte [Hibernar la instancia de Linux bajo demanda](#) en la Guía del usuario de Amazon Elastic Compute Cloud.

- Para obtener API más información, consulte la Referencia de comandos. [StopInstances](#) AWS CLI

## terminate-client-vpn-connections

En el siguiente ejemplo de código, se muestra cómo usar `terminate-client-vpn-connections`.

### AWS CLI

Para finalizar una conexión a un VPN punto final del cliente

El siguiente `terminate-client-vpn-connections` ejemplo finaliza la conexión especificada con el VPN punto final del cliente.

```

aws ec2 terminate-client-vpn-connections \
  --client-vpn-endpoint-id vpn-endpoint-123456789123abcde \
  --connection-id cvpn-connection-04edd76f5201e0cb8

```

Salida:

```

{
  "ClientVpnEndpointId": "vpn-endpoint-123456789123abcde",
  "ConnectionStatuses": [
    {
      "ConnectionId": "cvpn-connection-04edd76f5201e0cb8",
      "PreviousStatus": {
        "Code": "active"
      }
    }
  ]
}

```

```
    },
    "CurrentStatus": {
      "Code": "terminating"
    }
  ]
}
```

Para obtener más información, consulte [Conexiones de clientes](#) en la Guía del VPN administrador de AWS clientes.

- Para API obtener más información, consulte [TerminateClientVpnConnections](#) la Referencia de AWS CLI comandos.

## terminate-instances

En el siguiente ejemplo de código, se muestra cómo usar `terminate-instances`.

### AWS CLI

Para cerrar una EC2 instancia de Amazon

En este ejemplo, se termina la instancia especificada.

Comando:

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0
```

Salida:

```
{
  "TerminatingInstances": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "CurrentState": {
        "Code": 32,
        "Name": "shutting-down"
      },
      "PreviousState": {
        "Code": 16,
        "Name": "running"
      }
    }
  ]
}
```

```
    }  
  ]  
}
```

Para obtener más información, consulte [Uso de Amazon EC2 Instances](#) en la Guía del usuario de la interfaz de línea de AWS comandos.

- Para API obtener más información, consulte [TerminateInstances](#) la Referencia de AWS CLI comandos.

## **unassign-ipv6-addresses**

En el siguiente ejemplo de código, se muestra cómo usar `unassign-ipv6-addresses`.

### AWS CLI

Para anular la asignación de una IPv6 dirección de una interfaz de red

En este ejemplo, se anula la asignación de la IPv6 dirección especificada de la interfaz de red especificada.

Comando:

```
aws ec2 unassign-ipv6-addresses --ipv6-  
addresses 2001:db8:1234:1a00:3304:8879:34cf:4071 --network-interface-id eni-23c49b68
```

Salida:

```
{  
  "NetworkInterfaceId": "eni-23c49b68",  
  "UnassignedIpv6Addresses": [  
    "2001:db8:1234:1a00:3304:8879:34cf:4071"  
  ]  
}
```

- Para API obtener más información, consulte [UnassignIpv6 direcciones](#) en AWS CLI la referencia de comandos.

## **unassign-private-ip-addresses**

En el siguiente ejemplo de código, se muestra cómo usar `unassign-private-ip-addresses`.

## AWS CLI

Para anular la asignación de una dirección IP privada secundaria de una interfaz de red

En este ejemplo, se anula la asignación de la dirección IP privada especificada de la interfaz de red especificada. Si el comando se ejecuta correctamente, no se muestra ningún resultado.

Comando:

```
aws ec2 unassign-private-ip-addresses --network-interface-id eni-e5aa89a3 --private-ip-addresses 10.0.0.82
```

- Para API obtener más información, consulte la Referencia [UnassignPrivateIpAddresses](#) de AWS CLI comandos.

## unassign-private-nat-gateway-address

En el siguiente ejemplo de código, se muestra cómo usar `unassign-private-nat-gateway-address`.

## AWS CLI

Para anular la asignación de una dirección IP privada de su puerta de enlace privada NAT

En el siguiente `unassign-private-nat-gateway-address` ejemplo, se anula la asignación de la dirección IP especificada de la puerta de enlace privada especificada. NAT

```
aws ec2 unassign-private-nat-gateway-address \  
  --nat-gateway-id nat-1234567890abcdef0 \  
  --private-ip-addresses 10.0.20.197
```

Salida:

```
{  
  "NatGatewayId": "nat-0ee3edd182361f662",  
  "NatGatewayAddresses": [  
    {  
      "NetworkInterfaceId": "eni-0065a61b324d1897a",  
      "PrivateIp": "10.0.20.197",  
      "IsPrimary": false,  
      "Status": "unassigning"  
    }  
  ]  
}
```

```
]
}
```

Para obtener más información, consulta [NATlas pasarelas](#) en la Guía del VPC usuario de Amazon.

- Para API obtener más información, consulte [UnassignPrivateNatGatewayAddress](#) la Referencia de AWS CLI comandos.

## **unlock-snapshot**

En el siguiente ejemplo de código, se muestra cómo usar `unlock-snapshot`.

### AWS CLI

Para desbloquear una instantánea

En el siguiente `unlock-snapshot` ejemplo, se desbloquea la instantánea especificada.

```
aws ec2 unlock-snapshot \  
  --snapshot-id snap-0b5e733b4a8df6e0d
```

Salida:

```
{  
  "SnapshotId": "snap-0b5e733b4a8df6e0d"  
}
```

Para obtener más información, consulta [Snapshot Lock](#) en la Guía del EBS usuario de Amazon.

- Para API obtener más información, consulte [UnlockSnapshot](#) la Referencia de AWS CLI comandos.

## **unmonitor-instances**

En el siguiente ejemplo de código, se muestra cómo usar `unmonitor-instances`.

### AWS CLI

Deshabilitar el monitoreo detallado de una instancia

Este comando de ejemplo deshabilita el monitoreo detallado de la instancia especificada.



Comando:

```
aws ec2 unmonitor-instances --instance-ids i-1234567890abcdef0
```

Salida:

```
{
  "InstanceMonitorings": [
    {
      "InstanceId": "i-1234567890abcdef0",
      "Monitoring": {
        "State": "disabling"
      }
    }
  ]
}
```

- Para API obtener más información, consulte [UnmonitorInstances](#) la Referencia de AWS CLI comandos.

## update-security-group-rule-descriptions-egress

En el siguiente ejemplo de código, se muestra cómo usar `update-security-group-rule-descriptions-egress`.

AWS CLI

Para actualizar la descripción de una regla de grupo de seguridad saliente

En el siguiente `update-security-group-rule-descriptions-egress` ejemplo, se actualiza la descripción de la regla del grupo de seguridad para el puerto y el intervalo de IPv4 direcciones especificados. La descripción 'Outbound HTTP access to server 2' reemplaza cualquier descripción existente de la regla.

```
aws ec2 update-security-group-rule-descriptions-egress \  
  --group-id sg-02f0d35a850ba727f \  
  --ip-permissions  
  IpProtocol=tcp,FromPort=80,ToPort=80,IpRanges=[{CidrIp=203.0.113.0/24,Description="Outbound  
  HTTP access to server 2"}]
```

Salida:

```
{
  "Return": true
}
```

Para obtener más información, consulta [las reglas de los grupos de seguridad](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [UpdateSecurityGroupRuleDescriptionsEgress](#) la Referencia de AWS CLI comandos.

## update-security-group-rule-descriptions-ingress

En el siguiente ejemplo de código, se muestra cómo usar `update-security-group-rule-descriptions-ingress`.

### AWS CLI

Ejemplo 1: Para actualizar la descripción de una regla de grupo de seguridad entrante con una fuente CIDR

En el siguiente `update-security-group-rule-descriptions-ingress` ejemplo, se actualiza la descripción de la regla del grupo de seguridad para el puerto y el intervalo de IPv4 direcciones especificados. La descripción 'SSH access from ABC office' reemplaza cualquier descripción existente de la regla.

```
aws ec2 update-security-group-rule-descriptions-ingress \
  --group-id sg-02f0d35a850ba727f \
  --ip-permissions
  IpProtocol=tcp,FromPort=22,ToPort=22,IpRanges='[{"CidrIp=203.0.113.0/16,Description="SSH
  access from corpnet"}]'
```

Salida:

```
{
  "Return": true
}
```

Para obtener más información, consulta [las reglas de los grupos de seguridad](#) en la Guía del EC2 usuario de Amazon.

Ejemplo 2: Para actualizar la descripción de una regla de grupo de seguridad entrante con una fuente de lista de prefijos

En el siguiente `update-security-group-rule-descriptions-ingress` ejemplo, se actualiza la descripción de la regla del grupo de seguridad para la lista de puertos y prefijos especificada. La descripción 'SSH access from ABC office' reemplaza cualquier descripción existente de la regla.

```
aws ec2 update-security-group-rule-descriptions-ingress \  
  --group-id sg-02f0d35a850ba727f \  
  --ip-permissions  
  IpProtocol=tcp,FromPort=22,ToPort=22,PrefixListIds='[{PrefixListId=pl-12345678,Description=  
  access from corpnet}]'
```

Salida:

```
{  
  "Return": true  
}
```

Para obtener más información, consulta [las reglas de los grupos de seguridad](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [UpdateSecurityGroupRuleDescriptionsIngress](#) la Referencia de AWS CLI comandos.

## **withdraw-byoip-cidr**

En el siguiente ejemplo de código, se muestra cómo usar `withdraw-byoip-cidr`.

### AWS CLI

Para dejar de anunciar un rango de direcciones

En el siguiente `withdraw-byoip-cidr` ejemplo, se detiene la publicidad del intervalo de direcciones especificado.

```
aws ec2 withdraw-byoip-cidr  
  --cidr 203.0.113.25/24
```

Salida:

```
{
  "ByoipCidr": {
    "Cidr": "203.0.113.25/24",
    "StatusMessage": "ipv4pool-ec2-1234567890abcdef0",
    "State": "advertised"
  }
}
```

- Para API obtener más información, consulte [WithdrawByoipCidr](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon EC2 Instance Connect que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Amazon EC2 Instance Connect.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

#### **send-ssh-public-key**

En el siguiente ejemplo de código, se muestra cómo usar `send-ssh-public-key`.

#### AWS CLI

Para enviar una clave SSH pública a una instancia

En el siguiente `send-ssh-public-key` ejemplo, se envía la clave SSH pública especificada a la instancia especificada. La clave se usa para autenticar al usuario especificado.

```
aws ec2-instance-connect send-ssh-public-key \
```

```
--instance-id i-1234567890abcdef0 \  
--instance-os-user ec2-user \  
--availability-zone us-east-2b \  
--ssh-public-key file://path/my-rsa-key.pub
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SendSshPublicKey](#) la Referencia de AWS CLI comandos.

## ECREjemplos de Amazon que utilizan AWS CLI

En los siguientes ejemplos de código, se muestra cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con Amazon ECR.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **batch-check-layer-availability**

En el siguiente ejemplo de código, se muestra cómo usar `batch-check-layer-availability`.

AWS CLI

Para comprobar la disponibilidad de una capa

El siguiente `batch-check-layer-availability` ejemplo comprueba la disponibilidad de una capa con el resumen

`sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed`  
en el `cluster-autoscaler` repositorio.

```
aws ecr batch-check-layer-availability \  

```

```
--repository-name cluster-autoscaler \  
--layer-  
digests sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed
```

Salida:

```
{  
  "layers": [  
    {  
      "layerDigest":  
"sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed",  
      "layerAvailability": "AVAILABLE",  
      "layerSize": 2777,  
      "mediaType": "application/vnd.docker.container.image.v1+json"  
    }  
  ],  
  "failures": []  
}
```

- Para API obtener más información, consulte [BatchCheckLayerAvailability](#) la Referencia de AWS CLI comandos.

## batch-delete-image

En el siguiente ejemplo de código, se muestra cómo usar `batch-delete-image`.

AWS CLI

Ejemplo 1: Para eliminar una imagen

En el siguiente `batch-delete-image` ejemplo, se elimina una imagen con la etiqueta `precise` en el repositorio especificado del registro predeterminado de una cuenta.

```
aws ecr batch-delete-image \  
--repository-name ubuntu \  
--image-ids imageTag=precise
```

Salida:

```
{  
  "failures": [],  
  "imageIds": [  
    {  
      "imageId": "precise",  
      "imageIdDigest": "sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed"  
    }  
  ]  
}
```

```

    {
      "imageTag": "precise",
      "imageDigest":
"sha256:19665f1e6d1e504117a1743c0a3d3753086354a38375961f2e665416ef4b1b2f"
    }
  ]
}

```

Ejemplo 2: Para eliminar varias imágenes

En el siguiente `batch-delete-image` ejemplo, se eliminan todas las imágenes etiquetadas con `prod` y `team1` en el repositorio especificado.

```

aws ecr batch-delete-image \
  --repository-name MyRepository \
  --image-ids imageTag=prod imageTag=team1

```

Salida:

```

{
  "imageIds": [
    {
      "imageDigest": "sha256:123456789012",
      "imageTag": "prod"
    },
    {
      "imageDigest": "sha256:567890121234",
      "imageTag": "team1"
    }
  ],
  "failures": []
}

```

Para obtener más información, consulta [Eliminar una imagen](#) en la Guía del ECR usuario de Amazon.

- Para API obtener más información, consulte [BatchDeleteImage](#) la Referencia de AWS CLI comandos.

## **batch-get-image**

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-image`.

## AWS CLI

## Ejemplo 1: Para obtener una imagen

En el siguiente `batch-get-image` ejemplo, se obtiene una imagen con la etiqueta `v1.13.6` en un repositorio llamado `cluster-autoscaler` en el registro predeterminado de una cuenta.

```
aws ecr batch-get-image \
  --repository-name cluster-autoscaler \
  --image-ids imageTag=v1.13.6
```

## Salida:

```
{
  "images": [
    {
      "registryId": "012345678910",
      "repositoryName": "cluster-autoscaler",
      "imageId": {
        "imageDigest":
"sha256:4a1c6567c38904384ebc64e35b7eeddd8451110c299e3368d2210066487d97e5",
        "imageTag": "v1.13.6"
      },
      "imageManifest": "{\n  \"schemaVersion\": 2,\n
  \"mediaType\": \"application/vnd.docker.distribution.manifest.v2+json
\",\n  \"config\": {\n    \"mediaType\": \"application/
vnd.docker.container.image.v1+json\", \"size\": 2777,\n    \"digest
\": \"sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed
\"\n  },\n  \"layers\": [\n    {\n      \"mediaType
\": \"application/vnd.docker.image.rootfs.diff.tar.gzip
\", \"size\": 17743696,\n      \"digest\":
\"sha256:39fafc05754f195f134ca11ecdb1c9a691ab0848c697ffffeb5a85f900caaf6e1\"\n
    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\", \"size\": 2565026,\n
      \"digest\":
\"sha256:8c8a779d3a537b767ae1091fe6e00c2590afd16767aa6096d1b318d75494819f
\"\n    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\", \"size\": 28005981,\n
      \"digest\":
\"sha256:c44ba47496991c9982ee493b47fd25c252caabf2b4ae7dd679c9a27b6a3c8fb7\"\n
    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\", \"size\": 775,\n      \"digest
```



```

\": \"sha256:e2c388b44226544363ca007be7b896bcce1baebea04da23cbd165eac30be650f\"\\n
  }\n  ]\n}"
  }
  ],
  "failures": []
}

```

## Ejemplo 2: Para obtener varias imágenes

El siguiente `batch-get-image` ejemplo muestra los detalles de todas las imágenes etiquetadas con `prod` y `team1` en el repositorio especificado.

```

aws ecr batch-get-image \
  --repository-name MyRepository \
  --image-ids imageTag=prod imageTag=team1

```

## Salida:

```

{
  "images": [
    {
      "registryId": "123456789012",
      "repositoryName": "MyRepository",
      "imageId": {
        "imageDigest": "sha256:123456789012",
        "imageTag": "prod"
      },
      "imageManifest": "manifestExample1"
    },
    {
      "registryId": "567890121234",
      "repositoryName": "MyRepository",
      "imageId": {
        "imageDigest": "sha256:123456789012",
        "imageTag": "team1"
      },
      "imageManifest": "manifestExample2"
    }
  ],
  "failures": []
}

```

Para obtener más información, consulta [Imágenes](#) en la Guía del ECR usuario de Amazon.

- Para API obtener más información, consulte [BatchGetImage](#) la Referencia de AWS CLI comandos.

## complete-layer-upload

En el siguiente ejemplo de código, se muestra cómo usar `complete-layer-upload`.

### AWS CLI

Para completar la carga de una capa de imágenes

En el siguiente `complete-layer-upload` ejemplo, se completa la carga de una capa de imágenes en el `layer-test` repositorio.

```
aws ecr complete-layer-upload \  
  --repository-name layer-test \  
  --upload-id 6cb64b8a-9378-0e33-2ab1-b780fab8a9e9 \  
  --layer-digests 6cb64b8a-9378-0e33-2ab1-  
b780fab8a9e9:48074e6d3a68b39aad8ccc002cdad912d4148c0f92b3729323e
```

Salida:

```
{  
  "uploadId": "6cb64b8a-9378-0e33-2ab1-b780fab8a9e9",  
  "layerDigest":  
    "sha256:9a77f85878aa1906f2020a0ecdf7a7e962d57e882250acd773383224b3fe9a02",  
  "repositoryName": "layer-test",  
  "registryId": "130757420319"  
}
```

- Para API obtener más información, consulte [CompleteLayerUpload](#) la Referencia de AWS CLI comandos.

## create-repository

En el siguiente ejemplo de código, se muestra cómo usar `create-repository`.

### AWS CLI

Ejemplo 1: Para crear un repositorio

El siguiente `create-repository` ejemplo crea un repositorio dentro del espacio de nombres especificado en el registro predeterminado de una cuenta.

```
aws ecr create-repository \  
  --repository-name project-a/nginx-web-app
```

Salida:

```
{  
  "repository": {  
    "registryId": "123456789012",  
    "repositoryName": "sample-repo",  
    "repositoryArn": "arn:aws:ecr:us-west-2:123456789012:repository/project-a/  
nginx-web-app"  
  }  
}
```

Para obtener más información, consulta [Cómo crear un repositorio](#) en la Guía del ECR usuario de Amazon.

Ejemplo 2: Para crear un repositorio configurado con la inmutabilidad de las etiquetas de imagen

El siguiente `create-repository` ejemplo crea un repositorio configurado para la inmutabilidad de las etiquetas en el registro predeterminado de una cuenta.

```
aws ecr create-repository \  
  --repository-name sample-repo \  
  --image-tag-mutability IMMUTABLE
```

Salida:

```
{  
  "repository": {  
    "registryId": "123456789012",  
    "repositoryName": "sample-repo",  
    "repositoryArn": "arn:aws:ecr:us-west-2:123456789012:repository/sample-  
repo",  
    "imageTagMutability": "IMMUTABLE"  
  }  
}
```

Para obtener más información, consulta [la mutabilidad de etiquetas de imagen](#) en la Guía del ECR usuario de Amazon.

Ejemplo 3: Para crear un repositorio configurado con una configuración de digitalización

El siguiente `create-repository` ejemplo crea un repositorio configurado para realizar un análisis de vulnerabilidades al insertar una imagen en el registro predeterminado de una cuenta.

```
aws ecr create-repository \  
  --repository-name sample-repo \  
  --image-scanning-configuration scanOnPush=true
```

Salida:

```
{  
  "repository": {  
    "registryId": "123456789012",  
    "repositoryName": "sample-repo",  
    "repositoryArn": "arn:aws:ecr:us-west-2:123456789012:repository/sample-repo",  
    "imageScanningConfiguration": {  
      "scanOnPush": true  
    }  
  }  
}
```

Para obtener más información, consulta [Escaneo de imágenes](#) en la Guía del ECR usuario de Amazon.

- Para API obtener más información, consulte [CreateRepository](#) la Referencia de AWS CLI comandos.

## **delete-lifecycle-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-lifecycle-policy`.

### AWS CLI

Para eliminar la política de ciclo de vida de un repositorio

En el siguiente `delete-lifecycle-policy` ejemplo, se elimina la política de ciclo de vida del `hello-world` repositorio.

```
aws ecr delete-lifecycle-policy \
  --repository-name hello-world
```

Salida:

```
{
  "registryId": "012345678910",
  "repositoryName": "hello-world",
  "lifecyclePolicyText": "{\n  \"rules\": [\n    {\n      \"rulePriority\": 1,\n      \"description\": \"Remove untagged images.\",\n      \"selection\": {\n        \"tagStatus\": \"untagged\",\n        \"countType\": \"sinceImagePushed\",\n        \"countUnit\": \"days\",\n        \"countNumber\": 10\n      },\n      \"action\": {\n        \"type\": \"expire\"\n      }\n    }\n  ]\n}",
  "lastEvaluatedAt": 0.0
}
```

- Para API obtener más información, consulte [DeleteLifecyclePolicy](#) la Referencia de AWS CLI comandos.

## delete-repository-policy

En el siguiente ejemplo de código, se muestra cómo usar `delete-repository-policy`.

### AWS CLI

Para eliminar la política de repositorio de un repositorio

En el siguiente `delete-repository-policy` ejemplo, se elimina la política de repositorio del `cluster-autoscaler` repositorio.

```
aws ecr delete-repository-policy \
  --repository-name cluster-autoscaler
```

Salida:

```
{
  "registryId": "012345678910",
  "repositoryName": "cluster-autoscaler",
  "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n    \"Sid\" : \"allow public pull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" : \"*\",\n    \"Action\" : [ \"ecr:BatchCheckLayerAvailability\", \"ecr:BatchGetImage\", \"ecr:GetDownloadUrlForLayer\" ]\n  } ]\n}"
```

```
}
```

- Para API obtener más información, consulte [DeleteRepositoryPolicy](#) la Referencia de AWS CLI comandos.

## delete-repository

En el siguiente ejemplo de código, se muestra cómo usar `delete-repository`.

### AWS CLI

Para eliminar un repositorio

En el siguiente `delete-repository` ejemplo, Command Force elimina el repositorio especificado del registro predeterminado de una cuenta. La `--force` marca es obligatoria si el repositorio contiene imágenes.

```
aws ecr delete-repository \  
  --repository-name ubuntu \  
  --force
```

Salida:

```
{  
  "repository": {  
    "registryId": "123456789012",  
    "repositoryName": "ubuntu",  
    "repositoryArn": "arn:aws:ecr:us-west-2:123456789012:repository/ubuntu"  
  }  
}
```

Para obtener más información, consulta [Eliminar un repositorio](#) en la Guía del ECR usuario de Amazon.

- Para API obtener más información, consulte [DeleteRepository](#) la Referencia de AWS CLI comandos.

## describe-image-scan-findings

En el siguiente ejemplo de código, se muestra cómo usar `describe-image-scan-findings`.

## AWS CLI

Para describir los resultados del escaneo de una imagen

El siguiente `describe-image-scan-findings` ejemplo devuelve los resultados del escaneo de una imagen utilizando el resumen de imágenes del repositorio especificado en el registro predeterminado de una cuenta.

```
aws ecr describe-image-scan-findings \  
  --repository-name sample-repo \  
  --image-  
id imageDigest=sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6
```

Salida:

```
{  
  "imageScanFindings": {  
    "findings": [  
      {  
        "name": "CVE-2019-5188",  
        "description": "A code execution vulnerability exists in the directory  
rehashing functionality of E2fsprogs e2fsck 1.45.4. A specially crafted ext4  
directory can cause an out-of-bounds write on the stack, resulting in code  
execution. An attacker can corrupt a partition to trigger this vulnerability.",  
        "uri": "http://people.ubuntu.com/~ubuntu-security/cve/CVE-2019-5188",  
        "severity": "MEDIUM",  
        "attributes": [  
          {  
            "key": "package_version",  
            "value": "1.44.1-1ubuntu1.1"  
          },  
          {  
            "key": "package_name",  
            "value": "e2fsprogs"  
          },  
          {  
            "key": "CVSS2_VECTOR",  
            "value": "AV:L/AC:L/Au:N/C:P/I:P/A:P"  
          },  
          {  
            "key": "CVSS2_SCORE",  
            "value": "4.6"  
          }  
        ]  
      }  
    ]  
  }  
}
```

```

    ]
  }
],
"imageScanCompletedAt": 1579839105.0,
"vulnerabilitySourceUpdatedAt": 1579811117.0,
"findingSeverityCounts": {
  "MEDIUM": 1
}
},
"registryId": "123456789012",
"repositoryName": "sample-repo",
"imageId": {
  "imageDigest":
"sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6"
},
"imageScanStatus": {
  "status": "COMPLETE",
  "description": "The scan was completed successfully."
}
}
}

```

Para obtener más información, consulta [Escaneo de imágenes](#) en la Guía del ECR usuario de Amazon.

- Para API obtener más información, consulte [DescribeImageScanFindings](#) la Referencia de AWS CLI comandos.

## describe-images

En el siguiente ejemplo de código, se muestra cómo usar `describe-images`.

### AWS CLI

Para describir una imagen de un repositorio

En el siguiente `describe-images` ejemplo, se muestran detalles sobre una imagen del `cluster-autoscaler` repositorio con la etiqueta `v1.13.6`.

```

aws ecr describe-images \
  --repository-name cluster-autoscaler \
  --image-ids imageTag=v1.13.6

```

Salida:



```
{
  "imageDetails": [
    {
      "registryId": "012345678910",
      "repositoryName": "cluster-autoscaler",
      "imageDigest":
"sha256:4a1c6567c38904384ebc64e35b7eeddd8451110c299e3368d2210066487d97e5",
      "imageTags": [
        "v1.13.6"
      ],
      "imageSizeInBytes": 48318255,
      "imagePushedAt": 1565128275.0
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeImages](#) la Referencia de AWS CLI comandos.

## describe-repositories

En el siguiente ejemplo de código, se muestra cómo usar `describe-repositories`.

### AWS CLI

Descripción de los repositorios de imágenes en un registro

En este ejemplo, se describen los repositorios del registro predeterminado de una cuenta.

Comando:

```
aws ecr describe-repositories
```

Salida:

```
{
  "repositories": [
    {
      "registryId": "012345678910",
      "repositoryName": "ubuntu",
      "repositoryArn": "arn:aws:ecr:us-west-2:012345678910:repository/ubuntu"
    }
  ]
}
```

```
    },
    {
      "registryId": "012345678910",
      "repositoryName": "test",
      "repositoryArn": "arn:aws:ecr:us-west-2:012345678910:repository/test"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeRepositories](#) la Referencia de AWS CLI comandos.

## get-authorization-token

En el siguiente ejemplo de código, se muestra cómo usar `get-authorization-token`.

### AWS CLI

Para obtener un token de autorización para su registro predeterminado

El siguiente comando de `get-authorization-token` ejemplo obtiene un token de autorización para el registro predeterminado.

```
aws ecr get-authorization-token
```

Salida:

```
{
  "authorizationData": [
    {
      "authorizationToken": "QVdT0kN...",
      "expiresAt": 1448875853.241,
      "proxyEndpoint": "https://123456789012.dkr.ecr.us-west-2.amazonaws.com"
    }
  ]
}
```

- Para API obtener más información, consulte [GetAuthorizationToken](#) la Referencia de AWS CLI comandos.

## get-download-url-for-layer

En el siguiente ejemplo de código, se muestra cómo usar `get-download-url-for-layer`.

### AWS CLI

Para descargar URL una capa

En el siguiente `get-download-url-for-layer` ejemplo, se muestra la descarga URL de una capa con el resumen

`sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed` en el `cluster-autoscaler` repositorio.

```
aws ecr get-download-url-for-layer \
  --repository-name cluster-autoscaler \
  --layer-
digest sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed
```

Salida:

```
{
  "downloadUrl": "https://prod-us-west-2-starport-layer-bucket.s3.us-
west-2.amazonaws.com/e501-012345678910-9cb60dc0-7284-5643-3987-
da6dac0465f0/04620aac-66a5-4167-8232-55ee7ef6d565?X-Amz-Algorithm=AWS4-HMAC-
SHA256&X-Amz-Date=20190814T220617Z&X-Amz-SignedHeaders=host&X-Amz-Expires=3600&X-
Amz-Credential=AKIA32P3D2JDNMVAJLGF%2F20190814%2Fus-west-2%2Fs3%2Faws4_request&X-
Amz-Signature=9161345894947a1672467a0da7a1550f2f7157318312fe4941b59976239c3337",
  "layerDigest":
  "sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed"
}
```

- Para API obtener más información, consulte [GetDownloadUriForLayer](#) la Referencia de AWS CLI comandos.

## get-lifecycle-policy-preview

En el siguiente ejemplo de código, se muestra cómo usar `get-lifecycle-policy-preview`.

### AWS CLI

Para recuperar los detalles de una vista previa de la política del ciclo de vida

En el siguiente `get-lifecycle-policy-preview` ejemplo, se recupera el resultado de una vista previa de la política de ciclo de vida del repositorio especificado en el registro predeterminado de una cuenta.

Comando:

```
aws ecr get-lifecycle-policy-preview \
  --repository-name "project-a/amazon-ecs-sample"
```

Salida:

```
{
  "registryId": "012345678910",
  "repositoryName": "project-a/amazon-ecs-sample",
  "lifecyclePolicyText": "{\n  \"rules\": [\n    {\n      \"rulePriority\": 1,\n      \"description\": \"Expire images older than 14 days\",\n      \"selection\": {\n        \"tagStatus\": \"untagged\",\n        \"countType\": \"sinceImagePushed\",\n        \"countUnit\": \"days\",\n        \"countNumber\": 14\n      },\n      \"action\": {\n        \"type\": \"expire\"\n      }\n    }\n  ]\n}\n",
  "status": "COMPLETE",
  "previewResults": [],
  "summary": {
    "expiringImageTotalCount": 0
  }
}
```

Para obtener más información, consulta [las políticas de ciclo de vida](#) en la Guía del ECR usuario de Amazon.

- Para API obtener más información, consulte [GetLifecyclePolicyPreview](#) la Referencia de AWS CLI comandos.

## get-lifecycle-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-lifecycle-policy`.

### AWS CLI

Para recuperar una política de ciclo de vida

En el siguiente `get-lifecycle-policy` ejemplo, se muestran los detalles de la política de ciclo de vida del repositorio especificado en el registro predeterminado de la cuenta.

```
aws ecr get-lifecycle-policy \  
  --repository-name "project-a/amazon-ecs-sample"
```

Salida:

```
{  
  "registryId": "123456789012",  
  "repositoryName": "project-a/amazon-ecs-sample",  
  "lifecyclePolicyText": "{\"rules\": [{\"rulePriority\": 1, \"description\":  
  \"Expire images older than 14 days\", \"selection\": {\"tagStatus\": \"untagged\",  
  \"countType\": \"sinceImagePushed\", \"countUnit\": \"days\", \"countNumber\": 14},  
  \"action\": {\"type\": \"expire\"}}]}",  
  "lastEvaluatedAt": 1504295007.0  
}
```

Para obtener más información, consulta [las políticas de ciclo de vida](#) en la Guía del ECR usuario de Amazon.

- Para API obtener más información, consulte [GetLifecyclePolicy](#) la Referencia de AWS CLI comandos.

## get-login-password

En el siguiente ejemplo de código, se muestra cómo usar `get-login-password`.

AWS CLI

Para recuperar una contraseña para autenticarse en un registro

A continuación, se `get-login-password` muestra una contraseña que puedes usar con un cliente contenedor de tu elección para autenticarte en cualquier ECR registro de Amazon al que tenga acceso tu IAM principal.

```
aws ecr get-login-password
```

Salida:

```
<password>
```

Para usarlo con el DockerCLI, canaliza el resultado del `get-login-password` comando al `docker login` comando. Al recuperar la contraseña, asegúrate de especificar la misma región en la que se encuentra tu ECR registro de Amazon.

```
aws ecr get-login-password \  
  --region <region> \  
 \  
| docker login \  
  --username AWS \  
  --password-stdin <aws_account_id>.dkr.ecr.<region>.amazonaws.com
```

Para obtener más información, consulte [Autenticación del registro](#) en la Guía del ECR usuario de Amazon.

- Para API obtener más información, consulte [GetLoginPassword](#) la Referencia de AWS CLI comandos.

## get-login

En el siguiente ejemplo de código, se muestra cómo usar `get-login`.

### AWS CLI

Para recuperar un comando de inicio de sesión de Docker en su registro predeterminado

En este ejemplo, se imprime un comando que puedes usar para iniciar sesión en tu ECR registro predeterminado de Amazon.

Comando:

```
aws ecr get-login
```

Salida:

```
docker login -u AWS -p <password> -e none https://  
<aws_account_id>.dkr.ecr.<region>.amazonaws.com
```

Para iniciar sesión en el registro de otra cuenta

En este ejemplo, se imprimen uno o varios comandos que puedes usar para iniciar sesión en ECR los registros de Amazon asociados a otras cuentas.

Comando:

```
aws ecr get-login --registry-ids 012345678910 023456789012
```

Salida:

```
docker login -u <username> -p <token-1> -e none <endpoint-1>
docker login -u <username> -p <token-2> -e none <endpoint-2>
```

- Para API obtener más información, consulte [GetLogin](#) la Referencia de AWS CLI comandos.

## get-repository-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-repository-policy`.

AWS CLI

Para recuperar la política de repositorios de un repositorio

En el siguiente `get-repository-policy` ejemplo, se muestran detalles sobre la política de `cluster-autoscaler` repositorio del repositorio.

```
aws ecr get-repository-policy \
  --repository-name cluster-autoscaler
```

Salida:

```
{
  "registryId": "012345678910",
  "repositoryName": "cluster-autoscaler",
  "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n    \"Sid\" : \"allow public pull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" : \"*\",\n    \"Action\" : [ \"ecr:BatchCheckLayerAvailability\", \"ecr:BatchGetImage\", \"ecr:GetDownloadUrlForLayer\" ]\n  } ]\n}"
```

- Para API obtener más información, consulte [GetRepositoryPolicy](#) la Referencia de AWS CLI comandos.

## initiate-layer-upload

En el siguiente ejemplo de código, se muestra cómo usar `initiate-layer-upload`.

### AWS CLI

Para iniciar la carga de una capa de imágenes

El siguiente `initiate-layer-upload` ejemplo inicia la carga de una capa de imágenes en el `layer-test` repositorio.

```
aws ecr initiate-layer-upload \  
  --repository-name layer-test
```

Salida:

```
{  
  "partSize": 10485760,  
  "uploadId": "6cb64b8a-9378-0e33-2ab1-b780fab8a9e9"  
}
```

- Para API obtener más información, consulte [InitiateLayerUpload](#) la Referencia de AWS CLI comandos.

## list-images

En el siguiente ejemplo de código, se muestra cómo usar `list-images`.

### AWS CLI

Creación de una lista de las imágenes en un repositorio

En el siguiente ejemplo de `list-images` se muestra una lista de las imágenes del repositorio `cluster-autoscaler`.

```
aws ecr list-images \  
  --repository-name cluster-autoscaler
```

Salida:

```
{  
  "imageIds": [  
    "
```



```

    {
      "imageDigest":
"sha256:99c6fb4377e9a420a1eb3b410a951c9f464eff3b7dbc76c65e434e39b94b6570",
      "imageTag": "v1.13.8"
    },
    {
      "imageDigest":
"sha256:99c6fb4377e9a420a1eb3b410a951c9f464eff3b7dbc76c65e434e39b94b6570",
      "imageTag": "v1.13.7"
    },
    {
      "imageDigest":
"sha256:4a1c6567c38904384ebc64e35b7eeddd8451110c299e3368d2210066487d97e5",
      "imageTag": "v1.13.6"
    }
  ]
}

```

- Para API obtener más información, consulte [ListImages](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para ver una lista de las etiquetas del repositorio

En el siguiente `list-tags-for-resource` ejemplo, se muestra una lista de las etiquetas asociadas al `hello-world` repositorio.

```

aws ecr list-tags-for-resource \
  --resource-arn arn:aws:ecr:us-west-2:012345678910:repository/hello-world

```

Salida:

```

{
  "tags": [
    {
      "Key": "Stage",
      "Value": "Integ"
    }
  ]
}

```

```
]
}
```

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## put-image-scanning-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-image-scanning-configuration`.

### AWS CLI

Para actualizar la configuración de digitalización de imágenes de un repositorio

El siguiente `put-image-scanning-configuration` ejemplo actualiza la configuración de digitalización de imágenes del repositorio especificado.

```
aws ecr put-image-scanning-configuration \
  --repository-name sample-repo \
  --image-scanning-configuration scanOnPush=true
```

Salida:

```
{
  "registryId": "012345678910",
  "repositoryName": "sample-repo",
  "imageScanningConfiguration": {
    "scanOnPush": true
  }
}
```

Para obtener más información, consulta [Escaneo de imágenes](#) en la Guía del ECR usuario de Amazon.

- Para API obtener más información, consulte [PutImageScanningConfiguration](#) la Referencia de AWS CLI comandos.

## put-image-tag-mutability

En el siguiente ejemplo de código, se muestra cómo usar `put-image-tag-mutability`.

## AWS CLI

Para actualizar la configuración de mutabilidad de las etiquetas de imagen de un repositorio

El siguiente `put-image-tag-mutability` ejemplo configura el repositorio especificado para la inmutabilidad de las etiquetas. Esto evita que se sobrescriban todas las etiquetas de imagen del repositorio.

```
aws ecr put-image-tag-mutability \  
  --repository-name hello-repository \  
  --image-tag-mutability IMMUTABLE
```

Salida:

```
{  
  "registryId": "012345678910",  
  "repositoryName": "sample-repo",  
  "imageTagMutability": "IMMUTABLE"  
}
```

Para obtener más información, consulta [la mutabilidad de etiquetas de imagen](#) en la Guía del ECR usuario de Amazon.

- Para API obtener más información, consulte [PutImageTagMutability](#) la Referencia de AWS CLI comandos.

## put-image

En el siguiente ejemplo de código, se muestra cómo usar `put-image`.

## AWS CLI

Para volver a etiquetar una imagen con su manifiesto

En el siguiente `put-image` ejemplo, se crea una etiqueta nueva en el `hello-world` repositorio con un manifiesto de imagen existente.

```
aws ecr put-image \  
  --repository-name hello-world \  
  --image-tag 2019.08 \  
  --image-manifest file://hello-world.manifest.json
```

Contenidos de `hello-world.manifest.json`:

```
{
  "schemaVersion": 2,
  "mediaType": "application/vnd.docker.distribution.manifest.v2+json",
  "config": {
    "mediaType": "application/vnd.docker.container.image.v1+json",
    "size": 5695,
    "digest":
"sha256:cea5fe7701b7db3dd1c372f3cea6f43cdda444fcc488f530829145e426d8b980"
  },
  "layers": [
    {
      "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
      "size": 39096921,
      "digest":
"sha256:d8868e50ac4c7104d2200d42f432b661b2da8c1e417ccfae217e6a1e04bb9295"
    },
    {
      "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
      "size": 57938,
      "digest":
"sha256:83251ac64627fc331584f6c498b3aba5badc01574e2c70b2499af3af16630eed"
    },
    {
      "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
      "size": 423,
      "digest":
"sha256:589bba2f1b36ae56f0152c246e2541c5aa604b058febfcf2be32e9a304fec610"
    },
    {
      "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
      "size": 680,
      "digest":
"sha256:d62ecaceda3964b735cdd2af613d6bb136a52c1da0838b2ff4b4dab4212bcb1c"
    },
    {
      "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
      "size": 162,
      "digest":
"sha256:6d93b41cfc6bf0d2522b7cf61588de4cd045065b36c52bd3aec2ba0622b2b22b"
    },
    {
      "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
```

```

        "size": 28268840,
        "digest":
"sha256:6986b4d4c07932c680b3587f2eac8b0e013568c003cc23b04044628a5c5e599f"
    },
    {
        "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
        "size": 35369152,
        "digest":
"sha256:8c5ec60f10102dc8da0649d866c7c2f706e459d0bdc25c83ad2de86f4996c276"
    },
    {
        "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
        "size": 155,
        "digest":
"sha256:cde50b1c594539c5f67cbede9aef95c9ae321ccfb857f7b251b45b84198adc85"
    },
    {
        "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
        "size": 28737,
        "digest":
"sha256:2e102807ab72a73fc9abf53e8c50e421bdc337a0a8afcb242176edeec65977e4"
    },
    {
        "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
        "size": 190,
        "digest":
"sha256:fc379bbd5ed37808772bef016553a297356c59b8f134659e6ee4ecb563c2f5a7"
    },
    {
        "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
        "size": 28748,
        "digest":
"sha256:021db240dfccf5a1aff19507d17c0177e5888e518acf295b52204b1825e8b7ee"
    }
]
}

```

Salida:

```

{
  "image": {
    "registryId": "130757420319",
    "repositoryName": "hello-world",

```

```

    "imageId": {
      "imageDigest":
"sha256:8ece96b74f87652876199d83bd107d0435a196133af383ac54cb82b6cc5283ae",
      "imageTag": "2019.08"
    },
    "imageManifest": "{\n  \"schemaVersion\": 2,\n  \"mediaType
\": \"application/vnd.docker.distribution.manifest.v2+json
\",,\n  \"config\": {\n    \"mediaType\": \"application/
vnd.docker.container.image.v1+json\",,\n    \"size\": 5695,\n    \"digest\":
\"sha256:cea5fe7701b7db3dd1c372f3cea6f43cdda444fcc488f530829145e426d8b980\"
\",,\n    \"layers\": [\n      {\n        \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n        \"size\": 39096921,\n        \"digest
\": \"sha256:d8868e50ac4c7104d2200d42f432b661b2da8c1e417ccfae217e6a1e04bb9295\"
\",,\n        {\n          \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n          \"size\": 57938,\n          \"digest
\": \"sha256:83251ac64627fc331584f6c498b3aba5badc01574e2c70b2499af3af16630eed
\"
\",,\n        {\n          \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n          \"size\": 423,\n          \"digest\":
\"sha256:589bba2f1b36ae56f0152c246e2541c5aa604b058febfcf2be32e9a304fec610\"
\",,\n        {\n          \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",
\",,\n          \"size\": 680,\n          \"digest\":
\"sha256:d62ecaceda3964b735cdd2af613d6bb136a52c1da0838b2ff4b4dab4212bcb1c
\"
\",,\n        {\n          \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n          \"size\": 162,\n          \"digest
\": \"sha256:6d93b41cfc6bf0d2522b7cf61588de4cd045065b36c52bd3aec2ba0622b2b22b
\"
\",,\n        {\n          \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n          \"size\": 28268840,\n          \"digest
\": \"sha256:6986b4d4c07932c680b3587f2eac8b0e013568c003cc23b04044628a5c5e599f
\"
\",,\n        {\n          \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n          \"size\": 35369152,\n          \"digest
\": \"sha256:8c5ec60f10102dc8da0649d866c7c2f706e459d0bdc25c83ad2de86f4996c276\"
\",,\n        {\n          \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",,\n          \"size\": 155,\n          \"digest\":
\"sha256:cde50b1c594539c5f67cbede9aef95c9ae321ccfb857f7b251b45b84198adc85\"
\",,\n        {\n          \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",
\",,\n          \"size\": 28737,\n          \"digest\":
\"sha256:2e102807ab72a73fc9abf53e8c50e421bdc337a0a8afcb242176edeec65977e4\"
\",,\n        {\n          \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",
\",,\n          \"size\": 190,\n          \"digest\":
\"sha256:fc379bbd5ed37808772bef016553a297356c59b8f134659e6ee4ecb563c2f5a7\"
\",,\n        {\n          \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",
\",,\n          \"size\": 28748,\n          \"digest\":
\"sha256:021db240dfccf5a1aff19507d17c0177e5888e518acf295b52204b1825e8b7ee\"
\",,\n      ]\n    }\n  }"

```

```
}  
}
```

- Para API obtener más información, consulte [PutImage](#) la Referencia de AWS CLI comandos.

## put-lifecycle-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-lifecycle-policy`.

### AWS CLI

Para crear una política de ciclo de vida

En el siguiente `put-lifecycle-policy` ejemplo, se crea una política de ciclo de vida para el repositorio especificado en el registro predeterminado de una cuenta.

```
aws ecr put-lifecycle-policy \  
  --repository-name "project-a/amazon-ecs-sample" \  
  --lifecycle-policy-text "file://policy.json"
```

Contenidos de `policy.json`:

```
{  
  "rules": [  
    {  
      "rulePriority": 1,  
      "description": "Expire images older than 14 days",  
      "selection": {  
        "tagStatus": "untagged",  
        "countType": "sinceImagePushed",  
        "countUnit": "days",  
        "countNumber": 14  
      },  
      "action": {  
        "type": "expire"  
      }  
    }  
  ]  
}
```

Salida:

```
{
  "registryId": "<aws_account_id>",
  "repositoryName": "project-a/amazon-ecs-sample",
  "lifecyclePolicyText": "{\"rules\": [{\"rulePriority\": 1, \"description\": \"Expire images older than 14 days\", \"selection\": {\"tagStatus\": \"untagged\", \"countType\": \"sinceImagePushed\", \"countUnit\": \"days\", \"countNumber\": 14}, \"action\": {\"type\": \"expire\"}}]}"
}
```

Para obtener más información, consulta [las políticas de ciclo](#) de vida en la Guía del ECR usuario de Amazon.

- Para API obtener más información, consulte [PutLifecyclePolicy](#) la Referencia de AWS CLI comandos.

## set-repository-policy

En el siguiente ejemplo de código, se muestra cómo usar `set-repository-policy`.

### AWS CLI

Para establecer la política de repositorio de un repositorio

En el siguiente `set-repository-policy` ejemplo, se adjunta al repositorio una política de `cluster-autoscaler` repositorio incluida en un archivo.

```
aws ecr set-repository-policy \
  --repository-name cluster-autoscaler \
  --policy-text file://my-policy.json
```

Contenidos de `my-policy.json`:

```
{
  "Version" : "2008-10-17",
  "Statement" : [
    {
      "Sid" : "allow public pull",
      "Effect" : "Allow",
      "Principal" : "*",
      "Action" : [
        "ecr:BatchCheckLayerAvailability",
```



```

        "ecr:BatchGetImage",
        "ecr:GetDownloadUrlForLayer"
    ]
}

```

Salida:

```

{
  "registryId": "012345678910",
  "repositoryName": "cluster-autoscaler",
  "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n    \"Sid\" : \"allow public pull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" : \"*\",\n    \"Action\" : [ \"ecr:BatchCheckLayerAvailability\", \"ecr:BatchGetImage\", \"ecr:GetDownloadUrlForLayer\" ]\n  } ]\n}"
}

```

- Para API obtener más información, consulte [SetRepositoryPolicy](#) la Referencia de AWS CLI comandos.

## start-image-scan

En el siguiente ejemplo de código, se muestra cómo usar `start-image-scan`.

### AWS CLI

Para iniciar un análisis de vulnerabilidades de imagen

En el siguiente `start-image-scan` ejemplo, se inicia un escaneo de imágenes en busca de las imágenes especificadas en el resumen de imágenes del repositorio especificado.

```

aws ecr start-image-scan \
  --repository-name sample-repo \
  --image-  

  id imageDigest=sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6

```

Salida:

```

{

```

```
"registryId": "012345678910",
"repositoryName": "sample-repo",
"imageId": {
  "imageDigest":
"sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6"
},
"imageScanStatus": {
  "status": "IN_PROGRESS"
}
}
```

Para obtener más información, consulta [Escaneo de imágenes](#) en la Guía del ECR usuario de Amazon.

- Para API obtener más información, consulte [StartImageScan](#) la Referencia de AWS CLI comandos.

## start-lifecycle-policy-preview

En el siguiente ejemplo de código, se muestra cómo usar `start-lifecycle-policy-preview`.

### AWS CLI

Para crear una vista previa de la política del ciclo de vida

El siguiente `start-lifecycle-policy-preview` ejemplo crea una vista previa de la política del ciclo de vida definida por un JSON archivo para el repositorio especificado.

```
aws ecr start-lifecycle-policy-preview \
  --repository-name "project-a/amazon-ecs-sample" \
  --lifecycle-policy-text "file://policy.json"
```

Contenidos de `policy.json`:

```
{
  "rules": [
    {
      "rulePriority": 1,
      "description": "Expire images older than 14 days",
      "selection": {
        "tagStatus": "untagged",
```

```

        "countType": "sinceImagePushed",
        "countUnit": "days",
        "countNumber": 14
    },
    "action": {
        "type": "expire"
    }
}
]
}

```

Salida:

```

{
  "registryId": "012345678910",
  "repositoryName": "project-a/amazon-ecs-sample",
  "lifecyclePolicyText": "{\n  \"rules\": [\n    {\n\n      \"rulePriority\": 1,\n      \"description\": \"Expire images older than 14\n      days\",\n      \"selection\": {\n        \"tagStatus\": \"untagged\",\n\n        \"countType\": \"sinceImagePushed\",\n        \"countUnit\n\": \"days\",\n        \"countNumber\": 14\n      },\n      \"action\": {\n        \"type\": \"expire\"\n      }\n    }\n  ]\n}\n",
  "status": "IN_PROGRESS"
}

```

- Para API obtener más información, consulte [StartLifecyclePolicyPreview](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

### AWS CLI

Para etiquetar un repositorio

El siguiente tag-resource ejemplo establece una etiqueta con la clave Stage y el valor Integ en el hello-world repositorio.

```
aws ecr tag-resource \
```

```
--resource-arn arn:aws:ecr:us-west-2:012345678910:repository/hello-world \  
--tags Key=Stage,Value=Integ
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para quitar la etiqueta de un repositorio

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta con la clave `Stage` del `hello-world` repositorio.

```
aws ecr untag-resource \  
--resource-arn arn:aws:ecr:us-west-2:012345678910:repository/hello-world \  
--tag-keys Stage
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## upload-layer-part

En el siguiente ejemplo de código, se muestra cómo usar `upload-layer-part`.

### AWS CLI

Para cargar una parte de una capa

A continuación, se `upload-layer-part` carga una parte de la capa de imagen en el `layer-test` repositorio.

```
aws ecr upload-layer-part \  

```

```
--repository-name layer-test \  
--upload-id 6cb64b8a-9378-0e33-2ab1-b780fab8a9e9 \  
--part-first-byte 0 \  
--part-last-byte 8323314 \  
--layer-part-blob file:///var/lib/docker/image/overlay2/layerdb/sha256/  
ff986b10a018b48074e6d3a68b39aad8ccc002cdad912d4148c0f92b3729323e/layer.b64
```

Salida:

```
{  
  "uploadId": "6cb64b8a-9378-0e33-2ab1-b780fab8a9e9",  
  "registryId": "012345678910",  
  "lastByteReceived": 8323314,  
  "repositoryName": "layer-test"  
}
```

- Para API obtener más información, consulte [UploadLayerPart](#) la Referencia de AWS CLI comandos.

## ECSEjemplos de Amazon que utilizan AWS CLI

En los siguientes ejemplos de código, se muestra cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con AmazonECS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-capacity-provider**

En el siguiente ejemplo de código, se muestra cómo usar `create-capacity-provider`.

## AWS CLI

Para crear un proveedor de capacidad

El siguiente create-capacity-provider ejemplo crea un proveedor de capacidad que usa un grupo de Auto Scaling denominado My ASG y que tiene habilitada la protección de escalado administrado y terminación administrada. Esta configuración se usa para el escalado automático de ECS clústeres de Amazon.

```
aws ecs create-capacity-provider \  
  --name "MyCapacityProvider" \  
  --auto-scaling-group-provider "autoScalingGroupArn=arn:aws:autoscaling:us-  
east-1:123456789012:autoScalingGroup:57ffcb94-11f0-4d6d-  
bf60-3bac5EXAMPLE:autoScalingGroupName/  
MyASG,managedScaling={status=ENABLED,targetCapacity=100},managedTerminationProtection=ENABLED"
```

Salida:

```
{  
  "capacityProvider": {  
    "capacityProviderArn": "arn:aws:ecs:us-east-1:123456789012:capacity-provider/  
MyCapacityProvider",  
    "name": "MyCapacityProvider",  
    "status": "ACTIVE",  
    "autoScalingGroupProvider": {  
      "autoScalingGroupArn": "arn:aws:autoscaling:us-  
east-1:132456789012:autoScalingGroup:57ffcb94-11f0-4d6d-  
bf60-3bac5EXAMPLE:autoScalingGroupName/MyASG",  
      "managedScaling": {  
        "status": "ENABLED",  
        "targetCapacity": 100,  
        "minimumScalingStepSize": 1,  
        "maximumScalingStepSize": 10000,  
        "instanceWarmupPeriod": 300  
      },  
      "managedTerminationProtection": "ENABLED"  
    },  
    "tags": []  
  }  
}
```

Para obtener más información, consulta el [escalado automático de ECS clústeres](#) de Amazon en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [CreateCapacityProvider](#) la Referencia de AWS CLI comandos.

## create-cluster

En el siguiente ejemplo de código, se muestra cómo usar `create-cluster`.

### AWS CLI

Ejemplo 1: creación de un clúster nuevo

En el siguiente ejemplo de `create-cluster` se crea un clúster.

```
aws ecs create-cluster \  
  --cluster-name MyCluster
```

Salida:

```
{  
  "cluster": {  
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",  
    "clusterName": "MyCluster",  
    "status": "ACTIVE",  
    "registeredContainerInstancesCount": 0,  
    "pendingTasksCount": 0,  
    "runningTasksCount": 0,  
    "activeServicesCount": 0,  
    "statistics": [],  
    "tags": []  
  }  
}
```

Para obtener más información, consulte [Creación de un clúster](#) en la Guía para ECS desarrolladores de Amazon.

Ejemplo 2: Creación de un clúster nuevo con proveedores de capacidad

En el siguiente ejemplo de `create-cluster` se crea un clúster y se le asocian dos proveedores de capacidad existentes. El comando `create-capacity-provider` se utiliza para crear un proveedor de capacidad. La especificación de una estrategia predeterminada de proveedor de capacidad es opcional, pero se recomienda. En este ejemplo, creamos un clúster denominado

MyCluster y le asociamos los proveedores de capacidad MyCapacityProvider1 y MyCapacityProvider2. Se especifica una estrategia predeterminada de proveedor de capacidad que distribuye las tareas de manera uniforme entre ambos proveedores de capacidad.

```
aws ecs create-cluster --cluster-name MyCluster --capacity providers 1 2 --default-capacity-provider-strategy capacityProvider = 1, MyCapacityProvider peso=1 = MyCapacityProvider 2, peso=1 MyCapacityProvider capacityProvider MyCapacityProvider
```

Salida:

```
{
  "cluster": {
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "clusterName": "MyCluster",
    "status": "PROVISIONING",
    "registeredContainerInstancesCount": 0,
    "pendingTasksCount": 0,
    "runningTasksCount": 0,
    "activeServicesCount": 0,
    "statistics": [],
    "settings": [
      {
        "name": "containerInsights",
        "value": "enabled"
      }
    ],
    "capacityProviders": [
      "MyCapacityProvider1",
      "MyCapacityProvider2"
    ],
    "defaultCapacityProviderStrategy": [
      {
        "capacityProvider": "MyCapacityProvider1",
        "weight": 1,
        "base": 0
      },
      {
        "capacityProvider": "MyCapacityProvider2",
        "weight": 1,
        "base": 0
      }
    ],
    "attachments": [
```



```

    {
      "id": "0fb0c8f4-6edd-4de1-9b09-17e470ee1918",
      "type": "asp",
      "status": "PRECREATED",
      "details": [
        {
          "name": "capacityProviderName",
          "value": "MyCapacityProvider1"
        },
        {
          "name": "scalingPlanName",
          "value": "ECSManagedAutoScalingPlan-a1b2c3d4-5678-90ab-cdef-
EXAMPLE111111"
        }
      ]
    },
    {
      "id": "ae592060-2382-4663-9476-b015c685593c",
      "type": "asp",
      "status": "PRECREATED",
      "details": [
        {
          "name": "capacityProviderName",
          "value": "MyCapacityProvider2"
        },
        {
          "name": "scalingPlanName",
          "value": "ECSManagedAutoScalingPlan-a1b2c3d4-5678-90ab-cdef-
EXAMPLE222222"
        }
      ]
    }
  ],
  "attachmentsStatus": "UPDATE_IN_PROGRESS"
}

```

Para obtener más información, consulte [Proveedores de capacidad de clúster](#) en la Guía para ECS desarrolladores de Amazon.

Ejemplo 3: Creación de un clúster nuevo con varias etiquetas

En el siguiente ejemplo de `create-cluster` se crea un clúster con varias etiquetas. Para obtener más información sobre cómo añadir etiquetas mediante la sintaxis abreviada, consulte [Uso de la sintaxis abreviada con la interfaz de línea de AWS comandos en la Guía del usuario.AWS CLI](#)

```
aws ecs create-cluster \  
  --cluster-name MyCluster \  
  --tags key=key1,value=value1 key=key2,value=value2 key=key3,value=value3
```

Salida:

```
{  
  "cluster": {  
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",  
    "clusterName": "MyCluster",  
    "status": "ACTIVE",  
    "registeredContainerInstancesCount": 0,  
    "pendingTasksCount": 0,  
    "runningTasksCount": 0,  
    "activeServicesCount": 0,  
    "statistics": [],  
    "tags": [  
      {  
        "key": "key1",  
        "value": "value1"  
      },  
      {  
        "key": "key2",  
        "value": "value2"  
      },  
      {  
        "key": "key3",  
        "value": "value3"  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [Creación de un clúster](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [CreateCluster](#) la Referencia de AWS CLI comandos.

## create-service

En el siguiente ejemplo de código, se muestra cómo usar `create-service`.

### AWS CLI

Ejemplo 1: creación de un servicio con una tarea de Fargate

En el siguiente ejemplo de `create-service` se muestra cómo crear un servicio mediante una tarea de Fargate.

```
aws ecs create-service \
  --cluster MyCluster \
  --service-name MyService \
  --task-definition sample-fargate:1 \
  --desired-count 2 \
  --launch-type FARGATE \
  --platform-version LATEST \
  --network-
configuration "awsvpcConfiguration={subnets=[subnet-12344321],securityGroups=[sg-12344321],a
  \
  --tags key=key1,value=value1 key=key2,value=value2 key=key3,value=value3
```

Salida:

```
{
  "service": {
    "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/MyService",
    "serviceName": "MyService",
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "loadBalancers": [],
    "serviceRegistries": [],
    "status": "ACTIVE",
    "desiredCount": 2,
    "runningCount": 0,
    "pendingCount": 0,
    "launchType": "FARGATE",
    "platformVersion": "LATEST",
```

```
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
sample-fargate:1",
    "deploymentConfiguration": {
        "maximumPercent": 200,
        "minimumHealthyPercent": 100
    },
    "deployments": [
        {
            "id": "ecs-svc/1234567890123456789",
            "status": "PRIMARY",
            "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/sample-fargate:1",
            "desiredCount": 2,
            "pendingCount": 0,
            "runningCount": 0,
            "createdAt": 1557119253.821,
            "updatedAt": 1557119253.821,
            "launchType": "FARGATE",
            "platformVersion": "1.3.0",
            "networkConfiguration": {
                "awsvpcConfiguration": {
                    "subnets": [
                        "subnet-12344321"
                    ],
                    "securityGroups": [
                        "sg-12344321"
                    ],
                    "assignPublicIp": "ENABLED"
                }
            }
        }
    ],
    "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
ecs.amazonaws.com/AWSServiceRoleForECS",
    "events": [],
    "createdAt": 1557119253.821,
    "placementConstraints": [],
    "placementStrategy": [],
    "networkConfiguration": {
        "awsvpcConfiguration": {
            "subnets": [
                "subnet-12344321"
            ],
            "securityGroups": [
```

```

        "sg-12344321"
      ],
      "assignPublicIp": "ENABLED"
    }
  ],
  "schedulingStrategy": "REPLICA",
  "tags": [
    {
      "key": "key1",
      "value": "value1"
    },
    {
      "key": "key2",
      "value": "value2"
    },
    {
      "key": "key3",
      "value": "value3"
    }
  ],
  "enableECSTags": false,
  "propagateTags": "NONE"
}
}

```

Ejemplo 2: Para crear un servicio mediante el tipo de EC2 lanzamiento

El siguiente `create-service` ejemplo muestra cómo crear un servicio llamado `ecs-simple-service` con una tarea que utilice el tipo de EC2 lanzamiento. El servicio utiliza la definición de la tarea `sleep360` y mantiene 1 instancia de la tarea.

```

aws ecs create-service \
  --cluster MyCluster \
  --service-name ecs-simple-service \
  --task-definition sleep360:2 \
  --desired-count 1

```

Salida:

```

{
  "service": {
    "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/ecs-
simple-service",

```

```

    "serviceName": "ecs-simple-service",
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "loadBalancers": [],
    "serviceRegistries": [],
    "status": "ACTIVE",
    "desiredCount": 1,
    "runningCount": 0,
    "pendingCount": 0,
    "launchType": "EC2",
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
sleep360:2",
    "deploymentConfiguration": {
        "maximumPercent": 200,
        "minimumHealthyPercent": 100
    },
    "deployments": [
        {
            "id": "ecs-svc/1234567890123456789",
            "status": "PRIMARY",
            "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/sleep360:2",
            "desiredCount": 1,
            "pendingCount": 0,
            "runningCount": 0,
            "createdAt": 1557206498.798,
            "updatedAt": 1557206498.798,
            "launchType": "EC2"
        }
    ],
    "events": [],
    "createdAt": 1557206498.798,
    "placementConstraints": [],
    "placementStrategy": [],
    "schedulingStrategy": "REPLICA",
    "enableECSTags": false,
    "propagateTags": "NONE"
}
}

```

### Ejemplo 3: Creación de un servicio que utilice un controlador de implementación externo

En el siguiente ejemplo de `create-service` se crea un servicio que utiliza un controlador de implementación externo.

```
aws ecs create-service \  
  --cluster MyCluster \  
  --service-name MyService \  
  --deployment-controller type=EXTERNAL \  
  --desired-count 1
```

Salida:

```
{  
  "service": {  
    "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/  
MyService",  
    "serviceName": "MyService",  
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",  
    "loadBalancers": [],  
    "serviceRegistries": [],  
    "status": "ACTIVE",  
    "desiredCount": 1,  
    "runningCount": 0,  
    "pendingCount": 0,  
    "launchType": "EC2",  
    "deploymentConfiguration": {  
      "maximumPercent": 200,  
      "minimumHealthyPercent": 100  
    },  
    "taskSets": [],  
    "deployments": [],  
    "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/  
ecs.amazonaws.com/AWSServiceRoleForECS",  
    "events": [],  
    "createdAt": 1557128207.101,  
    "placementConstraints": [],  
    "placementStrategy": [],  
    "schedulingStrategy": "REPLICA",  
    "deploymentController": {  
      "type": "EXTERNAL"  
    },  
    "enableECSManagedTags": false,  
    "propagateTags": "NONE"  
  }  
}
```

#### Ejemplo 4: Creación de un nuevo servicio detrás de un equilibrador de carga

En el siguiente ejemplo de `create-service` se muestra cómo crear un servicio que se encuentra detrás de un equilibrador de carga. Debe tener un equilibrador de carga configurado en la misma región que la instancia de contenedor. En este ejemplo, se utiliza la `--cli-input-json` opción y un archivo de JSON entrada llamado `ecs-simple-service-elb.json` con el siguiente contenido:

```
{
  "serviceName": "ecs-simple-service-elb",
  "taskDefinition": "ecs-demo",
  "loadBalancers": [
    {
      "loadBalancerName": "EC2Contai-EcsElast-123456789012",
      "containerName": "simple-demo",
      "containerPort": 80
    }
  ],
  "desiredCount": 10,
  "role": "ecsServiceRole"
}
```

Comando:

```
aws ecs create-service \
  --cluster MyCluster \
  --service-name ecs-simple-service-elb \
  --cli-input-json file://ecs-simple-service-elb.json
```

Salida:

```
{
  "service": {
    "status": "ACTIVE",
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/ecs-demo:1",
    "pendingCount": 0,
    "loadBalancers": [
      {
        "containerName": "ecs-demo",
        "containerPort": 80,

```



```

        "loadBalancerName": "EC2Contai-EcsElast-123456789012"
    }
],
"roleArn": "arn:aws:iam::123456789012:role/ecsServiceRole",
"desiredCount": 10,
"serviceName": "ecs-simple-service-elb",
"clusterArn": "arn:aws:ecs:<us-west-2:123456789012:cluster/MyCluster",
"serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/ecs-simple-
service-elb",
"deployments": [
    {
        "status": "PRIMARY",
        "pendingCount": 0,
        "createdAt": 1428100239.123,
        "desiredCount": 10,
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/ecs-demo:1",
        "updatedAt": 1428100239.123,
        "id": "ecs-svc/1234567890123456789",
        "runningCount": 0
    }
],
"events": [],
"runningCount": 0
}
}

```

Para obtener más información, consulta [Cómo crear un servicio](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [CreateService](#) la Referencia de AWS CLI comandos.

## create-task-set

En el siguiente ejemplo de código, se muestra cómo usar create-task-set.

### AWS CLI

Para crear un conjunto de tareas

En el siguiente create-task-set ejemplo, se crea un conjunto de tareas en un servicio que utiliza un controlador de despliegue externo.

```
aws ecs create-task-set \  
  --cluster MyCluster \  
  --service MyService \  
  --task-definition MyTaskDefinition:2 \  
  --network-  
configuration "awsvpcConfiguration={subnets=[subnet-12344321],securityGroups=[sg-12344321]}"
```

Salida:

```
{  
  "taskSet": {  
    "id": "ecs-svc/1234567890123456789",  
    "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/  
MyService/ecs-svc/1234567890123456789",  
    "status": "ACTIVE",  
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/  
MyTaskDefinition:2",  
    "computedDesiredCount": 0,  
    "pendingCount": 0,  
    "runningCount": 0,  
    "createdAt": 1557128360.711,  
    "updatedAt": 1557128360.711,  
    "launchType": "EC2",  
    "networkConfiguration": {  
      "awsvpcConfiguration": {  
        "subnets": [  
          "subnet-12344321"  
        ],  
        "securityGroups": [  
          "sg-12344321"  
        ],  
        "assignPublicIp": "DISABLED"  
      }  
    },  
    "loadBalancers": [],  
    "serviceRegistries": [],  
    "scale": {  
      "value": 0.0,  
      "unit": "PERCENT"  
    },  
    "stabilityStatus": "STABILIZING",  
    "stabilityStatusAt": 1557128360.711  
  }  
}
```

```
}
```

- Para API obtener más información, consulte [CreateTaskSet](#) la Referencia de AWS CLI comandos.

## delete-account-setting

En el siguiente ejemplo de código, se muestra cómo usar `delete-account-setting`.

### AWS CLI

Para eliminar la configuración de la cuenta de un IAM usuario o IAM rol específico

En el siguiente ejemplo, `delete-account-setting` se elimina la configuración de la cuenta para el IAM usuario o IAM rol específico.

```
aws ecs delete-account-setting \  
  --name serviceLongArnFormat \  
  --principal-arn arn:aws:iam::123456789012:user/MyUser
```

Salida:

```
{  
  "setting": {  
    "name": "serviceLongArnFormat",  
    "value": "enabled",  
    "principalArn": "arn:aws:iam::123456789012:user/MyUser"  
  }  
}
```

Para obtener más información, consulte [Amazon Resource Names \(ARNs\) y IDs](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [DeleteAccountSetting](#) la Referencia de AWS CLI comandos.

## delete-attributes

En el siguiente ejemplo de código, se muestra cómo usar `delete-attributes`.

## AWS CLI

Para eliminar uno o más atributos personalizados de un ECS recurso de Amazon

Lo siguiente `delete-attributes` elimina un atributo con el nombre `stack` de una instancia de contenedor.

```
aws ecs delete-attributes \  
  --attributes name=stack, targetId=arn:aws:ecs:us-west-2:130757420319:container-  
instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34
```

Salida:

```
{  
  "attributes": [  
    {  
      "name": "stack",  
      "targetId": "arn:aws:ecs:us-west-2:130757420319:container-  
instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34",  
      "value": "production"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [DeleteAttributes](#) la Referencia de AWS CLI comandos.

## `delete-capacity-provider`

En el siguiente ejemplo de código, se muestra cómo usar `delete-capacity-provider`.

### AWS CLI

Ejemplo 1: Para eliminar un proveedor de capacidad mediante el nombre de recurso de Amazon (ARN)

En el siguiente `delete-capacity-provider` ejemplo, se elimina un proveedor de capacidad especificando el nombre de recurso de Amazon (ARN) del proveedor de capacidad. La eliminación del proveedor de capacidad, así ARN como el estado de la misma, se pueden recuperar mediante el `describe-capacity-providers` comando.

```
aws ecs delete-capacity-provider \
  --capacity-provider arn:aws:ecs:us-west-2:123456789012:capacity-provider/
  ExampleCapacityProvider
```

Salida:

```
{
  "capacityProvider": {
    "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
    provider/ExampleCapacityProvider",
    "name": "ExampleCapacityProvider",
    "status": "ACTIVE",
    "autoScalingGroupProvider": {
      "autoScalingGroupArn": "arn:aws:autoscaling:us-
      west-2:123456789012:autoScalingGroup:a1b2c3d4-5678-90ab-cdef-
      EXAMPLE11111:autoScalingGroupName/MyAutoScalingGroup",
      "managedScaling": {
        "status": "ENABLED",
        "targetCapacity": 100,
        "minimumScalingStepSize": 1,
        "maximumScalingStepSize": 10000
      },
      "managedTerminationProtection": "DISABLED"
    },
    "updateStatus": "DELETE_IN_PROGRESS",
    "tags": []
  }
}
```

Para obtener más información, consulte [Proveedores de capacidad de clúster](#) en la Guía para ECS desarrolladores de Amazon.

Ejemplo 2: Para eliminar un proveedor de capacidad utilizando el nombre

En el siguiente `delete-capacity-provider` ejemplo, se elimina un proveedor de capacidad especificando su nombre abreviado. El nombre abreviado y el estado de la eliminación del proveedor de capacidad se pueden recuperar mediante el `describe-capacity-providers` comando.

```
aws ecs delete-capacity-provider \
  --capacity-provider ExampleCapacityProvider
```

**Salida:**

```
{
  "capacityProvider": {
    "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/ExampleCapacityProvider",
    "name": "ExampleCapacityProvider",
    "status": "ACTIVE",
    "autoScalingGroupProvider": {
      "autoScalingGroupArn": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111:autoScalingGroupName/MyAutoScalingGroup",
      "managedScaling": {
        "status": "ENABLED",
        "targetCapacity": 100,
        "minimumScalingStepSize": 1,
        "maximumScalingStepSize": 10000
      },
      "managedTerminationProtection": "DISABLED"
    },
    "updateStatus": "DELETE_IN_PROGRESS",
    "tags": []
  }
}
```

Para obtener más información, consulte [Proveedores de capacidad de clúster](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [DeleteCapacityProvider](#) la Referencia de AWS CLI comandos.

**delete-cluster**

En el siguiente ejemplo de código, se muestra cómo usar `delete-cluster`.

**AWS CLI**

Eliminación de un clúster vacío

En el siguiente ejemplo de `delete-cluster` se elimina el clúster vacío especificado.

```
aws ecs delete-cluster --cluster MyCluster
```

Salida:

```
{
  "cluster": {
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "status": "INACTIVE",
    "clusterName": "MyCluster",
    "registeredContainerInstancesCount": 0,
    "pendingTasksCount": 0,
    "runningTasksCount": 0,
    "activeServicesCount": 0
    "statistics": [],
    "tags": []
  }
}
```

Para obtener más información, consulta [Eliminar un clúster](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [DeleteCluster](#) la Referencia de AWS CLI comandos.

## delete-service

En el siguiente ejemplo de código, se muestra cómo usar `delete-service`.

### AWS CLI

Eliminación de un servicio

En el siguiente ejemplo de `ecs delete-service` se elimina el servicio especificado de un clúster. Puede incluir el parámetro `--force` para eliminar un servicio aunque no se haya escalado a cero tareas.

```
aws ecs delete-service --cluster MyCluster --service MyService1 --force
```

Para obtener más información, consulta [Eliminar un servicio](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [DeleteService](#) la Referencia de AWS CLI comandos.

## delete-task-definitions

En el siguiente ejemplo de código, se muestra cómo usar `delete-task-definitions`.

### AWS CLI

Para eliminar una definición de tarea

En el siguiente `delete-task-definitions` ejemplo, se elimina una definición de INACTIVE tarea.

```
aws ecs delete-task-definitions \  
  --task-definition curltest:1
```

Salida:

```
{  
  "taskDefinitions": [  
    {  
      "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-definition/  
curltest:1",  
      "containerDefinitions": [  
        {  
          "name": "ctest",  
          "image": "mreferre/eksutils",  
          "cpu": 0,  
          "portMappings": [],  
          "essential": true,  
          "entryPoint": [  
            "sh",  
            "-c"  
          ],  
          "command": [  
            "curl ${ECS_CONTAINER_METADATA_URI_V4}/task"  
          ],  
          "environment": [],  
          "mountPoints": [],  
          "volumesFrom": [],  
          "logConfiguration": {  
            "logDriver": "awslogs",  
            "options": {  
              "awslogs-create-group": "true",  
              "awslogs-group": "/ecs/curltest",
```



```

        "awslogs-region": "us-east-1",
        "awslogs-stream-prefix": "ecs"
    }
}
],
"family": "curltest",
"taskRoleArn": "arn:aws:iam::123456789012:role/ecsTaskExecutionRole",
"executionRoleArn": "arn:aws:iam::123456789012:role/ecsTaskExecutionRole",
"networkMode": "awsvpc",
"revision": 1,
"volumes": [],
"status": "DELETE_IN_PROGRESS",
"compatibilities": [
    "EC2",
    "FARGATE"
],
"requiresCompatibilities": [
    "FARGATE"
],
"cpu": "256",
"memory": "512",
"registeredAt": "2021-09-10T12:56:24.704000+00:00",
"deregisteredAt": "2023-03-14T15:20:59.419000+00:00",
"registeredBy": "arn:aws:sts::123456789012:assumed-role/Admin/jdoe"
}
],
"failures": []
}

```

Para obtener más información, consulte [Definiciones de ECS tareas de Amazon](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [DeleteTaskDefinitions](#) la Referencia de AWS CLI comandos.

## delete-task-set

En el siguiente ejemplo de código, se muestra cómo usar `delete-task-set`.

### AWS CLI

Para eliminar un conjunto de tareas

El siguiente `delete-task-set` ejemplo muestra cómo eliminar un conjunto de tareas. Puede incluir el `--force` parámetro para eliminar un conjunto de tareas aunque no se haya escalado a cero.

```
aws ecs delete-task-set \  
  --cluster MyCluster \  
  --service MyService \  
  --task-set arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/MyService/ecs-  
svc/1234567890123456789 \  
  --force
```

Salida:

```
{  
  "taskSet": {  
    "id": "ecs-svc/1234567890123456789",  
    "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/  
MyService/ecs-svc/1234567890123456789",  
    "status": "DRAINING",  
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/  
sample-fargate:2",  
    "computedDesiredCount": 0,  
    "pendingCount": 0,  
    "runningCount": 0,  
    "createdAt": 1557130260.276,  
    "updatedAt": 1557130290.707,  
    "launchType": "EC2",  
    "networkConfiguration": {  
      "awsvpcConfiguration": {  
        "subnets": [  
          "subnet-12345678"  
        ],  
        "securityGroups": [  
          "sg-12345678"  
        ],  
        "assignPublicIp": "DISABLED"  
      }  
    },  
    "loadBalancers": [],  
    "serviceRegistries": [],  
    "scale": {  
      "value": 0.0,  
      "unit": "PERCENT"  
    }  
  }  
}
```

```

    },
    "stabilityStatus": "STABILIZING",
    "stabilityStatusAt": 1557130290.707
  }
}

```

- Para API obtener más información, consulte [DeleteTaskSet](#) la Referencia de AWS CLI comandos.

## deregister-container-instance

En el siguiente ejemplo de código, se muestra cómo usar `deregister-container-instance`.

### AWS CLI

Para anular el registro de una instancia de contenedor de un clúster

El siguiente `deregister-container-instance` ejemplo anula el registro de una instancia contenedora del clúster especificado. Si aún hay tareas en ejecución en la instancia contenedora, debes detenerlas antes de anular el registro o usar la opción. `--force`

```

aws ecs deregister-container-instance \
  --cluster arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster \
  --container-instance arn:aws:ecs:us-west-2:123456789012:container-instance/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
  --force

```

Salida:

```

{
  "containerInstance": {
    "remainingResources": [
      {
        "integerValue": 1024,
        "doubleValue": 0.0,
        "type": "INTEGER",
        "longValue": 0,
        "name": "CPU"
      },
      {
        "integerValue": 985,
        "doubleValue": 0.0,

```

```
        "type": "INTEGER",
        "longValue": 0,
        "name": "MEMORY"
    },
    {
        "type": "STRINGSET",
        "integerValue": 0,
        "name": "PORTS",
        "stringSetValue": [
            "22",
            "2376",
            "2375",
            "51678",
            "51679"
        ],
        "longValue": 0,
        "doubleValue": 0.0
    },
    {
        "type": "STRINGSET",
        "integerValue": 0,
        "name": "PORTS_UDP",
        "stringSetValue": [],
        "longValue": 0,
        "doubleValue": 0.0
    }
],
"agentConnected": true,
"attributes": [
    {
        "name": "ecs.capability.secrets.asm.environment-variables"
    },
    {
        "name": "com.amazonaws.ecs.capability.logging-driver.syslog"
    },
    {
        "value": "ami-01a82c3f3ce2c3ba58",
        "name": "ecs.ami-id"
    },
    {
        "name": "ecs.capability.secrets.asm.bootstrap.log-driver"
    },
    {
        "name": "com.amazonaws.ecs.capability.logging-driver.none"
    }
]
```

```
},
{
  "name": "ecs.capability.ecr-endpoint"
},
{
  "name": "com.amazonaws.ecs.capability.logging-driver.json-file"
},
{
  "value": "vpc-1234567890123467",
  "name": "ecs.vpc-id"
},
{
  "name": "ecs.capability.execution-role-awslogs"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.17"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.18"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.19"
},
{
  "name": "ecs.capability.docker-plugin.local"
},
{
  "name": "ecs.capability.task-eni"
},
{
  "name": "ecs.capability.task-cpu-mem-limit"
},
{
  "name": "ecs.capability.secrets.ssm.bootstrap.log-driver"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.30"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.31"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.32"
},
}
```

```
{
  "name": "ecs.capability.execution-role-ecr-pull"
},
{
  "name": "ecs.capability.container-health-check"
},
{
  "value": "subnet-1234567890123467",
  "name": "ecs.subnet-id"
},
{
  "value": "us-west-2a",
  "name": "ecs.availability-zone"
},
{
  "value": "t2.micro",
  "name": "ecs.instance-type"
},
{
  "name": "com.amazonaws.ecs.capability.task-iam-role-network-host"
},
{
  "name": "ecs.capability.aws-appmesh"
},
{
  "name": "com.amazonaws.ecs.capability.logging-driver.awslogs"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.24"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.25"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.26"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.27"
},
{
  "name": "com.amazonaws.ecs.capability.privileged-container"
},
{
  "name": "ecs.capability.container-ordering"
}
```

```
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.28"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.29"
    },
    {
      "value": "x86_64",
      "name": "ecs.cpu-architecture"
    },
    {
      "value": "93f43776-2018.10.0",
      "name": "ecs.capability.cni-plugin-version"
    },
    {
      "name": "ecs.capability.secrets.ssm.environment-variables"
    },
    {
      "name": "ecs.capability.pid-ipc-namespace-sharing"
    },
    {
      "name": "com.amazonaws.ecs.capability.ecr-auth"
    },
    {
      "value": "linux",
      "name": "ecs.os-type"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.20"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.21"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.22"
    },
    {
      "name": "ecs.capability.task-eia"
    },
    {
      "name": "ecs.capability.private-registry-
authentication.secretsmanager"
    },
  },
```

```
    {
      "name": "com.amazonaws.ecs.capability.task-iam-role"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.23"
    }
  ],
  "pendingTasksCount": 0,
  "tags": [],
  "containerInstanceArn": "arn:aws:ecs:us-west-2:123456789012:container-
instance/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
  "registeredResources": [
    {
      "integerValue": 1024,
      "doubleValue": 0.0,
      "type": "INTEGER",
      "longValue": 0,
      "name": "CPU"
    },
    {
      "integerValue": 985,
      "doubleValue": 0.0,
      "type": "INTEGER",
      "longValue": 0,
      "name": "MEMORY"
    },
    {
      "type": "STRINGSET",
      "integerValue": 0,
      "name": "PORTS",
      "stringSetValue": [
        "22",
        "2376",
        "2375",
        "51678",
        "51679"
      ],
      "longValue": 0,
      "doubleValue": 0.0
    },
    {
      "type": "STRINGSET",
      "integerValue": 0,
      "name": "PORTS_UDP",
```



```

        "stringSetValue": [],
        "longValue": 0,
        "doubleValue": 0.0
    }
],
"status": "INACTIVE",
"registeredAt": 1557768075.681,
"version": 4,
"versionInfo": {
    "agentVersion": "1.27.0",
    "agentHash": "aabe65ee",
    "dockerVersion": "DockerVersion: 18.06.1-ce"
},
"attachments": [],
"runningTasksCount": 0,
"ec2InstanceId": "i-12345678901234678"
}
}

```

Para obtener más información, consulte [Anular el registro de una instancia de contenedor](#) en la Guía para desarrolladores. ECS

- Para API obtener más información, consulte la Referencia [DeregisterContainerInstance](#) de AWS CLI comandos.

## deregister-task-definition

En el siguiente ejemplo de código, se muestra cómo usar `deregister-task-definition`.

### AWS CLI

Para anular el registro de una definición de tarea

El siguiente `deregister-task-definition` ejemplo anula el registro de la primera revisión de la definición de `curler` tarea en la región predeterminada.

```
aws ecs deregister-task-definition --task-definition curler:1
```

Tenga en cuenta que en el resultado resultante, el estado de la definición de la tarea muestra: INACTIVE

```
{
```

```
"taskDefinition": {
  "status": "INACTIVE",
  "family": "curler",
  "volumes": [],
  "taskDefinitionArn": "arn:aws:ecs:us-west-2:123456789012:task-definition/
curler:1",
  "containerDefinitions": [
    {
      "environment": [],
      "name": "curler",
      "mountPoints": [],
      "image": "curl:latest",
      "cpu": 100,
      "portMappings": [],
      "entryPoint": [],
      "memory": 256,
      "command": [
        "curl -v http://example.com/"
      ],
      "essential": true,
      "volumesFrom": []
    }
  ],
  "revision": 1
}
```

Para obtener más información, consulte [Definiciones de ECS tareas de Amazon](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [DeregisterTaskDefinition](#) la Referencia de AWS CLI comandos.

## describe-capacity-providers

En el siguiente ejemplo de código, se muestra cómo usar `describe-capacity-providers`.

### AWS CLI

Ejemplo 1: Para describir todos los proveedores de capacidad

El siguiente `describe-capacity-providers` ejemplo recupera detalles sobre todos los proveedores de capacidad.

**aws ecs describe-capacity-providers**

Salida:

```
{
  "capacityProviders": [
    {
      "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/MyCapacityProvider",
      "name": "MyCapacityProvider",
      "status": "ACTIVE",
      "autoScalingGroupProvider": {
        "autoScalingGroupArn": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111:autoScalingGroupName/MyAutoScalingGroup",
        "managedScaling": {
          "status": "ENABLED",
          "targetCapacity": 100,
          "minimumScalingStepSize": 1,
          "maximumScalingStepSize": 1000
        },
        "managedTerminationProtection": "ENABLED"
      },
      "tags": []
    },
    {
      "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/FARGATE",
      "name": "FARGATE",
      "status": "ACTIVE",
      "tags": []
    },
    {
      "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/FARGATE_SPOT",
      "name": "FARGATE_SPOT",
      "status": "ACTIVE",
      "tags": []
    }
  ]
}
```

Para obtener más información, consulte [Proveedores de capacidad de clúster](#) en la Guía para ECS desarrolladores de Amazon.

Ejemplo 2: Para describir un proveedor de capacidad específico

El siguiente describe-capacity-providers ejemplo recupera detalles sobre un proveedor de capacidad específico. Si se utiliza el `--include TAGS` parámetro, se añadirán a la salida las etiquetas asociadas al proveedor de capacidad.

```
aws ecs describe-capacity-providers \  
  --capacity-providers MyCapacityProvider \  
  --include TAGS
```

Salida:

```
{  
  "capacityProviders": [  
    {  
      "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-  
provider/MyCapacityProvider",  
      "name": "MyCapacityProvider",  
      "status": "ACTIVE",  
      "autoScalingGroupProvider": {  
        "autoScalingGroupArn": "arn:aws:autoscaling:us-  
west-2:123456789012:autoScalingGroup:a1b2c3d4-5678-90ab-cdef-  
EXAMPLE11111:autoScalingGroupName/MyAutoScalingGroup",  
        "managedScaling": {  
          "status": "ENABLED",  
          "targetCapacity": 100,  
          "minimumScalingStepSize": 1,  
          "maximumScalingStepSize": 1000  
        },  
        "managedTerminationProtection": "ENABLED"  
      },  
      "tags": [  
        {  
          "key": "environment",  
          "value": "production"  
        }  
      ]  
    }  
  ]  
}
```

```
}
```

Para obtener más información, consulte [Proveedores de capacidad de clúster](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeCapacityProviders](#) la Referencia de AWS CLI comandos.

## describe-clusters

En el siguiente ejemplo de código, se muestra cómo usar `describe-clusters`.

### AWS CLI

Ejemplo 1: Descripción de un clúster

En el siguiente ejemplo de `describe-clusters` se recuperan los detalles sobre el clúster especificado.

```
aws ecs describe-clusters \  
  --cluster default
```

Salida:

```
{  
  "clusters": [  
    {  
      "status": "ACTIVE",  
      "clusterName": "default",  
      "registeredContainerInstancesCount": 0,  
      "pendingTasksCount": 0,  
      "runningTasksCount": 0,  
      "activeServicesCount": 1,  
      "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/default"  
    }  
  ],  
  "failures": []  
}
```

Para obtener más información, consulte [Amazon ECS Clusters](#) en la Guía para ECS desarrolladores de Amazon.

## Ejemplo 2: Descripción de un clúster con la opción de adjuntar

En el siguiente `describe-clusters` ejemplo se especifica la `ATTACHMENTS` opción. Recupera detalles sobre el clúster especificado y una lista de los recursos adjuntos al clúster en forma de archivos adjuntos. Cuando se utiliza un proveedor de capacidad con un clúster, los recursos, ya sean AutoScaling planes o políticas de escalado, se representarán como `asp` o `ATTACHMENTS as_policy`.

```
aws ecs describe-clusters \  
  --include ATTACHMENTS \  
  --clusters sampleCluster
```

Salida:

```
{  
  "clusters": [  
    {  
      "clusterArn": "arn:aws:ecs:af-south-1:123456789222:cluster/  
sampleCluster",  
      "clusterName": "sampleCluster",  
      "status": "ACTIVE",  
      "registeredContainerInstancesCount": 0,  
      "runningTasksCount": 0,  
      "pendingTasksCount": 0,  
      "activeServicesCount": 0,  
      "statistics": [],  
      "tags": [],  
      "settings": [],  
      "capacityProviders": [  
        "sampleCapacityProvider"  
      ],  
      "defaultCapacityProviderStrategy": [],  
      "attachments": [  
        {  
          "id": "a1b2c3d4-5678-901b-cdef-EXAMPLE22222",  
          "type": "as_policy",  
          "status": "CREATED",  
          "details": [  
            {  
              "name": "capacityProviderName",  
              "value": "sampleCapacityProvider"  
            }  
          ],  
        }  
      ],  
    }  
  ]  
}
```

```

        {
            "name": "scalingPolicyName",
            "value": "ECSTaskSetAutoScalingPolicy-3048e262-
fe39-4eaf-826d-6f975d303188"
        }
    ]
}
],
"attachmentsStatus": "UPDATE_COMPLETE"
}
],
"failures": []
}

```

Para obtener más información, consulte [Amazon ECS Clusters](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeClusters](#) la Referencia de AWS CLI comandos.

## describe-container-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-container-instances`.

### AWS CLI

Para describir la instancia contenedora

En el siguiente `describe-container-instances` ejemplo, se recuperan los detalles de una instancia contenedora del update clúster, utilizando la instancia contenedora UUID como identificador.

```

aws ecs describe-container-instances \
  --cluster update \
  --container-instances a1b2c3d4-5678-90ab-cdef-11111EXAMPLE

```

Salida:

```

{
  "failures": [],
  "containerInstances": [
    {

```

```
"status": "ACTIVE",
"registeredResources": [
  {
    "integerValue": 2048,
    "longValue": 0,
    "type": "INTEGER",
    "name": "CPU",
    "doubleValue": 0.0
  },
  {
    "integerValue": 3955,
    "longValue": 0,
    "type": "INTEGER",
    "name": "MEMORY",
    "doubleValue": 0.0
  },
  {
    "name": "PORTS",
    "longValue": 0,
    "doubleValue": 0.0,
    "stringSetValue": [
      "22",
      "2376",
      "2375",
      "51678"
    ],
    "type": "STRINGSET",
    "integerValue": 0
  }
],
"ec2InstanceId": "i-A1B2C3D4",
"agentConnected": true,
"containerInstanceArn": "arn:aws:ecs:us-west-2:123456789012:container-
instance/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
"pendingTasksCount": 0,
"remainingResources": [
  {
    "integerValue": 2048,
    "longValue": 0,
    "type": "INTEGER",
    "name": "CPU",
    "doubleValue": 0.0
  },
  {
```



```
        "integerValue": 3955,
        "longValue": 0,
        "type": "INTEGER",
        "name": "MEMORY",
        "doubleValue": 0.0
    },
    {
        "name": "PORTS",
        "longValue": 0,
        "doubleValue": 0.0,
        "stringSetValue": [
            "22",
            "2376",
            "2375",
            "51678"
        ],
        "type": "STRINGSET",
        "integerValue": 0
    }
],
"runningTasksCount": 0,
"versionInfo": {
    "agentVersion": "1.0.0",
    "agentHash": "4023248",
    "dockerVersion": "DockerVersion: 1.5.0"
}
}
]
```

Para obtener más información, consulte [Amazon ECS Container Instances](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeContainerInstances](#) la Referencia de AWS CLI comandos.

## **describe-services**

En el siguiente ejemplo de código, se muestra cómo usar `describe-services`.

### AWS CLI

Para describir un servicio

En el siguiente describe-services ejemplo, se recuperan los detalles del my-http-service servicio del clúster predeterminado.

```
aws ecs describe-services --services my-http-service
```

Salida:

```
{
  "services": [
    {
      "status": "ACTIVE",
      "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
amazon-ecs-sample:1",
      "pendingCount": 0,
      "loadBalancers": [],
      "desiredCount": 10,
      "createdAt": 1466801808.595,
      "serviceName": "my-http-service",
      "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/default",
      "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/my-http-
service",
      "deployments": [
        {
          "status": "PRIMARY",
          "pendingCount": 0,
          "createdAt": 1466801808.595,
          "desiredCount": 10,
          "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/amazon-ecs-sample:1",
          "updatedAt": 1428326312.703,
          "id": "ecs-svc/1234567890123456789",
          "runningCount": 10
        }
      ],
      "events": [
        {
          "message": "(service my-http-service) has reached a steady
state.",
          "id": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
          "createdAt": 1466801812.435
        }
      ],
      "runningCount": 10
    }
  ]
}
```

```
    }
  ],
  "failures": []
}
```

Para obtener más información, consulta [Servicios](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeServices](#) la Referencia de AWS CLI comandos.

## describe-task-definition

En el siguiente ejemplo de código, se muestra cómo usar `describe-task-definition`.

### AWS CLI

Para describir una definición de tarea

En el siguiente `describe-task-definition` ejemplo, se recuperan los detalles de una definición de tarea.

```
aws ecs describe-task-definition \
  --task-definition hello_world:8
```

Salida:

```
{
  "tasks": [
    {
      "attachments": [
        {
          "id": "17f3dff6-a9e9-4d83-99a9-7eb5193c2634",
          "type": "ElasticNetworkInterface",
          "status": "ATTACHED",
          "details": [
            {
              "name": "subnetId",
              "value": "subnet-0d0eab1bb38d5ca64"
            },
            {
              "name": "networkInterfaceId",
              "value": "eni-0d542ffb4a12aa6d9"
            }
          ]
        }
      ]
    }
  ]
}
```

```

        },
        {
            "name": "macAddress",
            "value": "0e:6d:18:f6:2d:29"
        },
        {
            "name": "privateDnsName",
            "value": "ip-10-0-1-170.ec2.internal"
        },
        {
            "name": "privateIPv4Address",
            "value": "10.0.1.170"
        }
    ]
}
],
"attributes": [
    {
        "name": "ecs.cpu-architecture",
        "value": "x86_64"
    }
],
"availabilityZone": "us-east-1b",
"clusterArn": "arn:aws:ecs:us-east-1:053534965804:cluster/fargate-
cluster",
"connectivity": "CONNECTED",
"connectivityAt": "2023-11-28T11:10:52.907000-05:00",
"containers": [
    {
        "containerArn": "arn:aws:ecs:us-east-1:053534965804:container/
fargate-cluster/
c524291ae4154100b601a543108b193a/772c4784-92ae-414e-8df2-03d3358e39fa",
        "taskArn": "arn:aws:ecs:us-east-1:053534965804:task/fargate-
cluster/c524291ae4154100b601a543108b193a",
        "name": "web",
        "image": "nginx",
        "imageDigest":
"sha256:10d1f5b58f74683ad34eb29287e07dab1e90f10af243f151bb50aa5dbb4d62ee",
        "runtimeId": "c524291ae4154100b601a543108b193a-265927825",
        "lastStatus": "RUNNING",
        "networkBindings": [],
        "networkInterfaces": [
            {
                "attachmentId": "17f3dff6-a9e9-4d83-99a9-7eb5193c2634",

```

```
        "privateIpv4Address": "10.0.1.170"
      }
    ],
    "healthStatus": "HEALTHY",
    "cpu": "99",
    "memory": "100"
  },
  {
    "containerArn": "arn:aws:ecs:us-east-1:053534965804:container/
fargate-cluster/c524291ae4154100b601a543108b193a/c051a779-40d2-48ca-
ad5e-6ec875ceb610",
    "taskArn": "arn:aws:ecs:us-east-1:053534965804:task/fargate-
cluster/c524291ae4154100b601a543108b193a",
    "name": "aws-guardduty-agent-FvWGoDU",
    "imageDigest":
"sha256:359b8b014e5076c625daa1056090e522631587a7afa3b2e055edda6bd1141017",
    "runtimeId": "c524291ae4154100b601a543108b193a-505093495",
    "lastStatus": "RUNNING",
    "networkBindings": [],
    "networkInterfaces": [
      {
        "attachmentId": "17f3dff6-a9e9-4d83-99a9-7eb5193c2634",
        "privateIpv4Address": "10.0.1.170"
      }
    ],
    "healthStatus": "UNKNOWN"
  }
],
"cpu": "256",
"createdAt": "2023-11-28T11:10:49.299000-05:00",
"desiredStatus": "RUNNING",
"enableExecuteCommand": false,
"group": "family:webserver",
"healthStatus": "HEALTHY",
"lastStatus": "RUNNING",
"launchType": "FARGATE",
"memory": "512"
"platformVersion": "1.4.0",
"platformFamily": "Linux",
"pullStartedAt": "2023-11-28T11:10:59.773000-05:00",
"pullStoppedAt": "2023-11-28T11:11:12.624000-05:00",
"startedAt": "2023-11-28T11:11:20.316000-05:00",
"tags": [],
```

```

        "taskArn": "arn:aws:ecs:us-east-1:053534965804:task/fargate-cluster/
c524291ae4154100b601a543108b193a",
        "taskDefinitionArn": "arn:aws:ecs:us-east-1:053534965804:task-
definition/webserver:5",
        "version": 4,
        "ephemeralStorage": {
            "sizeInGiB": 20
        }
    },
    "failures": []
}

```

Para obtener más información, consulte [Definiciones de ECS tareas de Amazon](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeTaskDefinition](#) la Referencia de AWS CLI comandos.

## describe-task-sets

En el siguiente ejemplo de código, se muestra cómo usar `describe-task-sets`.

### AWS CLI

Para describir un conjunto de tareas

El siguiente `describe-task-sets` ejemplo describe un conjunto de tareas en un servicio que utiliza un implementador externo.

```

aws ecs describe-task-sets \
  --cluster MyCluster \
  --service MyService \
  --task-sets arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/MyService/ecs-
svc/1234567890123456789

```

Salida:

```

{
  "taskSets": [
    {

```

```

    "id": "ecs-svc/1234567890123456789",
    "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/
MyService/ecs-svc/1234567890123456789",
    "status": "ACTIVE",
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
sample-fargate:2",
    "computedDesiredCount": 0,
    "pendingCount": 0,
    "runningCount": 0,
    "createdAt": 1557207715.195,
    "updatedAt": 1557207740.014,
    "launchType": "EC2",
    "networkConfiguration": {
      "awsvpcConfiguration": {
        "subnets": [
          "subnet-12344321"
        ],
        "securityGroups": [
          "sg-1234431"
        ],
        "assignPublicIp": "DISABLED"
      }
    },
    "loadBalancers": [],
    "serviceRegistries": [],
    "scale": {
      "value": 0.0,
      "unit": "PERCENT"
    },
    "stabilityStatus": "STEADY_STATE",
    "stabilityStatusAt": 1557207740.014
  }
],
"failures": []
}

```

- Para API obtener más información, consulte [DescribeTaskSets](#) la Referencia de AWS CLI comandos.

## describe-tasks

En el siguiente ejemplo de código, se muestra cómo usar `describe-tasks`.

## AWS CLI

### Ejemplo 1: Descripción de una sola tarea

En el siguiente ejemplo de `describe-tasks` se recuperan los detalles de una tarea de un clúster. Puede especificar la tarea utilizando el identificador o la totalidad ARN de la tarea. En este ejemplo, se utiliza la totalidad ARN de la tarea.

```
aws ecs describe-tasks \  
  --cluster MyCluster \  
  --tasks arn:aws:ecs:us-east-1:123456789012:task/MyCluster/4d590253bb114126b7afa7b58EXAMPLE
```

Salida:

```
{  
  "tasks": [  
    {  
      "attachments": [],  
      "attributes": [  
        {  
          "name": "ecs.cpu-architecture",  
          "value": "x86_64"  
        }  
      ],  
      "availabilityZone": "us-east-1b",  
      "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/MyCluster",  
      "connectivity": "CONNECTED",  
      "connectivityAt": "2021-08-11T12:21:26.681000-04:00",  
      "containerInstanceArn": "arn:aws:ecs:us-east-1:123456789012:container-instance/test/025c7e2c5e054a6790a29fc1fEXAMPLE",  
      "containers": [  
        {  
          "containerArn": "arn:aws:ecs:us-east-1:123456789012:container/MyCluster/4d590253bb114126b7afa7b58eea9221/a992d1cc-ea46-474a-b6e8-24688EXAMPLE",  
          "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/MyCluster/4d590253bb114126b7afa7b58EXAMPLE",  
          "name": "simple-app",  
          "image": "httpd:2.4",  
          "runtimeId":  
            "91251eed27db90006ad67b1a08187290869f216557717dd5c39b37c94EXAMPLE",  
          "lastStatus": "RUNNING",  
          "networkBindings": [  

```



```

        {
            "bindIP": "0.0.0.0",
            "containerPort": 80,
            "hostPort": 80,
            "protocol": "tcp"
        }
    ],
    "networkInterfaces": [],
    "healthStatus": "UNKNOWN",
    "cpu": "10",
    "memory": "300"
}
],
"cpu": "10",
"createdAt": "2021-08-11T12:21:26.681000-04:00",
"desiredStatus": "RUNNING",
"enableExecuteCommand": false,
"group": "service:testupdate",
"healthStatus": "UNKNOWN",
"lastStatus": "RUNNING",
"launchType": "EC2",
"memory": "300",
"overrides": {
    "containerOverrides": [
        {
            "name": "simple-app"
        }
    ],
    "inferenceAcceleratorOverrides": []
},
"pullStartedAt": "2021-08-11T12:21:28.234000-04:00",
"pullStoppedAt": "2021-08-11T12:21:33.793000-04:00",
"startedAt": "2021-08-11T12:21:34.945000-04:00",
"startedBy": "ecs-svc/968695068243EXAMPLE",
"tags": [],
"taskArn": "arn:aws:ecs:us-east-1:123456789012:task/MyCluster/4d590253bb114126b7afa7b58eea9221",
"taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-definition/console-sample-app-static2:1",
"version": 2
}
],
"failures": []

```

```
}
```

Para obtener más información, consulte [Definiciones de ECS tareas de Amazon](#) en la Guía para ECS desarrolladores de Amazon.

## Ejemplo 2: Descripción de varias tareas

En el siguiente ejemplo de `describe-tasks` se recuperan los detalles de varias tareas de un clúster. Puede especificar la tarea utilizando el ID o la totalidad ARN de la tarea. En este ejemplo, se utilizan todas IDs las tareas.

```
aws ecs describe-tasks \  
  --cluster MyCluster \  
  --tasks "74de0355a10a4f979ac495c14EXAMPLE" "d789e94343414c25b9f6bd59eEXAMPLE"
```

Salida:

```
{  
  "tasks": [  
    {  
      "attachments": [  
        {  
          "id": "d9e7735a-16aa-4128-bc7a-b2d51EXAMPLE",  
          "type": "ElasticNetworkInterface",  
          "status": "ATTACHED",  
          "details": [  
            {  
              "name": "subnetId",  
              "value": "subnet-0d0eab1bb3EXAMPLE"  
            },  
            {  
              "name": "networkInterfaceId",  
              "value": "eni-0fa40520aeEXAMPLE"  
            },  
            {  
              "name": "macAddress",  
              "value": "0e:89:76:28:07:b3"  
            },  
            {  
              "name": "privateDnsName",  
              "value": "ip-10-0-1-184.ec2.internal"  
            },  
            {
```

```
        "name": "privateIPv4Address",
        "value": "10.0.1.184"
      }
    ]
  },
  "attributes": [
    {
      "name": "ecs.cpu-architecture",
      "value": "x86_64"
    }
  ],
  "availabilityZone": "us-east-1b",
  "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/MyCluster",
  "connectivity": "CONNECTED",
  "connectivityAt": "2021-12-20T12:13:37.875000-05:00",
  "containers": [
    {
      "containerArn": "arn:aws:ecs:us-east-1:123456789012:container/
MyCluster/74de0355a10a4f979ac495c14EXAMPLE/aad3ba00-83b3-4dac-84d4-11f8cEXAMPLE",
      "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/
MyCluster/74de0355a10a4f979ac495c14EXAMPLE",
      "name": "web",
      "image": "nginx",
      "runtimeId": "74de0355a10a4f979ac495c14EXAMPLE-265927825",
      "lastStatus": "RUNNING",
      "networkBindings": [],
      "networkInterfaces": [
        {
          "attachmentId": "d9e7735a-16aa-4128-bc7a-b2d51EXAMPLE",
          "privateIpv4Address": "10.0.1.184"
        }
      ],
      "healthStatus": "UNKNOWN",
      "cpu": "99",
      "memory": "100"
    }
  ],
  "cpu": "256",
  "createdAt": "2021-12-20T12:13:20.226000-05:00",
  "desiredStatus": "RUNNING",
  "enableExecuteCommand": false,
  "group": "service:tdsevicetag",
  "healthStatus": "UNKNOWN",
```

```

    "lastStatus": "RUNNING",
    "launchType": "FARGATE",
    "memory": "512",
    "overrides": {
      "containerOverrides": [
        {
          "name": "web"
        }
      ],
      "inferenceAcceleratorOverrides": []
    },
    "platformVersion": "1.4.0",
    "platformFamily": "Linux",
    "pullStartedAt": "2021-12-20T12:13:42.665000-05:00",
    "pullStoppedAt": "2021-12-20T12:13:46.543000-05:00",
    "startedAt": "2021-12-20T12:13:48.086000-05:00",
    "startedBy": "ecs-svc/988401040018EXAMPLE",
    "tags": [],
    "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/MyCluster/74de0355a10a4f979ac495c14EXAMPLE",
    "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-definition/webserver:2",
    "version": 3,
    "ephemeralStorage": {
      "sizeInGiB": 20
    }
  },
  {
    "attachments": [
      {
        "id": "214eb5a9-45cd-4bf8-87bc-57fefEXAMPLE",
        "type": "ElasticNetworkInterface",
        "status": "ATTACHED",
        "details": [
          {
            "name": "subnetId",
            "value": "subnet-0d0eab1bb3EXAMPLE"
          },
          {
            "name": "networkInterfaceId",
            "value": "eni-064c7766daEXAMPLE"
          },
          {
            "name": "macAddress",

```

```

        "value": "0e:76:83:01:17:a9"
      },
      {
        "name": "privateDnsName",
        "value": "ip-10-0-1-41.ec2.internal"
      },
      {
        "name": "privateIPv4Address",
        "value": "10.0.1.41"
      }
    ]
  },
  "attributes": [
    {
      "name": "ecs.cpu-architecture",
      "value": "x86_64"
    }
  ],
  "availabilityZone": "us-east-1b",
  "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/MyCluster",
  "connectivity": "CONNECTED",
  "connectivityAt": "2021-12-20T12:13:35.243000-05:00",
  "containers": [
    {
      "containerArn": "arn:aws:ecs:us-east-1:123456789012:container/MyCluster/d789e94343414c25b9f6bd59eEXAMPLE/9afef792-609b-43a5-bb6a-3efdbEXAMPLE",
      "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/MyCluster/d789e94343414c25b9f6bd59eEXAMPLE",
      "name": "web",
      "image": "nginx",
      "runtimeId": "d789e94343414c25b9f6bd59eEXAMPLE-265927825",
      "lastStatus": "RUNNING",
      "networkBindings": [],
      "networkInterfaces": [
        {
          "attachmentId": "214eb5a9-45cd-4bf8-87bc-57fefEXAMPLE",
          "privateIpv4Address": "10.0.1.41"
        }
      ],
      "healthStatus": "UNKNOWN",
      "cpu": "99",
      "memory": "100"
    }
  ]
}

```

```
    ],
    "cpu": "256",
    "createdAt": "2021-12-20T12:13:20.226000-05:00",
    "desiredStatus": "RUNNING",
    "enableExecuteCommand": false,
    "group": "service:tdsevicetag",
    "healthStatus": "UNKNOWN",
    "lastStatus": "RUNNING",
    "launchType": "FARGATE",
    "memory": "512",
    "overrides": {
      "containerOverrides": [
        {
          "name": "web"
        }
      ],
      "inferenceAcceleratorOverrides": []
    },
    "platformVersion": "1.4.0",
    "platformFamily": "Linux",
    "pullStartedAt": "2021-12-20T12:13:44.611000-05:00",
    "pullStoppedAt": "2021-12-20T12:13:48.251000-05:00",
    "startedAt": "2021-12-20T12:13:49.326000-05:00",
    "startedBy": "ecs-svc/988401040018EXAMPLE",
    "tags": [],
    "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/MyCluster/d789e94343414c25b9f6bd59eEXAMPLE",
    "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-definition/webserver:2",
    "version": 3,
    "ephemeralStorage": {
      "sizeInGiB": 20
    }
  }
],
"failures": []
}
```

Para obtener más información, consulte [Definiciones de ECS tareas de Amazon](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeTasks](#) la Referencia de AWS CLI comandos.

## execute-command

En el siguiente ejemplo de código, se muestra cómo usar `execute-command`.

### AWS CLI

Para ejecutar un comando `/bin/sh` interactivo

El siguiente `execute-command` ejemplo ejecuta un comando `/bin/sh` interactivo en un contenedor con el nombre de una tarea con un MyContainer identificador de. `arn:aws:ecs:us-east-1:123456789012:task/MyCluster/d789e94343414c25b9f6bd59eEXAMPLE`

```
aws ecs execute-command \  
  --cluster MyCluster \  
  --task arn:aws:ecs:us-east-1:123456789012:task/MyCluster/  
d789e94343414c25b9f6bd59eEXAMPLE \  
  --container MyContainer \  
  --interactive \  
  --command "/bin/sh"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de Amazon ECS Exec para la depuración](#) en la Guía para ECSdesarrolladores de Amazon.

- Para API obtener más información, consulte la Referencia [ExecuteCommand](#) de AWS CLI comandos.

## list-account-settings

En el siguiente ejemplo de código, se muestra cómo usar `list-account-settings`.

### AWS CLI

Ejemplo 1: Para ver la configuración de una cuenta

El siguiente `list-account-settings` ejemplo muestra la configuración de cuenta efectiva de una cuenta.

```
aws ecs list-account-settings --effective-settings
```

Salida:

```
{
  "settings": [
    {
      "name": "containerInstanceLongArnFormat",
      "value": "enabled",
      "principalArn": "arn:aws:iam::123456789012:root"
    },
    {
      "name": "serviceLongArnFormat",
      "value": "enabled",
      "principalArn": "arn:aws:iam::123456789012:root"
    },
    {
      "name": "taskLongArnFormat",
      "value": "enabled",
      "principalArn": "arn:aws:iam::123456789012:root"
    }
  ]
}
```

Ejemplo 2: Para ver la configuración de la cuenta de un IAM usuario o IAM rol específico

En el siguiente `list-account-settings` ejemplo, se muestra la configuración de la cuenta para el IAM usuario o IAM rol especificado.

```
aws ecs list-account-settings --principal-arn arn:aws:iam::123456789012:user/MyUser
```

Salida:

```
{
  "settings": [
    {
      "name": "serviceLongArnFormat",
      "value": "enabled",
      "principalArn": "arn:aws:iam::123456789012:user/MyUser"
    }
  ]
}
```

Para obtener más información, consulte [Amazon Resource Names \(ARNs\) y IDs](#) en la Guía para ECS desarrolladores de Amazon.



- Para API obtener más información, consulte [ListAccountSettings](#) la Referencia de AWS CLI comandos.

## list-attributes

En el siguiente ejemplo de código, se muestra cómo usar `list-attributes`.

### AWS CLI

Para enumerar las instancias de contenedor que contienen un atributo específico

En el siguiente ejemplo, se enumeran los atributos de las instancias de contenedor que tienen el `stack=production` atributo en el clúster predeterminado.

```
aws ecs list-attributes \  
  --target-type container-instance \  
  --attribute-name stack \  
  --attribute-value production \  
  --cluster default
```

Salida:

```
{  
  "attributes": [  
    {  
      "name": "stack",  
      "targetId": "arn:aws:ecs:us-west-2:130757420319:container-  
instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34",  
      "value": "production"  
    }  
  ]  
}
```

Para obtener más información, consulte [Configuración de Amazon ECS Container Agent](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [ListAttributes](#) la Referencia de AWS CLI comandos.

## list-clusters

En el siguiente ejemplo de código, se muestra cómo usar `list-clusters`.

## AWS CLI

### Creación de una lista de los clústeres disponibles

En el siguiente ejemplo de `list-clusters` se enumeran todos los clústeres disponibles.

```
aws ecs list-clusters
```

Salida:

```
{
  "clusterArns": [
    "arn:aws:ecs:us-west-2:123456789012:cluster/MyECSCluster1",
    "arn:aws:ecs:us-west-2:123456789012:cluster/AnotherECSCluster"
  ]
}
```

Para obtener más información, consulte [Amazon ECS Clusters](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [ListClusters](#) la Referencia de AWS CLI comandos.

## `list-container-instances`

En el siguiente ejemplo de código, se muestra cómo usar `list-container-instances`.

## AWS CLI

Para enumerar las instancias de contenedor de un clúster

En el siguiente `list-container-instances` ejemplo, se enumeran todas las instancias de contenedor disponibles en un clúster.

```
aws ecs list-container-instances --cluster MyCluster
```

Salida:

```
{
  "containerInstanceArns": [
```

```

    "arn:aws:ecs:us-west-2:123456789012:container-instance/MyCluster/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "arn:aws:ecs:us-west-2:123456789012:container-instance/MyCluster/
a1b2c3d4-5678-90ab-cdef-22222EXAMPLE"
  ]
}

```

Para obtener más información, consulte [Amazon ECS Container Instances](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [ListContainerInstances](#) la Referencia de AWS CLI comandos.

## list-services-by-namespace

En el siguiente ejemplo de código, se muestra cómo usar `list-services-by-namespace`.

### AWS CLI

Para enumerar los servicios de un espacio de nombres

En el siguiente `list-services-by-namespace` ejemplo, se enumeran todos los servicios configurados para el espacio de nombres especificado en la región predeterminada.

```

aws ecs list-services-by-namespace \
  --namespace service-connect

```

Salida:

```

{
  "serviceArns": [
    "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/MyService",
    "arn:aws:ecs:us-west-2:123456789012:service/tutorial/service-connect-nginx-
service"
  ]
}

```

Para obtener más información, consulte [Service Connect](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [ListServicesByNamespace](#) la Referencia de AWS CLI comandos.

## list-services

En el siguiente ejemplo de código, se muestra cómo usar `list-services`.

### AWS CLI

Creación de una lista de los servicios de un clúster

En el siguiente ejemplo de `list-services` se muestra cómo enumerar los servicios que se ejecutan en un clúster.

```
aws ecs list-services --cluster MyCluster
```

Salida:

```
{
  "serviceArns": [
    "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/MyService"
  ]
}
```

Para obtener más información, consulta [Servicios](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [ListServices](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas de un clúster específico.

```
aws ecs list-tags-for-resource \
  --resource-arn arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster
```

Salida:

```
{
  "tags": [
    {
      "key": "key1",
      "value": "value1"
    },
    {
      "key": "key2",
      "value": "value2"
    },
    {
      "key": "key3",
      "value": "value3"
    }
  ]
}
```

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## **list-task-definition-families**

En el siguiente ejemplo de código, se muestra cómo usar `list-task-definition-families`.

### AWS CLI

Ejemplo 1: Para enumerar las familias de definiciones de tareas registradas

En el siguiente `list-task-definition-families` ejemplo se enumeran todas las familias de definiciones de tareas registradas.

```
aws ecs list-task-definition-families
```

Salida:

```
{
  "families": [
    "node-js-app",
    "web-timer",
    "hpcc",
    "hpcc-c4-8xlarge"
  ]
}
```

```
]
}
```

Ejemplo 2: Para filtrar las familias de definiciones de tareas registradas

El siguiente `list-task-definition-families` ejemplo muestra las revisiones de la definición de tareas que comienzan por «hpcc».

```
aws ecs list-task-definition-families --family-prefix hpcc
```

Salida:

```
{
  "families": [
    "hpcc",
    "hpcc-c4-8xlarge"
  ]
}
```

Para obtener más información, consulte [Parámetros de definición de tareas](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [ListTaskDefinitionFamilies](#) la Referencia de AWS CLI comandos.

## list-task-definitions

En el siguiente ejemplo de código, se muestra cómo usar `list-task-definitions`.

AWS CLI

Ejemplo 1: Para enumerar las definiciones de tareas registradas

El siguiente `list-task-definitions` ejemplo muestra todas las definiciones de tareas registradas.

```
aws ecs list-task-definitions
```

Salida:

```
{
```

```
"taskDefinitionArns": [  
  "arn:aws:ecs:us-west-2:123456789012:task-definition/sleep300:2",  
  "arn:aws:ecs:us-west-2:123456789012:task-definition/sleep360:1",  
  "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:3",  
  "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:4",  
  "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:5",  
  "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:6"  
]
```

Ejemplo 2: Para enumerar las definiciones de tareas registradas en una familia

En el siguiente `list-task-definitions` ejemplo, se enumeran las revisiones de las definiciones de tareas de una familia específica.

```
aws ecs list-task-definitions --family-prefix wordpress
```

Salida:

```
{  
  "taskDefinitionArns": [  
    "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:3",  
    "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:4",  
    "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:5",  
    "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:6"  
  ]  
}
```

Para obtener más información, consulte [Definiciones de ECS tareas de Amazon](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [ListTaskDefinitions](#) la Referencia de AWS CLI comandos.

## list-tasks

En el siguiente ejemplo de código, se muestra cómo usar `list-tasks`.

### AWS CLI

Ejemplo 1: Creación de una lista de las tareas de un clúster

En el siguiente ejemplo de `list-tasks` se muestran todas las tareas de un clúster.

```
aws ecs list-tasks --cluster default
```

Salida:

```
{
  "taskArns": [
    "arn:aws:ecs:us-west-2:123456789012:task/a1b2c3d4-5678-90ab-
cdef-11111EXAMPLE",
    "arn:aws:ecs:us-west-2:123456789012:task/a1b2c3d4-5678-90ab-
cdef-22222EXAMPLE"
  ]
}
```

Ejemplo 2: Creación de una lista de las tareas de una instancia de contenedor concreta

En el siguiente `list-tasks` ejemplo, se enumeran las tareas de una instancia contenedora, utilizando la instancia contenedora UUID como filtro.

```
aws ecs list-tasks --cluster default --container-instance a1b2c3d4-5678-90ab-
cdef-33333EXAMPLE
```

Salida:

```
{
  "taskArns": [
    "arn:aws:ecs:us-west-2:123456789012:task/a1b2c3d4-5678-90ab-
cdef-44444EXAMPLE"
  ]
}
```

Para obtener más información, consulte [Definiciones de ECS tareas de Amazon](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [ListTasks](#) la Referencia de AWS CLI comandos.

## **put-account-setting-default**

En el siguiente ejemplo de código, se muestra cómo usar `put-account-setting-default`.



## AWS CLI

Para modificar la configuración de la cuenta por defecto

En el siguiente `put-account-setting-default` ejemplo, se modifica la configuración de cuenta predeterminada para todos IAM los usuarios o roles de la cuenta. Estos cambios se aplican a toda la AWS cuenta, a menos que un IAM usuario o rol anule explícitamente esta configuración por sí mismo.

```
aws ecs put-account-setting-default --name serviceLongArnFormat --value enabled
```

Salida:

```
{
  "setting": {
    "name": "serviceLongArnFormat",
    "value": "enabled",
    "principalArn": "arn:aws:iam::123456789012:root"
  }
}
```

Para obtener más información, consulte [Amazon Resource Names \(ARNs\) y IDs](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [PutAccountSettingDefault](#) la Referencia de AWS CLI comandos.

## `put-account-setting`

En el siguiente ejemplo de código, se muestra cómo usar `put-account-setting`.

### AWS CLI

Para modificar la configuración de su cuenta IAM de usuario

El siguiente `put-account-setting` ejemplo habilita la `serviceLongArnFormat` configuración de la cuenta IAM de usuario.

```
aws ecs put-account-setting --name serviceLongArnFormat --value enabled
```

Salida:

```
{
  "setting": {
    "name": "serviceLongArnFormat",
    "value": "enabled",
    "principalArn": "arn:aws:iam::130757420319:user/your_username"
  }
}
```

Para obtener más información, consulte [Modificación de la configuración de la cuenta](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [PutAccountSetting](#) la Referencia de AWS CLI comandos.

## put-account-settings

En el siguiente ejemplo de código, se muestra cómo usar `put-account-settings`.

AWS CLI

Para modificar la configuración de la cuenta de un IAM usuario o IAM rol

En el siguiente `put-account-setting` ejemplo, se modifica la configuración de la cuenta para el IAM usuario o IAM rol especificado.

```
aws ecs put-account-setting \
  --name serviceLongArnFormat \
  --value enabled \
  --principal-arn arn:aws:iam::123456789012:user/MyUser
```

Salida:

```
{
  "setting": {
    "name": "serviceLongArnFormat",
    "value": "enabled",
    "principalArn": "arn:aws:iam::123456789012:user/MyUser"
  }
}
```

- Para API obtener más información, consulte [PutAccountSettings](#) la Referencia de AWS CLI comandos.

## put-attributes

En el siguiente ejemplo de código, se muestra cómo usar `put-attributes`.

### AWS CLI

Para crear un atributo y asociarlo a un ECS recurso de Amazon

A continuación, `put-attributes` se aplica un atributo con la pila de nombres y el valor `production` a una instancia de contenedor.

```
aws ecs put-attributes \  
  --attributes name=stack,value=production,targetId=arn:aws:ecs:us-west-2:130757420319:container-instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34
```

Salida:

```
{  
  "attributes": [  
    {  
      "name": "stack",  
      "targetId": "arn:aws:ecs:us-west-2:130757420319:container-instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34",  
      "value": "production"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [PutAttributes](#) la Referencia de AWS CLI comandos.

## put-cluster-capacity-providers

En el siguiente ejemplo de código, se muestra cómo usar `put-cluster-capacity-providers`.

### AWS CLI

Ejemplo 1: Para añadir un proveedor de capacidad existente a un clúster

El siguiente `put-cluster-capacity-providers` ejemplo agrega un proveedor de capacidad existente a un clúster. El comando `create-capacity-provider` se utiliza para crear un proveedor de capacidad. El `describe-clusters` comando se utiliza para describir los proveedores de capacidad actuales y la estrategia del proveedor de capacidad predeterminado asociada a un clúster. Al agregar un nuevo proveedor de capacidad a un clúster, debe especificar todos los proveedores de capacidad existentes además del nuevo proveedor de capacidad que desee asociar al clúster. También debe especificar la estrategia del proveedor de capacidad predeterminado para asociarla al clúster. En este ejemplo, el `MyCluster` clúster tiene un proveedor de `MyCapacityProvider1` capacidad asociado y usted desea agregar el proveedor de `MyCapacityProvider2` capacidad e incluirlo en la estrategia de proveedor de capacidad predeterminada para que las tareas se distribuyan de manera uniforme entre ambos proveedores de capacidad.

```
aws ecs put-cluster-capacity-providers \
  --cluster MyCluster \
  --capacity-providers MyCapacityProvider1 MyCapacityProvider2 \
  --default-capacity-provider-
strategy capacityProvider=MyCapacityProvider1,weight=1 capacityProvider=MyCapacityProvider2,
```

Salida:

```
{
  "cluster": {
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "clusterName": "MyCluster",
    "status": "ACTIVE",
    "registeredContainerInstancesCount": 0,
    "runningTasksCount": 0,
    "pendingTasksCount": 0,
    "activeServicesCount": 0,
    "statistics": [],
    "tags": [],
    "settings": [
      {
        "name": "containerInsights",
        "value": "enabled"
      }
    ],
    "capacityProviders": [
      "MyCapacityProvider1",
      "MyCapacityProvider2"
    ]
  }
}
```

```
    ],
    "defaultCapacityProviderStrategy": [
      {
        "capacityProvider": "MyCapacityProvider1",
        "weight": 1,
        "base": 0
      },
      {
        "capacityProvider": "MyCapacityProvider2",
        "weight": 1,
        "base": 0
      }
    ],
    "attachments": [
      {
        "id": "0fb0c8f4-6edd-4de1-9b09-17e470ee1918",
        "type": "as_policy",
        "status": "ACTIVE",
        "details": [
          {
            "name": "capacityProviderName",
            "value": "MyCapacityProvider1"
          },
          {
            "name": "scalingPolicyName",
            "value": "ECManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111"
          }
        ]
      },
      {
        "id": "ae592060-2382-4663-9476-b015c685593c",
        "type": "as_policy",
        "status": "ACTIVE",
        "details": [
          {
            "name": "capacityProviderName",
            "value": "MyCapacityProvider2"
          },
          {
            "name": "scalingPolicyName",
            "value": "ECManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222"
          }
        ]
      }
    ]
  }
}
```

```

    ]
  }
],
"attachmentsStatus": "UPDATE_IN_PROGRESS"
}
}

```

Para obtener más información, consulte [Proveedores de capacidad de clúster](#) en la Guía para ECS desarrolladores de Amazon.

### Ejemplo 2: Eliminar un proveedor de capacidad de un clúster

El siguiente `put-cluster-capacity-providers` ejemplo elimina un proveedor de capacidad de un clúster. El `describe-clusters` comando se usa para describir los proveedores de capacidad actuales asociados a un clúster. Al eliminar un proveedor de capacidad de un clúster, debe especificar los proveedores de capacidad que desea que permanezcan asociados al clúster, así como la estrategia predeterminada del proveedor de capacidad para asociarlos al clúster. En este ejemplo, el clúster tiene asociados los proveedores de `MyCapacityProvider2` capacidad `MyCapacityProvider1` y, si desea eliminarlos, debe especificarlo únicamente `MyCapacityProvider1` en el comando junto con la estrategia actualizada del proveedor de capacidad predeterminado. `MyCapacityProvider2`

```

aws ecs put-cluster-capacity-providers \
  --cluster MyCluster \
  --capacity-providers MyCapacityProvider1 \
  --default-capacity-provider-
strategy capacityProvider=MyCapacityProvider1,weight=1,base=0

```

Salida:

```

{
  "cluster": {
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "clusterName": "MyCluster",
    "status": "ACTIVE",
    "registeredContainerInstancesCount": 0,
    "runningTasksCount": 0,
    "pendingTasksCount": 0,
    "activeServicesCount": 0,
    "statistics": [],
    "tags": [],

```

```
"settings": [
  {
    "name": "containerInsights",
    "value": "enabled"
  }
],
"capacityProviders": [
  "MyCapacityProvider1"
],
"defaultCapacityProviderStrategy": [
  {
    "capacityProvider": "MyCapacityProvider1",
    "weight": 1,
    "base": 0
  }
],
"attachments": [
  {
    "id": "0fb0c8f4-6edd-4de1-9b09-17e470ee1918",
    "type": "as_policy",
    "status": "ACTIVE",
    "details": [
      {
        "name": "capacityProviderName",
        "value": "MyCapacityProvider1"
      },
      {
        "name": "scalingPolicyName",
        "value": "ECManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111"
      }
    ]
  },
  {
    "id": "ae592060-2382-4663-9476-b015c685593c",
    "type": "as_policy",
    "status": "DELETING",
    "details": [
      {
        "name": "capacityProviderName",
        "value": "MyCapacityProvider2"
      },
      {
        "name": "scalingPolicyName",
        "value": "ECManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222"
      }
    ]
  }
]
```

```

        }
      ]
    }
  ],
  "attachmentsStatus": "UPDATE_IN_PROGRESS"
}
}

```

Para obtener más información, consulte [Proveedores de capacidad de clúster](#) en la Guía para ECS desarrolladores de Amazon.

Ejemplo 3: Eliminar todos los proveedores de capacidad de un clúster

El siguiente `put-cluster-capacity-providers` ejemplo elimina todos los proveedores de capacidad existentes del clúster.

```

aws ecs put-cluster-capacity-providers \
  --cluster MyCluster \
  --capacity-providers [] \
  --default-capacity-provider-strategy []

```

Salida:

```

{
  "cluster": {
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
    "clusterName": "MyCluster",
    "status": "ACTIVE",
    "registeredContainerInstancesCount": 0,
    "runningTasksCount": 0,
    "pendingTasksCount": 0,
    "activeServicesCount": 0,
    "statistics": [],
    "tags": [],
    "settings": [
      {
        "name": "containerInsights",
        "value": "enabled"
      }
    ],
    "capacityProviders": [],
    "defaultCapacityProviderStrategy": [],
  }
}

```



```

    "attachments": [
      {
        "id": "0fb0c8f4-6edd-4de1-9b09-17e470ee1918",
        "type": "as_policy",
        "status": "DELETING",
        "details": [
          {
            "name": "capacityProviderName",
            "value": "MyCapacityProvider1"
          },
          {
            "name": "scalingPolicyName",
            "value": "ECManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111"
          }
        ]
      },
      {
        "id": "ae592060-2382-4663-9476-b015c685593c",
        "type": "as_policy",
        "status": "DELETING",
        "details": [
          {
            "name": "capacityProviderName",
            "value": "MyCapacityProvider2"
          },
          {
            "name": "scalingPolicyName",
            "value": "ECManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222"
          }
        ]
      }
    ],
    "attachmentsStatus": "UPDATE_IN_PROGRESS"
  }
}

```

Para obtener más información, consulte [Proveedores de capacidad de clúster](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [PutClusterCapacityProviders](#) la Referencia de AWS CLI comandos.

## register-task-definition

En el siguiente ejemplo de código, se muestra cómo usar `register-task-definition`.

### AWS CLI

Ejemplo 1: Para registrar una definición de tarea con un JSON archivo

El siguiente `register-task-definition` ejemplo registra una definición de tarea en la familia especificada. Las definiciones de contenedor se guardan en JSON formato en la ubicación del archivo especificada.

```
aws ecs register-task-definition \  
  --cli-input-json file://<path_to_json_file>/sleep360.json
```

Contenidos de `sleep360.json`:

```
{  
  "containerDefinitions": [  
    {  
      "name": "sleep",  
      "image": "busybox",  
      "cpu": 10,  
      "command": [  
        "sleep",  
        "360"  
      ],  
      "memory": 10,  
      "essential": true  
    }  
  ],  
  "family": "sleep360"  
}
```

Salida:

```
{  
  "taskDefinition": {  
    "status": "ACTIVE",  
    "family": "sleep360",  
    "placementConstraints": [],  
    "compatibilities": [  

```

```

        "EXTERNAL",
        "EC2"
    ],
    "volumes": [],
    "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-definition/
sleep360:1",
    "containerDefinitions": [
        {
            "environment": [],
            "name": "sleep",
            "mountPoints": [],
            "image": "busybox",
            "cpu": 10,
            "portMappings": [],
            "command": [
                "sleep",
                "360"
            ],
            "memory": 10,
            "essential": true,
            "volumesFrom": []
        }
    ],
    "revision": 1
}

```

Para obtener más información, consulte [Ejemplos de definiciones de tareas](#) en la Guía para ECS desarrolladores de Amazon.

Ejemplo 2: Para registrar una definición de tarea con un parámetro de JSON cadena

El siguiente `register-task-definition` ejemplo registra una definición de tarea mediante las definiciones de contenedor proporcionadas como un parámetro de JSON cadena con comillas dobles de escape.

```

aws ecs register-task-definition \
  --family sleep360 \
  --container-definitions "[{\\"name\\":\\"sleep\\",\\"image\\":\\"busybox\\",\\"cpu\\":10,
\\\"command\\":[\\"sleep\\",\\"360\\"],\\"memory\\":10,\\"essential\\":true}]"

```

El resultado es idéntico al ejemplo anterior.

Para obtener más información, consulte [Creación de una definición de tarea](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [RegisterTaskDefinition](#) la Referencia de AWS CLI comandos.

## run-task

En el siguiente ejemplo de código, se muestra cómo usar `run-task`.

### AWS CLI

Para ejecutar una tarea en el clúster predeterminado

En el siguiente `run-task` ejemplo, se ejecuta una tarea en el clúster predeterminado y se utiliza un token de cliente.

```
aws ecs run-task \  
  --cluster default \  
  --task-definition sleep360:1 \  
  --client-token 550e8400-e29b-41d4-a716-446655440000
```

Salida:

```
{  
  "tasks": [  
    {  
      "attachments": [],  
      "attributes": [  
        {  
          "name": "ecs.cpu-architecture",  
          "value": "x86_64"  
        }  
      ],  
      "availabilityZone": "us-east-1b",  
      "capacityProviderName": "example-capacity-provider",  
      "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/default",  
      "containerInstanceArn": "arn:aws:ecs:us-east-1:123456789012:container-instance/default/bc4d2ec611d04bb7bb97e83ceEXAMPLE",  
      "containers": [  
        {  
          "containerArn": "arn:aws:ecs:us-east-1:123456789012:container/default/d6f51cc5bbc94a47969c92035e9f66f8/75853d2d-711e-458a-8362-0f0aEXAMPLE",
```

```

        "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/default/
d6f51cc5bbc94a47969c9203EXAMPLE",
        "name": "sleep",
        "image": "busybox",
        "lastStatus": "PENDING",
        "networkInterfaces": [],
        "cpu": "10",
        "memory": "10"
    }
],
"cpu": "10",
"createdAt": "2023-11-21T16:59:34.403000-05:00",
"desiredStatus": "RUNNING",
"enableExecuteCommand": false,
"group": "family:sleep360",
"lastStatus": "PENDING",
"launchType": "EC2",
"memory": "10",
"overrides": {
    "containerOverrides": [
        {
            "name": "sleep"
        }
    ],
    "inferenceAcceleratorOverrides": []
},
"tags": [],
"taskArn": "arn:aws:ecs:us-east-1:123456789012:task/default/
d6f51cc5bbc94a47969c9203EXAMPLE",
"taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-
definition/sleep360:1",
"version": 1
}
],
"failures": []
}

```

Para obtener más información, consulte [Ejecución de tareas](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [RunTask](#) la Referencia de AWS CLI comandos.

## start-task

En el siguiente ejemplo de código, se muestra cómo usar `start-task`.

### AWS CLI

Para iniciar una nueva tarea

A continuación, se `start-task` inicia una tarea con la última revisión de la definición de la `sleep360` tarea en la instancia de contenedor especificada en el clúster predeterminado.

```
aws ecs start-task \  
  --task-definition sleep360 \  
  --container-instances 765936fadbdd46b5991a4bd70c2a43d4
```

Salida:

```
{  
  "tasks": [  
    {  
      "taskArn": "arn:aws:ecs:us-west-2:130757420319:task/  
default/666fdccc2e2d4b6894dd422f4eeee8f8",  
      "clusterArn": "arn:aws:ecs:us-west-2:130757420319:cluster/default",  
      "taskDefinitionArn": "arn:aws:ecs:us-west-2:130757420319:task-  
definition/sleep360:3",  
      "containerInstanceArn": "arn:aws:ecs:us-west-2:130757420319:container-  
instance/default/765936fadbdd46b5991a4bd70c2a43d4",  
      "overrides": {  
        "containerOverrides": [  
          {  
            "name": "sleep"  
          }  
        ]  
      },  
      "lastStatus": "PENDING",  
      "desiredStatus": "RUNNING",  
      "cpu": "128",  
      "memory": "128",  
      "containers": [  
        {  
          "containerArn": "arn:aws:ecs:us-  
west-2:130757420319:container/75f11ed4-8a3d-4f26-a33b-ad1db9e02d41",
```

```

        "taskArn": "arn:aws:ecs:us-west-2:130757420319:task/
default/666fdccc2e2d4b6894dd422f4eeee8f8",
        "name": "sleep",
        "lastStatus": "PENDING",
        "networkInterfaces": [],
        "cpu": "10",
        "memory": "10"
    }
],
"version": 1,
"createdAt": 1563421494.186,
"group": "family:sleep360",
"launchType": "EC2",
"attachments": [],
"tags": []
}
],
"failures": []
}

```

- Para API obtener más información, consulte [StartTask](#) la Referencia de AWS CLI comandos.

## stop-task

En el siguiente ejemplo de código, se muestra cómo usar stop-task.

### AWS CLI

Para detener una tarea

Lo siguiente stop-task impide que la tarea especificada se ejecute en el clúster predeterminado.

```

aws ecs stop-task \
  --task 666fdccc2e2d4b6894dd422f4eeee8f8

```

Salida:

```

{
  "task": {
    "taskArn": "arn:aws:ecs:us-west-2:130757420319:task/
default/666fdccc2e2d4b6894dd422f4eeee8f8",

```

```
    "clusterArn": "arn:aws:ecs:us-west-2:130757420319:cluster/default",
    "taskDefinitionArn": "arn:aws:ecs:us-west-2:130757420319:task-definition/
sleep360:3",
    "containerInstanceArn": "arn:aws:ecs:us-west-2:130757420319:container-
instance/default/765936fadbdd46b5991a4bd70c2a43d4",
    "overrides": {
      "containerOverrides": []
    },
    "lastStatus": "STOPPED",
    "desiredStatus": "STOPPED",
    "cpu": "128",
    "memory": "128",
    "containers": [],
    "version": 2,
    "stoppedReason": "Taskfailedtostart",
    "stopCode": "TaskFailedToStart",
    "connectivity": "CONNECTED",
    "connectivityAt": 1563421494.186,
    "pullStartedAt": 1563421494.252,
    "pullStoppedAt": 1563421496.252,
    "executionStoppedAt": 1563421497,
    "createdAt": 1563421494.186,
    "stoppingAt": 1563421497.252,
    "stoppedAt": 1563421497.252,
    "group": "family:sleep360",
    "launchType": "EC2",
    "attachments": [],
    "tags": []
  }
}
```

- Para API obtener más información, consulte [StopTask](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

### AWS CLI

Para etiquetar un recurso

En el siguiente tag-resource ejemplo, se agrega una sola etiqueta al recurso especificado.



```
aws ecs tag-resource \  
  --resource-arn arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster \  
  --tags key=key1,value=value1
```

Este comando no genera ninguna salida.

Para agregar varias etiquetas a un recurso

En el siguiente `tag-resource` ejemplo, se agregan varias etiquetas al recurso especificado.

```
aws ecs tag-resource \  
  --resource-arn arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster \  
  --tags key=key1,value=value1 key=key2,value=value2 key=key3,value=value3
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar una etiqueta de un recurso

En el siguiente `untag-resource` ejemplo, se quitan las etiquetas de la lista del recurso especificado.

```
aws ecs untag-resource \  
  --resource-arn arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster \  
  --tag-keys key1,key2
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-cluster-settings

En el siguiente ejemplo de código, se muestra cómo usar `update-cluster-settings`.

## AWS CLI

Para modificar la configuración del clúster

El siguiente `update-cluster-settings` ejemplo habilita CloudWatch Container Insights para el `default` clúster.

```
aws ecs update-cluster-settings \  
  --cluster default \  
  --settings name=containerInsights,value=enabled
```

Salida:

```
{  
  "cluster": {  
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",  
    "clusterName": "default",  
    "status": "ACTIVE",  
    "registeredContainerInstancesCount": 0,  
    "runningTasksCount": 0,  
    "pendingTasksCount": 0,  
    "activeServicesCount": 0,  
    "statistics": [],  
    "tags": [],  
    "settings": [  
      {  
        "name": "containerInsights",  
        "value": "enabled"  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [Modificación de la configuración de la cuenta](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [UpdateClusterSettings](#) la Referencia de AWS CLI comandos.

## update-container-agent

En el siguiente ejemplo de código, se muestra cómo usar `update-container-agent`.

## AWS CLI

Para actualizar el agente de contenedor en una instancia de ECS contenedor de Amazon

En el siguiente `update-container-agent` ejemplo, se actualiza el agente contenedor en la instancia contenedora especificada del clúster predeterminado.

```
aws ecs update-container-agent --cluster default --container-  
instance a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Salida:

```
{  
  "containerInstance": {  
    "status": "ACTIVE",  
    ...  
    "agentUpdateStatus": "PENDING",  
    "versionInfo": {  
      "agentVersion": "1.0.0",  
      "agentHash": "4023248",  
      "dockerVersion": "DockerVersion: 1.5.0"  
    }  
  }  
}
```

Para obtener más información, consulta [Actualización del Amazon ECS Container Agent](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [UpdateContainerAgent](#) la Referencia de AWS CLI comandos.

## `update-container-instances-state`

En el siguiente ejemplo de código, se muestra cómo usar `update-container-instances-state`.

## AWS CLI

Para actualizar el estado de una instancia contenedora

A continuación, se `update-container-instances-state` actualiza el estado de la instancia contenedora especificada, en la DRAINING que se eliminará del clúster en el que está registrada.

```
aws ecs update-container-instances-state \  
  --container-instances 765936fadbdd46b5991a4bd70c2a43d4 \  
  --status DRAINING
```

Salida:

```
{  
  "containerInstances": [  
    {  
      "containerInstanceArn": "arn:aws:ecs:us-west-2:130757420319:container-  
instance/default/765936fadbdd46b5991a4bd70c2a43d4",  
      "ec2InstanceId": "i-013d87ffbb4d513bf",  
      "version": 4390,  
      "versionInfo": {  
        "agentVersion": "1.29.0",  
        "agentHash": "a190a73f",  
        "dockerVersion": "DockerVersion:18.06.1-ce"  
      },  
      "remainingResources": [  
        {  
          "name": "CPU",  
          "type": "INTEGER",  
          "doubleValue": 0,  
          "longValue": 0,  
          "integerValue": 1536  
        },  
        {  
          "name": "MEMORY",  
          "type": "INTEGER",  
          "doubleValue": 0,  
          "longValue": 0,  
          "integerValue": 2681  
        },  
        {  
          "name": "PORTS",  
          "type": "STRINGSET",  
          "doubleValue": 0,  
          "longValue": 0,  
          "integerValue": 0,  
          "stringSetValue": [  
            "22",  
            "2376",  
            "2375",  
          ]  
        }  
      ]  
    }  
  ]  
}
```

```
        "51678",
        "51679"
    ]
},
{
    "name": "PORTS_UDP",
    "type": "STRINGSET",
    "doubleValue": 0,
    "longValue": 0,
    "integerValue": 0,
    "stringSetValue": []
}
],
"registeredResources": [
    {
        "name": "CPU",
        "type": "INTEGER",
        "doubleValue": 0,
        "longValue": 0,
        "integerValue": 2048
    },
    {
        "name": "MEMORY",
        "type": "INTEGER",
        "doubleValue": 0,
        "longValue": 0,
        "integerValue": 3705
    },
    {
        "name": "PORTS",
        "type": "STRINGSET",
        "doubleValue": 0,
        "longValue": 0,
        "integerValue": 0,
        "stringSetValue": [
            "22",
            "2376",
            "2375",
            "51678",
            "51679"
        ]
    }
},
{
    "name": "PORTS_UDP",
```

```
        "type": "STRINGSET",
        "doubleValue": 0,
        "longValue": 0,
        "integerValue": 0,
        "stringSetValue": []
    }
],
"status": "DRAINING",
"agentConnected": true,
"runningTasksCount": 2,
"pendingTasksCount": 0,
"attributes": [
    {
        "name": "ecs.capability.secrets.asm.environment-variables"
    },
    {
        "name": "ecs.capability.branch-cni-plugin-version",
        "value": "e0703516-"
    },
    {
        "name": "ecs.ami-id",
        "value": "ami-00e0090ac21971297"
    },
    {
        "name": "ecs.capability.secrets.asm.bootstrap.log-driver"
    },
    {
        "name": "com.amazonaws.ecs.capability.logging-driver.none"
    },
    {
        "name": "ecs.capability.ecr-endpoint"
    },
    {
        "name": "ecs.capability.docker-plugin.local"
    },
    {
        "name": "ecs.capability.task-cpu-mem-limit"
    },
    {
        "name": "ecs.capability.secrets.ssm.bootstrap.log-driver"
    },
    {
        "name": "com.amazonaws.ecs.capability.docker-remote-api.1.30"
    },
],
```

```
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.31"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.32"
},
{
  "name": "ecs.availability-zone",
  "value": "us-west-2c"
},
{
  "name": "ecs.capability.aws-appmesh"
},
{
  "name": "com.amazonaws.ecs.capability.logging-driver.awslogs"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.24"
},
{
  "name": "ecs.capability.task-eni-trunking"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.25"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.26"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.27"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.28"
},
{
  "name": "com.amazonaws.ecs.capability.privileged-container"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.29"
},
{
  "name": "ecs.cpu-architecture",
  "value": "x86_64"
},
},
```

```
{
  "name": "com.amazonaws.ecs.capability.ecr-auth"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.20"
},
{
  "name": "ecs.os-type",
  "value": "linux"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.21"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.22"
},
{
  "name": "ecs.capability.task-eia"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.23"
},
{
  "name": "ecs.capability.private-registry-
authentication.secretsmanager"
},
{
  "name": "com.amazonaws.ecs.capability.logging-driver.syslog"
},
{
  "name": "com.amazonaws.ecs.capability.logging-driver.json-file"
},
{
  "name": "ecs.capability.execution-role-awslogs"
},
{
  "name": "ecs.vpc-id",
  "value": "vpc-1234"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.17"
},
{
  "name": "com.amazonaws.ecs.capability.docker-remote-api.1.18"
```



```
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.19"
    },
    {
      "name": "ecs.capability.task-eni"
    },
    {
      "name": "ecs.capability.execution-role-ecr-pull"
    },
    {
      "name": "ecs.capability.container-health-check"
    },
    {
      "name": "ecs.subnet-id",
      "value": "subnet-1234"
    },
    {
      "name": "ecs.instance-type",
      "value": "c5.large"
    },
    {
      "name": "com.amazonaws.ecs.capability.task-iam-role-network-
host"
    },
    {
      "name": "ecs.capability.container-ordering"
    },
    {
      "name": "ecs.capability.cni-plugin-version",
      "value": "91ccefc8-2019.06.0"
    },
    {
      "name": "ecs.capability.pid-ipc-namespace-sharing"
    },
    {
      "name": "ecs.capability.secrets.ssm.environment-variables"
    },
    {
      "name": "com.amazonaws.ecs.capability.task-iam-role"
    }
  ],
  "registeredAt": 1560788724.507,
  "attachments": [],
```

```
        "tags": []
      }
    ],
    "failures": []
  }
}
```

- Para API obtener más información, consulte [UpdateContainerInstancesState](#) la Referencia de AWS CLI comandos.

## update-service-primary-task-set

En el siguiente ejemplo de código, se muestra cómo usar `update-service-primary-task-set`.

### AWS CLI

Para actualizar el conjunto de tareas principal de un servicio

El siguiente `update-service-primary-task-set` ejemplo actualiza el conjunto de tareas principal del servicio especificado.

```
aws ecs update-service-primary-task-set \
  --cluster MyCluster \
  --service MyService \
  --primary-task-set arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/MyService/ecs-svc/1234567890123456789
```

Salida:

```
{
  "taskSet": {
    "id": "ecs-svc/1234567890123456789",
    "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/MyService/ecs-svc/1234567890123456789",
    "status": "PRIMARY",
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/sample-fargate:2",
    "computedDesiredCount": 1,
    "pendingCount": 0,
    "runningCount": 0,
    "createdAt": 1557128360.711,
    "updatedAt": 1557129412.653,
    "launchType": "EC2",
  }
}
```

```

    "networkConfiguration": {
      "awsvpcConfiguration": {
        "subnets": [
          "subnet-12344321"
        ],
        "securityGroups": [
          "sg-12344312"
        ],
        "assignPublicIp": "DISABLED"
      }
    },
    "loadBalancers": [],
    "serviceRegistries": [],
    "scale": {
      "value": 50.0,
      "unit": "PERCENT"
    },
    "stabilityStatus": "STABILIZING",
    "stabilityStatusAt": 1557129279.914
  }
}

```

- Para API obtener más información, consulte [UpdateServicePrimaryTaskSet](#) la Referencia de AWS CLI comandos.

## update-service

En el siguiente ejemplo de código, se muestra cómo usar `update-service`.

### AWS CLI

Ejemplo 1: Cambio de la definición de tarea utilizada en un servicio

En el siguiente ejemplo de `update-service` se actualiza el servicio `my-http-service` para usar la definición de la tarea `amazon-ecs-sample`.

```
aws ecs update-service --service my-http-service --task-definition amazon-ecs-sample
```

Ejemplo 2: Cambio del número de tareas de un servicio

En el siguiente ejemplo de `update-service` se actualiza el recuento de tareas deseado del servicio `my-http-service` a 3.

```
aws ecs update-service --service my-http-service --desired-count 3
```

Para obtener más información, consulta [Actualización de un servicio](#) en la Guía para ECS desarrolladores de Amazon.

- Para API obtener más información, consulte [UpdateService](#) la Referencia de AWS CLI comandos.

## update-task-set

En el siguiente ejemplo de código, se muestra cómo usar `update-task-set`.

### AWS CLI

Para actualizar un conjunto de tareas

En el siguiente `update-task-set` ejemplo, se actualiza un conjunto de tareas para ajustar la escala.

```
aws ecs update-task-set \  
  --cluster MyCluster \  
  --service MyService \  
  --task-set arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/MyService/ecs-  
svc/1234567890123456789 \  
  --scale value=50,unit=PERCENT
```

Salida:

```
{  
  "taskSet": {  
    "id": "ecs-svc/1234567890123456789",  
    "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/  
MyService/ecs-svc/1234567890123456789",  
    "status": "ACTIVE",  
    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/  
sample-fargate:2",  
    "computedDesiredCount": 0,  
    "pendingCount": 0,  
    "runningCount": 0,  
    "createdAt": 1557128360.711,  
    "updatedAt": 1557129279.914,
```

```
    "launchType": "EC2",
    "networkConfiguration": {
      "awsVpcConfiguration": {
        "subnets": [
          "subnet-12344321"
        ],
        "securityGroups": [
          "sg-12344321"
        ],
        "assignPublicIp": "DISABLED"
      }
    },
    "loadBalancers": [],
    "serviceRegistries": [],
    "scale": {
      "value": 50.0,
      "unit": "PERCENT"
    },
    "stabilityStatus": "STABILIZING",
    "stabilityStatusAt": 1557129279.914
  }
}
```

- Para API obtener más información, consulte [UpdateTaskSet](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon que utilizan AWS CLI

En los siguientes ejemplos de código, se muestra cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con AmazonEFS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### create-file-system

En el siguiente ejemplo de código, se muestra cómo usar `create-file-system`.

#### AWS CLI

Para crear un sistema de archivos cifrados

En el siguiente `create-file-system` ejemplo, se crea un sistema de archivos cifrados con el valor predeterminado `CMK`. También añade la etiqueta `Name=my-file-system`.

```
aws efs create-file-system \  
  --performance-mode generalPurpose \  
  --throughput-mode bursting \  
  --encrypted \  
  --tags Key=Name,Value=my-file-system
```

Salida:

```
{  
  "OwnerId": "123456789012",  
  "CreationToken": "console-d7f56c5f-e433-41ca-8307-9d9c0example",  
  "FileSystemId": "fs-c7a0456e",  
  "FileSystemArn": "arn:aws:elasticfilesystem:us-west-2:123456789012:file-system/  
fs-48499b4d",  
  "CreationTime": 1595286880.0,  
  "LifecycleState": "creating",  
  "Name": "my-file-system",  
  "NumberOfMountTargets": 0,  
  "SizeInBytes": {  
    "Value": 0,  
    "ValueInIA": 0,  
    "ValueInStandard": 0  
  },  
  "PerformanceMode": "generalPurpose",  
  "Encrypted": true,  
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/a59b3472-e62c-42e4-  
adcf-30d92example",  
  "ThroughputMode": "bursting",  
  "Tags": [  
    {  
      "Key": "Name",  
      "Value": "my-file-system"  
    }  
  ]  
}
```

```
{
  "Key": "Name",
  "Value": "my-file-system"
}
]
```

Para obtener más información, consulte [Creación de sistemas de EFS archivos de Amazon](#) en la Guía del usuario de Amazon Elastic File System.

- Para API obtener más información, consulte [CreateFileSystem](#) la Referencia de AWS CLI comandos.

## create-mount-target

En el siguiente ejemplo de código, se muestra cómo usar `create-mount-target`.

### AWS CLI

Para crear un objetivo de montaje

El siguiente `create-mount-target` ejemplo crea un destino de montaje para el sistema de archivos especificado.

```
aws efs create-mount-target \
  --file-system-id fs-c7a0456e \
  --subnet-id subnet-02bf4c428bexample \
  --security-groups sg-068f739363example
```

Salida:

```
{
  "OwnerId": "123456789012",
  "MountTargetId": "fsmt-f9a14450",
  "FileSystemId": "fs-c7a0456e",
  "SubnetId": "subnet-02bf4c428bexample",
  "LifecycleState": "creating",
  "IpAddress": "10.0.1.24",
  "NetworkInterfaceId": "eni-02d542216aexample",
  "AvailabilityZoneId": "use2-az2",
  "AvailabilityZoneName": "us-east-2b",
  "VpcId": "vpc-0123456789abcdef0"
```

```
}
```

Para obtener más información, consulte [Creación de objetivos de montaje](#) en la Guía del usuario de Amazon Elastic File System.

- Para API obtener más información, consulte [CreateMountTarget](#) la Referencia de AWS CLI comandos.

## **delete-file-system**

En el siguiente ejemplo de código, se muestra cómo usar `delete-file-system`.

### AWS CLI

Para eliminar un sistema de archivos

En el siguiente `delete-file-system` ejemplo, se elimina el sistema de archivos especificado.

```
aws efs delete-file-system \  
  --file-system-id fs-c7a0456e
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un sistema de EFS archivos de Amazon](#) en la Guía del usuario de Amazon Elastic File System.

- Para API obtener más información, consulte [DeleteFileSystem](#) la Referencia de AWS CLI comandos.

## **delete-mount-target**

En el siguiente ejemplo de código, se muestra cómo usar `delete-mount-target`.

### AWS CLI

Para eliminar un objetivo de montaje

En el siguiente `delete-mount-target` ejemplo, se elimina el objetivo de montaje especificado.

```
aws efs delete-mount-target \  
  --mount-target-id fsmt-f9a14450
```



Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de objetivos de montaje](#) en la Guía del usuario de Amazon Elastic File System.

- Para API obtener más información, consulte [DeleteMountTarget](#) la Referencia de AWS CLI comandos.

## describe-file-systems

En el siguiente ejemplo de código, se muestra cómo usar `describe-file-systems`.

### AWS CLI

Para describir un sistema de archivos

El siguiente `describe-file-systems` ejemplo describe el sistema de archivos especificado.

```
aws efs describe-file-systems \  
  --file-system-id fs-c7a0456e
```

Salida:

```
{  
  "FileSystems": [  
    {  
      "OwnerId": "123456789012",  
      "CreationToken": "console-d7f56c5f-e433-41ca-8307-9d9c0example",  
      "FileSystemId": "fs-c7a0456e",  
      "FileSystemArn": "arn:aws:elasticfilesystem:us-west-2:123456789012:file-  
system/fs-48499b4d",  
      "CreationTime": 1595286880.0,  
      "LifecycleState": "available",  
      "Name": "my-file-system",  
      "NumberOfMountTargets": 3,  
      "SizeInBytes": {  
        "Value": 6144,  
        "Timestamp": 1600991437.0,  
        "ValueInIA": 0,  
        "ValueInStandard": 6144  
      },  
      "PerformanceMode": "generalPurpose",  
      "Encrypted": true,  
    }  
  ]  
}
```

```

        "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/a59b3472-e62c-42e4-
adcf-30d92example",
        "ThroughputMode": "bursting",
        "Tags": [
            {
                "Key": "Name",
                "Value": "my-file-system"
            }
        ]
    }
]
}

```

Para obtener más información, consulte [Gestión de los sistemas de EFS archivos](#) de Amazon en la Guía del usuario de Amazon Elastic File System.

- Para API obtener más información, consulte [DescribeFileSystems](#) la Referencia de AWS CLI comandos.

## describe-mount-targets

En el siguiente ejemplo de código, se muestra cómo usar `describe-mount-targets`.

### AWS CLI

Para describir un objetivo de montaje

El siguiente `describe-mount-targets` ejemplo describe el objetivo de montaje especificado.

```

aws efs describe-mount-targets \
  --mount-target-id fsmt-f9a14450

```

Salida:

```

{
  "MountTargets": [
    {
      "OwnerId": "123456789012",
      "MountTargetId": "fsmt-f9a14450",
      "FileSystemId": "fs-c7a0456e",
      "SubnetId": "subnet-02bf4c428bexample",
      "LifeCycleState": "creating",
    }
  ]
}

```

```
        "IpAddress": "10.0.1.24",
        "NetworkInterfaceId": "eni-02d542216aexample",
        "AvailabilityZoneId": "use2-az2",
        "AvailabilityZoneName": "us-east-2b",
        "VpcId": "vpc-0123456789abcdef0"
    }
]
}
```

Para obtener más información, consulte [Creación de objetivos de montaje](#) en la Guía del usuario de Amazon Elastic File System.

- Para API obtener más información, consulte [DescribeMountTargets](#) la Referencia de AWS CLI comandos.

## describe-tags

En el siguiente ejemplo de código, se muestra cómo usar `describe-tags`.

### AWS CLI

Para describir las etiquetas de un sistema de archivos

En el siguiente `describe-tags` ejemplo, se describen las etiquetas del sistema de archivos especificado.

```
aws efs describe-tags \
  --file-system-id fs-c7a0456e
```

Salida:

```
{
  "Tags": [
    {
      "Key": "Name",
      "Value": "my-file-system"
    },
    {
      "Key": "Department",
      "Value": "Business Intelligence"
    }
  ]
}
```

```
}
```

Para obtener más información, consulte [Gestión de etiquetas del sistema de archivos](#) en la Guía del usuario de Amazon Elastic File System.

- Para API obtener más información, consulte [DescribeTags](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para recuperar las etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se recuperan las etiquetas asociadas al sistema de archivos especificado.

```
aws efs list-tags-for-resource \  
  --resource-id fs-c7a0456e
```

Salida:

```
{  
  "Tags": [  
    {  
      "Key": "Name",  
      "Value": "my-file-system"  
    },  
    {  
      "Key": "Department",  
      "Value": "Business Intelligence"  
    }  
  ]  
}
```

Para obtener más información, consulte [Gestión de etiquetas del sistema de archivos](#) en la Guía del usuario de Amazon Elastic File System.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar un recurso

En el siguiente `tag-resource` ejemplo, se agrega la etiqueta `Department=Business Intelligence` al sistema de archivos especificado.

```
aws efs tag-resource \  
  --resource-id fs-c7a0456e \  
  --tags Key=Department,Value="Business Intelligence"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión de etiquetas del sistema de archivos](#) en la Guía del usuario de Amazon Elastic File System.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar una etiqueta de un recurso

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta con la clave de `Department` etiqueta del sistema de archivos especificado.

```
aws efs untag-resource \  
  --resource-id fs-c7a0456e \  
  --tag-keys Department
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión de etiquetas del sistema de archivos](#) en la Guía del usuario de Amazon Elastic File System.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## EKS Ejemplos de Amazon que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante AmazonEKS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **associate-encryption-config**

En el siguiente ejemplo de código, se muestra cómo usar `associate-encryption-config`.

AWS CLI

Para asociar una configuración de cifrado a un clúster existente

El siguiente `associate-encryption-config` ejemplo habilita el cifrado en un EKS clúster existente que aún no tiene el cifrado activado.

```
aws eks associate-encryption-config \  
  --cluster-name my-eks-cluster \  
  --encryption-config '[{"resources":["secrets"],"provider":  
{"keyArn":"arn:aws:kms:region-code:account:key/key"}]'
```

Salida:

```
{
```

```

"update": {
  "id": "3141b835-8103-423a-8e68-12c2521ffa4d",
  "status": "InProgress",
  "type": "AssociateEncryptionConfig",
  "params": [
    {
      "type": "EncryptionConfig",
      "value": "[{\"resources\":[\"secrets\"],\"provider\":{\"keyArn\":
\\\"arn:aws:kms:region-code:account:key/key\\\"}]}]"
    }
  ],
  "createdAt": "2024-03-14T11:01:26.297000-04:00",
  "errors": []
}
}

```

Para obtener más información, consulta Cómo [habilitar el cifrado secreto en un clúster existente](#) en la Guía del EKS usuario de Amazon.

- Para API obtener más información, consulte [AssociateEncryptionConfig](#) la Referencia de AWS CLI comandos.

## associate-identity-provider-config

En el siguiente ejemplo de código, se muestra cómo usar `associate-identity-provider-config`.

### AWS CLI

Asocia el proveedor de identidad a tu Amazon EKS Cluster

El siguiente `associate-identity-provider-config` ejemplo asocia un proveedor de identidad a tu Amazon EKS Cluster.

```

aws eks associate-identity-provider-config \
  --cluster-name my-eks-cluster \
  --oidc 'identityProviderConfigName=my-identity-provider,issuerUrl=https://
oidc.eks.us-east-2.amazonaws.com/
id/38D6A4619A0A69E342B113ED7F1A7652,clientId=kubernetes,usernameClaim=email,usernamePrefix=
username-prefix,groupsClaim=my-claim,groupsPrefix=my-groups-
prefix,requiredClaims={Claim1=value1,Claim2=value2}' \
  --tags env=dev

```

**Salida:**

```
{
  "update": {
    "id": "8c6c1bef-61fe-42ac-a242-89412387b8e7",
    "status": "InProgress",
    "type": "AssociateIdentityProviderConfig",
    "params": [
      {
        "type": "IdentityProviderConfig",
        "value": "[{\"type\": \"oidc\", \"name\": \"my-identity-provider\"}]"
      }
    ],
    "createdAt": "2024-04-11T13:46:49.648000-04:00",
    "errors": []
  },
  "tags": {
    "env": "dev"
  }
}
```

Para obtener más información, consulte [Autenticar usuarios de su clúster desde un proveedor de identidad de OpenID Connect: OIDC asociar un proveedor](#) de identidad en la Guía del usuario de EKS Amazon.

- Para API obtener más información, consulte la Referencia [AssociateIdentityProviderConfig](#) de AWS CLI comandos.

**create-addon**

En el siguiente ejemplo de código, se muestra cómo usar `create-addon`.

**AWS CLI**

Ejemplo 1: Para crear un EKS complemento de Amazon con una versión compatible predeterminada para la versión del EKS clúster correspondiente

El siguiente comando de `create-addon` ejemplo crea un EKS complemento de Amazon con una versión compatible predeterminada para la versión del EKS clúster correspondiente.

```
aws eks create-addon \
  --cluster-name my-eks-cluster \
```



```
--addon-name my-eks-addon \  
--service-account-role-arn arn:aws:iam::111122223333:role/role-name
```

Salida:

```
{  
  "addon": {  
    "addonName": "my-eks-addon",  
    "clusterName": "my-eks-cluster",  
    "status": "CREATING",  
    "addonVersion": "v1.15.1-eksbuild.1",  
    "health": {  
      "issues": []  
    },  
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-  
addon/1ec71ee1-b9c2-8915-4e17-e8be0a55a149",  
    "createdAt": "2024-03-14T12:20:03.264000-04:00",  
    "modifiedAt": "2024-03-14T12:20:03.283000-04:00",  
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",  
    "tags": {}  
  }  
}
```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: creación de un complemento](#) en la Guía del EKS usuario de Amazon.

Ejemplo 2: Para crear un EKS complemento de Amazon con una versión de complemento específica

El siguiente comando de `create-addon` ejemplo crea un EKS complemento de Amazon con una versión de complemento específica.

```
aws eks create-addon \  
  --cluster-name my-eks-cluster \  
  --addon-name my-eks-addon \  
  --service-account-role-arn arn:aws:iam::111122223333:role/role-name \  
  --addon-version v1.16.4-eksbuild.2
```

Salida:

```
{
```

```

"addon": {
  "addonName": "my-eks-addon",
  "clusterName": "my-eks-cluster",
  "status": "CREATING",
  "addonVersion": "v1.16.4-eksbuild.2",
  "health": {
    "issues": []
  },
  "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-
addon/34c71ee6-7738-6c8b-c6bd-3921a176b5ff",
  "createdAt": "2024-03-14T12:30:24.507000-04:00",
  "modifiedAt": "2024-03-14T12:30:24.521000-04:00",
  "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
  "tags": {}
}
}

```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: creación de un complemento](#) en la Guía del EKS usuario de Amazon.

Ejemplo 3: Para crear un EKS complemento de Amazon con valores de configuración personalizados y resolver conflictos: detalles

El siguiente comando de `create-addon` ejemplo crea un EKS complemento de Amazon con valores de configuración personalizados y resuelve los detalles de los conflictos.

```

aws eks create-addon \
  --cluster-name my-eks-cluster \
  --addon-name my-eks-addon \
  --service-account-role-arn arn:aws:iam::111122223333:role/role-name \
  --addon-version v1.16.4-eksbuild.2 \
  --configuration-values '{"resources":{"limits":{"cpu":"100m"}}}' \
  --resolve-conflicts OVERWRITE

```

Salida:

```

{
  "addon": {
    "addonName": "my-eks-addon",
    "clusterName": "my-eks-cluster",
    "status": "CREATING",
    "addonVersion": "v1.16.4-eksbuild.2",

```

```

    "health": {
      "issues": []
    },
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-
addon/a6c71ee9-0304-9237-1be8-25af1b0f1ffb",
    "createdAt": "2024-03-14T12:35:58.313000-04:00",
    "modifiedAt": "2024-03-14T12:35:58.327000-04:00",
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "tags": {},
    "configurationValues": "{\"resources\":{\"limits\":{\"cpu\":\"100m\"}}}"
  }
}

```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: creación de un complemento](#) en la Guía del EKS usuario de Amazon.

Ejemplo 4: Para crear un EKS complemento de Amazon con un archivo de valores JSON de configuración personalizados

El siguiente comando de `create-addon` ejemplo crea un EKS complemento de Amazon con valores de configuración personalizados y resuelve los detalles de los conflictos.

```

aws eks create-addon \
  --cluster-name my-eks-cluster \
  --addon-name my-eks-addon \
  --service-account-role-arn arn:aws:iam::111122223333:role/role-name \
  --addon-version v1.16.4-eksbuild.2 \
  --configuration-values 'file://configuration-values.json' \
  --resolve-conflicts OVERWRITE \
  --tags '{"eks-addon-key-1": "value-1" , "eks-addon-key-2": "value-2"}'

```

Contenidos de `configuration-values.json`:

```

{
  "resources": {
    "limits": {
      "cpu": "150m"
    }
  },
  "env": {
    "AWS_VPC_K8S_CNI_LOGLEVEL": "ERROR"
  }
}

```

```
}

```

Salida:

```
{
  "addon": {
    "addonName": "my-eks-addon",
    "clusterName": "my-eks-cluster",
    "status": "CREATING",
    "addonVersion": "v1.16.4-eksbuild.2",
    "health": {
      "issues": []
    },
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-addon/d8c71ef8-fbd8-07d0-fb32-6a7be19eeced",
    "createdAt": "2024-03-14T13:10:51.763000-04:00",
    "modifiedAt": "2024-03-14T13:10:51.777000-04:00",
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "tags": {
      "eks-addon-key-1": "value-1",
      "eks-addon-key-2": "value-2"
    },
    "configurationValues": "{\n  \"resources\": {\n    \"limits\": {\n      \"cpu\": \"150m\"\n    },\n    \"env\": {\n      \"AWS_VPC_K8S_CNI_LOGLEVEL\": \"ERROR\"\n    }\n  }"
  }
}
```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: creación de un complemento](#) en la Guía del EKS usuario de Amazon.

Ejemplo 5: Para crear un EKS complemento de Amazon con un archivo de valores YAML de configuración personalizados

El siguiente comando de `create-addon` ejemplo crea un EKS complemento de Amazon con valores de configuración personalizados y resuelve los detalles de los conflictos.

```
aws eks create-addon \
  --cluster-name my-eks-cluster \
  --addon-name my-eks-addon \
  --service-account-role-arn arn:aws:iam::111122223333:role/role-name \
  --addon-version v1.16.4-eksbuild.2 \
```

```
--configuration-values 'file://configuration-values.yaml' \
--resolve-conflicts OVERWRITE \
--tags '{"eks-addon-key-1": "value-1" , "eks-addon-key-2": "value-2"}'
```

Contenidos de configuration-values.yaml:

```
resources:
  limits:
    cpu: '100m'
env:
  AWS_VPC_K8S_CNI_LOGLEVEL: 'DEBUG'
```

Salida:

```
{
  "addon": {
    "addonName": "my-eks-addon",
    "clusterName": "my-eks-cluster",
    "status": "CREATING",
    "addonVersion": "v1.16.4-eksbuild.2",
    "health": {
      "issues": []
    },
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-addon/d4c71efb-3909-6f36-a548-402cd4b5d59e",
    "createdAt": "2024-03-14T13:15:45.220000-04:00",
    "modifiedAt": "2024-03-14T13:15:45.237000-04:00",
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "tags": {
      "eks-addon-key-3": "value-3",
      "eks-addon-key-4": "value-4"
    },
    "configurationValues": "resources:\n  limits:\n    cpu: '100m'\nenv:\n  AWS_VPC_K8S_CNI_LOGLEVEL: 'INFO'"
  }
}
```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: creación de un complemento](#) en la Guía del EKS usuario de Amazon.

- Para API obtener más información, consulte [CreateAddon](#) la Referencia de AWS CLI comandos.

## create-cluster

En el siguiente ejemplo de código, se muestra cómo usar `create-cluster`.

### AWS CLI

Creación de un nuevo clúster

Este comando de ejemplo crea un clúster denominado `prod` en la región predeterminada.

Comando:

```
aws eks create-cluster --name prod \  
--role-arn arn:aws:iam::012345678910:role/eks-service-role-  
AWSServiceRoleForAmazonEKS-J7ONKE3BQ4PI \  
--resources-vpc-config subnetIds=subnet-6782e71e,subnet-  
e7e761ac,securityGroupIds=sg-6979fe18
```

Salida:

```
{  
  "cluster": {  
    "name": "prod",  
    "arn": "arn:aws:eks:us-west-2:012345678910:cluster/prod",  
    "createdAt": 1527808069.147,  
    "version": "1.10",  
    "roleArn": "arn:aws:iam::012345678910:role/eks-service-role-  
AWSServiceRoleForAmazonEKS-J7ONKE3BQ4PI",  
    "resourcesVpcConfig": {  
      "subnetIds": [  
        "subnet-6782e71e",  
        "subnet-e7e761ac"  
      ],  
      "securityGroupIds": [  
        "sg-6979fe18"  
      ],  
      "vpcId": "vpc-950809ec"  
    },  
    "status": "CREATING",  
    "certificateAuthority": {}  
  }  
}
```

## Creación de un clúster nuevo con acceso privado al punto de conexión y el registro activados

Este comando de ejemplo crea un clúster denominado `example` de la región predeterminada con acceso público al punto de conexión desactivado, acceso privado al punto de conexión activado y todos los tipos de registro activados.

Comando:

```
aws eks create-cluster --name example --kubernetes-version 1.12 \
--role-arn arn:aws:iam::012345678910:role/example-cluster-ServiceRole-1XWBQWYSFRE2Q \
--resources-vpc-
config subnetIds=subnet-0a188dccd2f9a632f,subnet-09290d93da4278664,subnet-0f21dd86e0e91134a, \
--logging '{"clusterLogging":[{"types":
["api","audit","authenticator","controllerManager","scheduler"],"enabled":true}]}'
```

Salida:

```
{
  "cluster": {
    "name": "example",
    "arn": "arn:aws:eks:us-west-2:012345678910:cluster/example",
    "createdAt": 1565804921.901,
    "version": "1.12",
    "roleArn": "arn:aws:iam::012345678910:role/example-cluster-
ServiceRole-1XWBQWYSFRE2Q",
    "resourcesVpcConfig": {
      "subnetIds": [
        "subnet-0a188dccd2f9a632f",
        "subnet-09290d93da4278664",
        "subnet-0f21dd86e0e91134a",
        "subnet-0173dead68481a583",
        "subnet-051f70a57ed6fcab6",
        "subnet-01322339c5c7de9b4"
      ],
      "securityGroupIds": [
        "sg-0c5b580845a031c10"
      ],
      "vpcId": "vpc-0f622c01f68d4afec",
      "endpointPublicAccess": false,
      "endpointPrivateAccess": true
    },
  },
}
```

```

    "logging": {
      "clusterLogging": [
        {
          "types": [
            "api",
            "audit",
            "authenticator",
            "controllerManager",
            "scheduler"
          ],
          "enabled": true
        }
      ]
    },
    "status": "CREATING",
    "certificateAuthority": {},
    "platformVersion": "eks.3"
  }
}

```

- Para API obtener más información, consulte [CreateCluster](#) la Referencia de AWS CLI comandos.

## create-fargate-profile

En el siguiente ejemplo de código, se muestra cómo usar `create-fargate-profile`.

### AWS CLI

Ejemplo 1: Crear un perfil de EKS Fargate para un selector con un espacio de nombres

En el siguiente `create-fargate-profile` ejemplo, se crea un perfil de EKS Fargate para un selector con un espacio de nombres.

```

aws eks create-fargate-profile \
  --cluster-name my-eks-cluster \
  --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \
  --fargate-profile-name my-fargate-profile \
  --selectors '[{"namespace": "default"}]'

```

Salida:



```
{
  "fargateProfile": {
    "fargateProfileName": "my-fargate-profile",
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-eks-cluster/my-fargate-profile/a2c72bca-318e-abe8-8ed1-27c6d4892e9e",
    "clusterName": "my-eks-cluster",
    "createdAt": "2024-03-19T12:38:47.368000-04:00",
    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "subnets": [
      "subnet-09d912bb63ef21b9a",
      "subnet-04ad87f71c6e5ab4d",
      "subnet-0e2907431c9988b72"
    ],
    "selectors": [
      {
        "namespace": "default"
      }
    ],
    "status": "CREATING",
    "tags": {}
  }
}
```

Para obtener más información, consulta Perfil de [AWS Fargate: Creación de un perfil de Fargate en la Guía del usuario](#) de Amazon. EKS

Ejemplo 2: Crear un perfil de EKS Fargate para un selector con un espacio de nombres y etiquetas

El siguiente `create-fargate-profile` ejemplo crea un perfil de EKS Fargate para un selector con un espacio de nombres y etiquetas.

```
aws eks create-fargate-profile \
  --cluster-name my-eks-cluster \
  --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \
  --fargate-profile-name my-fargate-profile \
  --selectors '["namespace": "default", "labels": {"labelname1":  
"labelvalue1"}]'
```

Salida:

```
{
```

```

"fargateProfile": {
  "fargateProfileName": "my-fargate-profile",
  "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-eks-cluster/my-fargate-profile/88c72bc7-e8a4-fa34-44e4-2f1397224bb3",
  "clusterName": "my-eks-cluster",
  "createdAt": "2024-03-19T12:33:48.125000-04:00",
  "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
  "subnets": [
    "subnet-09d912bb63ef21b9a",
    "subnet-04ad87f71c6e5ab4d",
    "subnet-0e2907431c9988b72"
  ],
  "selectors": [
    {
      "namespace": "default",
      "labels": {
        "labelname1": "labelvalue1"
      }
    }
  ],
  "status": "CREATING",
  "tags": {}
}
}

```

Para obtener más información, consulta Perfil de [AWS Fargate: Creación de un perfil de Fargate en la Guía del usuario](#) de Amazon. EKS

Ejemplo 3: Crea un perfil de EKS Fargate para un selector con un espacio de nombres y etiquetas, junto con subredes en IDs las que lanzar un pod.

En el siguiente create-fargate-profile ejemplo, se crea un perfil EKS Fargate para un selector con un espacio de nombres y etiquetas, junto con subredes en IDs las que lanzar un pod.

```

aws eks create-fargate-profile \
  --cluster-name my-eks-cluster \
  --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \
  --fargate-profile-name my-fargate-profile \
  --selectors '[{"namespace": "default", "labels": {"labelname1":  

"labelvalue1"}}]' \
  --subnets ["subnet-09d912bb63ef21b9a", "subnet-04ad87f71c6e5ab4d",  

  "subnet-0e2907431c9988b72"]'

```

Salida:

```
{
  "fargateProfile": {
    "fargateProfileName": "my-fargate-profile",
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-eks-cluster/my-fargate-profile/e8c72bc8-e87b-5eb6-57cb-ed4fe57577e3",
    "clusterName": "my-eks-cluster",
    "createdAt": "2024-03-19T12:35:58.640000-04:00",
    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "subnets": [
      "subnet-09d912bb63ef21b9a",
      "subnet-04ad87f71c6e5ab4d",
      "subnet-0e2907431c9988b72"
    ],
    "selectors": [
      {
        "namespace": "default",
        "labels": {
          "labelname1": "labelvalue1"
        }
      }
    ],
    "status": "CREATING",
    "tags": {}
  }
}
```

Para obtener más información, consulta Perfil de [AWS Fargate: Creación de un perfil de Fargate en la Guía del usuario](#) de Amazon. EKS

Ejemplo 4: Crear un perfil de EKS Fargate para un selector con varios espacios de nombres y etiquetas, junto con subredes en las que IDs lanzar un pod

En el siguiente `create-fargate-profile` ejemplo, se crea un perfil de EKS Fargate para un selector con varios espacios de nombres y etiquetas, además IDs de subredes en las que lanzar un pod.

```
aws eks create-fargate-profile \
  --cluster-name my-eks-cluster \
  --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \
  --fargate-profile-name my-fargate-profile \
```

```

--selectors '[{"namespace": "default1", "labels": {"labelname1": "labelvalue1",
"labelname2": "labelvalue2"}}, {"namespace": "default2", "labels": {"labelname1":
"labelvalue1", "labelname2": "labelvalue2"}}]' \
--subnets ["subnet-09d912bb63ef21b9a", "subnet-04ad87f71c6e5ab4d",
"subnet-0e2907431c9988b72"]' \
--tags '{"eks-fargate-profile-key-1": "value-1" , "eks-fargate-profile-key-2":
"value-2"}'

```

Salida:

```

{
  "fargateProfile": {
    "fargateProfileName": "my-fargate-profile",
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-
eks-cluster/my-fargate-profile/4cc72bbf-b766-8ee6-8d29-e62748feb3cd",
    "clusterName": "my-eks-cluster",
    "createdAt": "2024-03-19T12:15:55.271000-04:00",
    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "subnets": [
      "subnet-09d912bb63ef21b9a",
      "subnet-04ad87f71c6e5ab4d",
      "subnet-0e2907431c9988b72"
    ],
    "selectors": [
      {
        "namespace": "default1",
        "labels": {
          "labelname2": "labelvalue2",
          "labelname1": "labelvalue1"
        }
      },
      {
        "namespace": "default2",
        "labels": {
          "labelname2": "labelvalue2",
          "labelname1": "labelvalue1"
        }
      }
    ],
    "status": "CREATING",
    "tags": {
      "eks-fargate-profile-key-2": "value-2",
      "eks-fargate-profile-key-1": "value-1"
    }
  }
}

```

```

    }
  }
}

```

Para obtener más información, consulta Perfil de [AWS Fargate: Creación de un perfil de Fargate en la Guía del usuario](#) de Amazon. EKS

Ejemplo 5: Crear un perfil de EKS Fargate con un selector comodín para los espacios de nombres y las etiquetas, junto con subredes en las que IDs lanzar un pod

En el siguiente `create-fargate-profile` ejemplo, se crea un perfil de EKS Fargate para un selector con varios espacios de nombres y etiquetas, además IDs de subredes en las que lanzar un pod.

```

aws eks create-fargate-profile \
  --cluster-name my-eks-cluster \
  --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \
  --fargate-profile-name my-fargate-profile \
  --selectors '[{"namespace": "prod*", "labels": {"labelname*?": "*value1"}}, {"namespace": "*dev*", "labels": {"labelname*?": "*value*"}}]' \
  --subnets ['subnet-09d912bb63ef21b9a', "subnet-04ad87f71c6e5ab4d", "subnet-0e2907431c9988b72"] \
  --tags '{"eks-fargate-profile-key-1": "value-1" , "eks-fargate-profile-key-2": "value-2"}'

```

Salida:

```

{
  "fargateProfile": {
    "fargateProfileName": "my-fargate-profile",
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-eks-cluster/my-fargate-profile/e8c72bd6-5966-0bfe-b77b-1802893e5a6f",
    "clusterName": "my-eks-cluster",
    "createdAt": "2024-03-19T13:05:20.550000-04:00",
    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "subnets": [
      "subnet-09d912bb63ef21b9a",
      "subnet-04ad87f71c6e5ab4d",
      "subnet-0e2907431c9988b72"
    ],
    "selectors": [
      {

```

```
        "namespace": "prod*",
        "labels": {
            "labelname*?": "*value1"
        }
    },
    {
        "namespace": "*dev*",
        "labels": {
            "labelname*?": "*value*"
        }
    }
],
"status": "CREATING",
"tags": {
    "eks-fargate-profile-key-2": "value-2",
    "eks-fargate-profile-key-1": "value-1"
}
}
```

Para obtener más información, consulta Perfil de [AWS Fargate: Creación de un perfil de Fargate en la Guía del usuario](#) de Amazon. EKS

- Para API obtener más información, consulte la Referencia [CreateFargateProfile](#) de AWS CLI comandos.

## create-nodegroup

En el siguiente ejemplo de código, se muestra cómo usar `create-nodegroup`.

### AWS CLI

Ejemplo 1: Crea un grupo de nodos gestionado para un EKS clúster de Amazon

En el siguiente `create-nodegroup` ejemplo, se crea un grupo de nodos gestionado para un EKS clúster de Amazon.

```
aws eks create-nodegroup \
  --cluster-name my-eks-cluster \
  --nodegroup-name my-eks-nodegroup \
  --node-role arn:aws:iam::111122223333:role/role-name \
```

```

--
subnets "subnet-0e2907431c9988b72" "subnet-04ad87f71c6e5ab4d" "subnet-09d912bb63ef21b9a"
\
--scaling-config minSize=1,maxSize=3,desiredSize=1 \
--region us-east-2

```

Salida:

```

{
  "nodegroup": {
    "nodegroupName": "my-eks-nodegroup",
    "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-
cluster/my-eks-nodegroup/bac7550f-b8b8-5fbb-4f3e-7502a931119e",
    "clusterName": "my-eks-cluster",
    "version": "1.26",
    "releaseVersion": "1.26.12-20240329",
    "createdAt": "2024-04-04T13:19:32.260000-04:00",
    "modifiedAt": "2024-04-04T13:19:32.260000-04:00",
    "status": "CREATING",
    "capacityType": "ON_DEMAND",
    "scalingConfig": {
      "minSize": 1,
      "maxSize": 3,
      "desiredSize": 1
    },
    "instanceTypes": [
      "t3.medium"
    ],
    "subnets": [
      "subnet-0e2907431c9988b72, subnet-04ad87f71c6e5ab4d,
subnet-09d912bb63ef21b9a"
    ],
    "amiType": "AL2_x86_64",
    "nodeRole": "arn:aws:iam::111122223333:role/role-name",
    "diskSize": 20,
    "health": {
      "issues": []
    },
    "updateConfig": {
      "maxUnavailable": 1
    },
    "tags": {}
  }
}

```

```
}

```

Para obtener más información, consulta [Cómo crear un grupo de nodos gestionado](#) en la Guía del EKS usuario de Amazon.

Ejemplo 2: Crea un grupo de nodos gestionado para un EKS clúster de Amazon con tipos de instancia y tamaño de disco personalizados

En el siguiente `create-nodegroup` ejemplo, se crea un grupo de nodos gestionado para un EKS clúster de Amazon con tipos de instancia y tamaño de disco personalizados.

```
aws eks create-nodegroup \
  --cluster-name my-eks-cluster \
  --nodegroup-name my-eks-nodegroup \
  --node-role arn:aws:iam::111122223333:role/role-name \
  --
subnets "subnet-0e2907431c9988b72" "subnet-04ad87f71c6e5ab4d" "subnet-09d912bb63ef21b9a" \
  \
  --scaling-config minSize=1,maxSize=3,desiredSize=1 \
  --capacity-type ON_DEMAND \
  --instance-types 'm5.large' \
  --disk-size 50 \
  --region us-east-2
```

Salida:

```
{
  "nodegroup": {
    "nodegroupName": "my-eks-nodegroup",
    "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-cluster/my-eks-nodegroup/c0c7551b-e4f9-73d9-992c-a450fdb82322",
    "clusterName": "my-eks-cluster",
    "version": "1.26",
    "releaseVersion": "1.26.12-20240329",
    "createdAt": "2024-04-04T13:46:07.595000-04:00",
    "modifiedAt": "2024-04-04T13:46:07.595000-04:00",
    "status": "CREATING",
    "capacityType": "ON_DEMAND",
    "scalingConfig": {
      "minSize": 1,
      "maxSize": 3,
      "desiredSize": 1
    }
  }
}
```



```

    },
    "instanceTypes": [
      "m5.large"
    ],
    "subnets": [
      "subnet-0e2907431c9988b72",
      "subnet-04ad87f71c6e5ab4d",
      "subnet-09d912bb63ef21b9a"
    ],
    "amiType": "AL2_x86_64",
    "nodeRole": "arn:aws:iam::111122223333:role/role-name",
    "diskSize": 50,
    "health": {
      "issues": []
    },
    "updateConfig": {
      "maxUnavailable": 1
    },
    "tags": {}
  }
}

```

Para obtener más información, consulta [Cómo crear un grupo de nodos gestionado](#) en la Guía del EKS usuario de Amazon.

Ejemplo 3: Crea un grupo de nodos gestionado para un EKS clúster de Amazon con tipos de instancia personalizados, tamaño de disco, tipo ami, tipo de capacidad, configuración de actualización, etiquetas, contaminaciones y etiquetas.

En el siguiente `create-nodegroup` ejemplo, se crea un grupo de nodos gestionado para un EKS clúster de Amazon con tipos de instancia personalizados, tamaño de disco, tipo ami, tipo de capacidad, configuración de actualización, etiquetas, contaminaciones y etiquetas.

```

aws eks create-nodegroup \
  --cluster-name my-eks-cluster \
  --nodegroup-name my-eks-nodegroup \
  --node-role arn:aws:iam::111122223333:role/role-name \
  --
subnets "subnet-0e2907431c9988b72" "subnet-04ad87f71c6e5ab4d" "subnet-09d912bb63ef21b9a" \
  --scaling-config minSize=1,maxSize=5,desiredSize=4 \
  --instance-types 't3.large' \
  --disk-size 50 \

```

```

--ami-type AL2_x86_64 \
--capacity-type SPOT \
--update-config maxUnavailable=2 \
--labels '{"my-eks-nodegroup-label-1": "value-1" , "my-eks-nodegroup-label-2":
"value-2"}' \
--taints '{"key": "taint-key-1" , "value": "taint-value-1", "effect":
"NO_EXECUTE"}' \
--tags '{"my-eks-nodegroup-key-1": "value-1" , "my-eks-nodegroup-key-2":
"value-2"}'

```

Salida:

```

{
  "nodegroup": {
    "nodegroupName": "my-eks-nodegroup",
    "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-
cluster/my-eks-nodegroup/88c75524-97af-0cb9-a9c5-7c0423ab5314",
    "clusterName": "my-eks-cluster",
    "version": "1.26",
    "releaseVersion": "1.26.12-20240329",
    "createdAt": "2024-04-04T14:05:07.940000-04:00",
    "modifiedAt": "2024-04-04T14:05:07.940000-04:00",
    "status": "CREATING",
    "capacityType": "SPOT",
    "scalingConfig": {
      "minSize": 1,
      "maxSize": 5,
      "desiredSize": 4
    },
    "instanceTypes": [
      "t3.large"
    ],
    "subnets": [
      "subnet-0e2907431c9988b72",
      "subnet-04ad87f71c6e5ab4d",
      "subnet-09d912bb63ef21b9a"
    ],
    "amiType": "AL2_x86_64",
    "nodeRole": "arn:aws:iam::111122223333:role/role-name",
    "labels": {
      "my-eks-nodegroup-label-2": "value-2",
      "my-eks-nodegroup-label-1": "value-1"
    }
  },

```

```
    "taints": [
      {
        "key": "taint-key-1",
        "value": "taint-value-1",
        "effect": "NO_EXECUTE"
      }
    ],
    "diskSize": 50,
    "health": {
      "issues": []
    },
    "updateConfig": {
      "maxUnavailable": 2
    },
    "tags": {
      "my-eks-nodegroup-key-1": "value-1",
      "my-eks-nodegroup-key-2": "value-2"
    }
  }
}
```

Para obtener más información, consulta [Cómo crear un grupo de nodos gestionado](#) en la Guía del EKS usuario de Amazon.

- Para API obtener más información, consulte [CreateNodegroup](#) la Referencia de AWS CLI comandos.

## delete-addon

En el siguiente ejemplo de código, se muestra cómo usar delete-addon.

### AWS CLI

Ejemplo 1. Para eliminar un EKS complemento de Amazon pero conservar el software del complemento en el clúster EKS

El siguiente comando de delete-addon ejemplo elimina un EKS complemento de Amazon, pero conserva el software del complemento en el EKS clúster.

```
aws eks delete-addon \
  --cluster-name my-eks-cluster \
  --addon-name my-eks-addon \
```

**--preserve**

Salida:

```
{
  "addon": {
    "addonName": "my-eks-addon",
    "clusterName": "my-eks-cluster",
    "status": "DELETING",
    "addonVersion": "v1.9.3-eksbuild.7",
    "health": {
      "issues": []
    },
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-addon/a8c71ed3-944e-898b-9167-c763856af4b8",
    "createdAt": "2024-03-14T11:49:09.009000-04:00",
    "modifiedAt": "2024-03-14T12:03:49.776000-04:00",
    "tags": {}
  }
}
```

Para obtener más información, consulta [Administrar EKS complementos de Amazon: Eliminar un complemento](#) en Amazon EKS.

Ejemplo 2. Para eliminar un EKS complemento de Amazon y también eliminar el software del complemento del clúster EKS

El siguiente comando de `delete-addon` ejemplo elimina un EKS complemento de Amazon y también elimina el software del complemento del EKS clúster.

```
aws eks delete-addon \
  --cluster-name my-eks-cluster \
  --addon-name my-eks-addon
```

Salida:

```
{
  "addon": {
    "addonName": "my-eks-addon",
    "clusterName": "my-eks-cluster",
    "status": "DELETING",
    "addonVersion": "v1.15.1-eksbuild.1",
```

```
    "health": {
      "issues": []
    },
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-
addon/bac71ed1-ec43-3bb6-88ea-f243cdb58954",
    "createdAt": "2024-03-14T11:45:31.983000-04:00",
    "modifiedAt": "2024-03-14T11:58:40.136000-04:00",
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
    "tags": {}
  }
}
```

Para obtener más información, consulta [Administrar EKS complementos de Amazon: Eliminar un complemento](#) en Amazon EKS.

- Para API obtener más información, consulte [DeleteAddon](#) la Referencia de AWS CLI comandos.

## delete-cluster

En el siguiente ejemplo de código, se muestra cómo usar `delete-cluster`.

### AWS CLI

Eliminar un plano de control de un EKS clúster de Amazon

El siguiente `delete-cluster` ejemplo elimina un plano de control de un EKS clúster de Amazon.

```
aws eks delete-cluster \
  --name my-eks-cluster
```

Salida:

```
{
  "cluster": {
    "name": "my-eks-cluster",
    "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster",
    "createdAt": "2024-03-14T11:31:44.348000-04:00",
    "version": "1.27",
    "endpoint": "https://DAL5J343KE23J3RN45653DSKJTT647TYD.y14.us-
east-2.eks.amazonaws.com",
    "roleArn": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-cluster-
ServiceRole-zMF6CBakwwbW",
```

```
"resourcesVpcConfig": {
  "subnetIds": [
    "subnet-0fb75d2d8401716e7",
    "subnet-02184492f67a3d0f9",
    "subnet-04098063527aab776",
    "subnet-0e2907431c9988b72",
    "subnet-04ad87f71c6e5ab4d",
    "subnet-09d912bb63ef21b9a"
  ],
  "securityGroupIds": [
    "sg-0c1327f6270afbb36"
  ],
  "clusterSecurityGroupId": "sg-01c84d09d70f39a7f",
  "vpcId": "vpc-0012b8e1cc0abb17d",
  "endpointPublicAccess": true,
  "endpointPrivateAccess": true,
  "publicAccessCidrs": [
    "0.0.0.0/0"
  ]
},
"kubernetesNetworkConfig": {
  "serviceIpv4Cidr": "10.100.0.0/16",
  "ipFamily": "ipv4"
},
"logging": {
  "clusterLogging": [
    {
      "types": [
        "api",
        "audit",
        "authenticator",
        "controllerManager",
        "scheduler"
      ],
      "enabled": true
    }
  ]
},
"identity": {
  "oidc": {
    "issuer": "https://oidc.eks.us-east-2.amazonaws.com/id/
DALSJ343KE23J3RN45653DSKJTT647TYD"
  }
},
```

```

    "status": "DELETING",
    "certificateAuthority": {
      "data": "XXX_CA_DATA_XXX"
    },
    "platformVersion": "eks.16",
    "tags": {
      "aws:cloudformation:stack-name": "eksctl-my-eks-cluster-cluster",
      "alpha.eksctl.io/cluster-name": "my-eks-cluster",
      "karpenter.sh/discovery": "my-eks-cluster",
      "aws:cloudformation:stack-id": "arn:aws:cloudformation:us-
east-2:111122223333:stack/eksctl-my-eks-cluster-cluster/e752ea00-e217-11ee-
beae-0a9599c8c7ed",
      "auto-delete": "no",
      "eksctl.cluster.k8s.io/v1alpha1/cluster-name": "my-eks-cluster",
      "EKS-Cluster-Name": "my-eks-cluster",
      "alpha.eksctl.io/cluster-oidc-enabled": "true",
      "aws:cloudformation:logical-id": "ControlPlane",
      "alpha.eksctl.io/eksctl-version": "0.173.0-dev
+a7ee89342.2024-03-01T03:40:57Z",
      "Name": "eksctl-my-eks-cluster-cluster/ControlPlane"
    },
    "accessConfig": {
      "authenticationMode": "API_AND_CONFIG_MAP"
    }
  }
}

```

Para obtener más información, consulta [Eliminar un EKS clúster de Amazon](#) en la Guía del EKS usuario de Amazon.

- Para API obtener más información, consulte [DeleteCluster](#) la Referencia de AWS CLI comandos.

## delete-fargate-profile

En el siguiente ejemplo de código, se muestra cómo usar `delete-fargate-profile`.

### AWS CLI

Ejemplo 1: Crear un perfil de EKS Fargate para un selector con un espacio de nombres

En el siguiente `delete-fargate-profile` ejemplo, se crea un perfil de EKS Fargate para un selector con un espacio de nombres.

```
aws eks delete-fargate-profile \  
  --cluster-name my-eks-cluster \  
  --fargate-profile-name my-fargate-profile
```

Salida:

```
{  
  "fargateProfile": {  
    "fargateProfileName": "my-fargate-profile",  
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-  
eks-cluster/my-fargate-profile/1ac72bb3-3fc6-2631-f1e1-98bff53bed62",  
    "clusterName": "my-eks-cluster",  
    "createdAt": "2024-03-19T11:48:39.975000-04:00",  
    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",  
    "subnets": [  
      "subnet-09d912bb63ef21b9a",  
      "subnet-04ad87f71c6e5ab4d",  
      "subnet-0e2907431c9988b72"  
    ],  
    "selectors": [  
      {  
        "namespace": "default",  
        "labels": {  
          "foo": "bar"  
        }  
      }  
    ],  
    "status": "DELETING",  
    "tags": {}  
  }  
}
```

Para obtener más información, consulta [Perfil de AWS Fargate: Eliminar un Fargate en la Guía del usuario](#) de Amazon. EKS

- Para API obtener más información, consulte la Referencia [DeleteFargateProfile](#) de AWS CLI comandos.

## delete-nodegroup

En el siguiente ejemplo de código, se muestra cómo usar delete-nodegroup.



## AWS CLI

### Ejemplo 1: Eliminar un grupo de nodos gestionado para un EKS clúster de Amazon

En el siguiente `delete-nodegroup` ejemplo, se elimina un grupo de nodos gestionado de un EKS clúster de Amazon.

```
aws eks delete-nodegroup \  
  --cluster-name my-eks-cluster \  
  --nodegroup-name my-eks-nodegroup
```

Salida:

```
{  
  "nodegroup": {  
    "nodegroupName": "my-eks-nodegroup",  
    "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-  
cluster/my-eks-nodegroup/1ec75f5f-0e21-dcc0-b46e-f9c442685cd8",  
    "clusterName": "my-eks-cluster",  
    "version": "1.26",  
    "releaseVersion": "1.26.12-20240329",  
    "createdAt": "2024-04-08T13:25:15.033000-04:00",  
    "modifiedAt": "2024-04-08T13:25:31.252000-04:00",  
    "status": "DELETING",  
    "capacityType": "SPOT",  
    "scalingConfig": {  
      "minSize": 1,  
      "maxSize": 5,  
      "desiredSize": 4  
    },  
    "instanceTypes": [  
      "t3.large"  
    ],  
    "subnets": [  
      "subnet-0e2907431c9988b72",  
      "subnet-04ad87f71c6e5ab4d",  
      "subnet-09d912bb63ef21b9a"  
    ],  
    "amiType": "AL2_x86_64",  
    "nodeRole": "arn:aws:iam::111122223333:role/role-name",  
    "labels": {  
      "my-eks-nodegroup-label-2": "value-2",  
      "my-eks-nodegroup-label-1": "value-1"  
    }  
  }  
}
```

```
    },
    "taints": [
      {
        "key": "taint-key-1",
        "value": "taint-value-1",
        "effect": "NO_EXECUTE"
      }
    ],
    "diskSize": 50,
    "health": {
      "issues": []
    },
    "updateConfig": {
      "maxUnavailable": 2
    },
    "tags": {
      "my-eks-nodegroup-key-1": "value-1",
      "my-eks-nodegroup-key-2": "value-2"
    }
  }
}
```

- Para API obtener más información, consulte [DeleteNodegroup](#) la Referencia de AWS CLI comandos.

## deregister-cluster

En el siguiente ejemplo de código, se muestra cómo usar `deregister-cluster`.

### AWS CLI

Para anular el registro de un clúster conectado y eliminarlo del plano de control de Amazon EKS

El siguiente `deregister-cluster` ejemplo anula el registro de un clúster conectado para eliminarlo del plano de EKS control de Amazon.

```
aws eks deregister-cluster \  
  --name my-eks-anywhere-cluster
```

Salida:

```
{
```

```

"cluster": {
  "name": "my-eks-anywhere-cluster",
  "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-eks-anywhere-cluster",
  "createdAt": "2024-04-12T12:38:37.561000-04:00",
  "status": "DELETING",
  "tags": {},
  "connectorConfig": {
    "activationId": "dfb5ad28-13c3-4e26-8a19-5b2457638c74",
    "activationExpiry": "2024-04-15T12:38:37.082000-04:00",
    "provider": "EKS_ANYWHERE",
    "roleArn": "arn:aws:iam::111122223333:role/AmazonEKSCONNECTORAGENTROLE"
  }
}
}
}

```

Para obtener más información, consulta Cómo [anular el registro de un clúster](#) en la Guía del EKSusuario de Amazon.

- Para API obtener más información, consulte la Referencia [DeregisterCluster](#) de AWS CLI comandos.

## describe-addon-configuration

En el siguiente ejemplo de código, se muestra cómo usar `describe-addon-configuration`.

### AWS CLI

Ejemplo 1: Opciones de configuración disponibles al crear o actualizar Amazon vpc-cni AddOns

El siguiente `describe-addon-configuration` ejemplo devuelve todo el esquema de configuración disponible que se utiliza cuando se crea o actualiza un complemento para el complemento vpc-cni con la versión correspondiente.

```

aws eks describe-addon-configuration \
  --addon-name vpc-cni \
  --addon-version v1.15.1-eksbuild.1

```

Salida:

```

{
  "addonName": "vpc-cni",
  "addonVersion": "v1.15.1-eksbuild.1",

```

```

"configurationSchema": "{ \"$ref\": \"#/definitions/VpcCni\", \"$schema\": \"http://
json-schema.org/draft-06/schema#\", \"definitions\": { \"Affinity\": { \"type\":
[\"object\", \"null\"] }, \"EniConfig\": { \"additionalProperties\": false, \"properties
\": { \"create\": { \"type\": \"boolean\" }, \"region\": { \"type\": \"string\" }, \"subnets\":
{ \"additionalProperties\": { \"additionalProperties\": false, \"properties\": { \"id\":
{ \"type\": \"string\" }, \"securityGroups\": { \"items\": { \"type\": \"string\" }, \"type\":
\"array\" } }, \"required\": [\"id\"], \"type\": \"object\" }, \"minProperties\": 1, \"type
\": \"object\" } }, \"required\": [\"create\", \"region\", \"subnets\"], \"type\": \"object
\", \"Env\": { \"additionalProperties\": false, \"properties\": { \"ADDITIONAL_ENI_TAGS
\": { \"type\": \"string\" }, \"ANNOTATE_POD_IP\": { \"format\": \"boolean\", \"type\":
\"string\" }, \"AWS_EC2_ENDPOINT\": { \"type\": \"string\" }, \"AWS_EXTERNAL_SERVICE_CIDRS
\": { \"type\": \"string\" }, \"AWS_MANAGE_ENIS_NON_SCHEDULABLE\": { \"format\": \"boolean
\", \"type\": \"string\" }, \"AWS_VPC_CNI_NODE_PORT_SUPPORT\": { \"format\": \"boolean
\", \"type\": \"string\" }, \"AWS_VPC_ENI_MTU\": { \"format\": \"integer\", \"type\":
\"string\" }, \"AWS_VPC_K8S_CNI_CUSTOM_NETWORK_CFG\": { \"format\": \"boolean\", \"type
\": \"string\" }, \"AWS_VPC_K8S_CNI_EXCLUDE_SNAT_CIDRS\": { \"type\": \"string\" },
\"AWS_VPC_K8S_CNI_EXTERNALSNAT\": { \"format\": \"boolean\", \"type\": \"string\" },
\"AWS_VPC_K8S_CNI_LOGLEVEL\": { \"type\": \"string\" }, \"AWS_VPC_K8S_CNI_LOG_FILE\":
{ \"type\": \"string\" }, \"AWS_VPC_K8S_CNI_RANDOMIZESNAT\": { \"type\": \"string\" },
\"AWS_VPC_K8S_CNI_VETHPREFIX\": { \"type\": \"string\" }, \"AWS_VPC_K8S_PLUGIN_LOG_FILE
\": { \"type\": \"string\" }, \"AWS_VPC_K8S_PLUGIN_LOG_LEVEL\": { \"type\": \"string
\" }, \"CLUSTER_ENDPOINT\": { \"type\": \"string\" }, \"DISABLE_INTROSPECTION\":
{ \"format\": \"boolean\", \"type\": \"string\" }, \"DISABLE_LEAKED_ENI_CLEANUP\":
{ \"format\": \"boolean\", \"type\": \"string\" }, \"DISABLE_METRICS\": { \"format
\": \"boolean\", \"type\": \"string\" }, \"DISABLE_NETWORK_RESOURCE_PROVISIONING
\": { \"format\": \"boolean\", \"type\": \"string\" }, \"DISABLE_POD_V6\": { \"format
\": \"boolean\", \"type\": \"string\" }, \"ENABLE_BANDWIDTH_PLUGIN\": { \"format\":
\"boolean\", \"type\": \"string\" }, \"ENABLE_POD_ENI\": { \"format\": \"boolean\",
\"type\": \"string\" }, \"ENABLE_PREFIX_DELEGATION\": { \"format\": \"boolean\",
\"type\": \"string\" }, \"ENABLE_V4_EGRESS\": { \"format\": \"boolean\", \"type\":
\"string\" }, \"ENABLE_V6_EGRESS\": { \"format\": \"boolean\", \"type\": \"string\" },
\"ENI_CONFIG_ANNOTATION_DEF\": { \"type\": \"string\" }, \"ENI_CONFIG_LABEL_DEF\":
{ \"type\": \"string\" }, \"INTROSPECTION_BIND_ADDRESS\": { \"type\": \"string\" },
\"IP_COOLDOWN_PERIOD\": { \"format\": \"integer\", \"type\": \"string\" }, \"MAX_ENI
\": { \"format\": \"integer\", \"type\": \"string\" }, \"MINIMUM_IP_TARGET\": { \"format
\": \"integer\", \"type\": \"string\" }, \"POD_SECURITY_GROUP_ENFORCING_MODE\":
{ \"type\": \"string\" }, \"WARM_ENI_TARGET\": { \"format\": \"integer\", \"type\":
\"string\" }, \"WARM_IP_TARGET\": { \"format\": \"integer\", \"type\": \"string\" },
\"WARM_PREFIX_TARGET\": { \"format\": \"integer\", \"type\": \"string\" } }, \"title
\": \"Env\", \"type\": \"object\" }, \"Init\": { \"additionalProperties\": false,
\"properties\": { \"env\": { \"$ref\": \"#/definitions/InitEnv\" } }, \"title\": \"Init
\", \"type\": \"object\" }, \"InitEnv\": { \"additionalProperties\": false, \"properties
\": { \"DISABLE_TCP_EARLY_DEMUX\": { \"format\": \"boolean\", \"type\": \"string\" },
\"ENABLE_V6_EGRESS\": { \"format\": \"boolean\", \"type\": \"string\" } }, \"title\":

```

```

\ "InitEnv", \ "type": \ "object", \ "Limits": { \ "additionalProperties": false,
\ "properties": { \ "cpu": { \ "type": \ "string"}, \ "memory": { \ "type": \ "string"}},
\ "title": \ "Limits", \ "type": \ "object", \ "NodeAgent": { \ "additionalProperties
\ ": false, \ "properties": { \ "enableCloudWatchLogs": { \ "format": \ "boolean",
\ "type": \ "string"}, \ "enablePolicyEventLogs": { \ "format": \ "boolean", \ "type":
\ "string"}, \ "healthProbeBindAddr": { \ "format": \ "integer", \ "type": \ "string
\ }, \ "metricsBindAddr": { \ "format": \ "integer", \ "type": \ "string"}}, \ "title":
\ "NodeAgent", \ "type": \ "object", \ "Resources": { \ "additionalProperties": false,
\ "properties": { \ "limits": { \ "$ref": \ "#/definitions/Limits"}, \ "requests":
{ \ "$ref": \ "#/definitions/Limits"}}, \ "title": \ "Resources", \ "type": \ "object
\ }, \ "Tolerations": { \ "additionalProperties": false, \ "items": { \ "type": \ "object
\ }, \ "type": \ "array"}, \ "VpcCni": { \ "additionalProperties": false, \ "properties
\ ": { \ "affinity": { \ "$ref": \ "#/definitions/Affinity"}, \ "enableNetworkPolicy":
{ \ "format": \ "boolean", \ "type": \ "string"}, \ "enableWindowsIpam": { \ "format":
\ "boolean", \ "type": \ "string"}, \ "eniConfig": { \ "$ref": \ "#/definitions/EniConfig
\ }, \ "env": { \ "$ref": \ "#/definitions/Env"}, \ "init": { \ "$ref": \ "#/definitions/Init
\ }, \ "livenessProbeTimeoutSeconds": { \ "type": \ "integer"}, \ "nodeAgent": { \ "$ref":
\ "#/definitions/NodeAgent"}, \ "readinessProbeTimeoutSeconds": { \ "type": \ "integer
\ }, \ "resources": { \ "$ref": \ "#/definitions/Resources"}, \ "tolerations": { \ "$ref
\ ": \ "#/definitions/Tolerations"}}, \ "title": \ "VpcCni", \ "type": \ "object"},
\ "description": \ "vpc-cni"
}

```

Ejemplo 2: Opciones de configuración disponibles al crear o actualizar Amazon coreDNS AddOns

El siguiente describe-addon-configuration ejemplo devuelve todo el esquema de configuración disponible que se utiliza cuando se crea o actualiza un complemento para el complemento coreDNS con la versión correspondiente.

```

aws eks describe-addon-configuration \
  --addon-name coredns \
  --addon-version v1.8.7-eksbuild.4

```

Salida:

```

{
  "addonName": "coredns",
  "addonVersion": "v1.8.7-eksbuild.4",
  "configurationSchema": "{ \ "$ref": \ "#/definitions/Coredns", \ "$schema
\ ": \ "http://json-schema.org/draft-06/schema#", \ "definitions": { \ "Coredns":
{ \ "additionalProperties": false, \ "properties": { \ "computeType": { \ "type":
\ "string"}, \ "corefile": { \ "description": \ "Entire corefile contents to use with
\ installation", \ "type": \ "string"}, \ "nodeSelector": { \ "additionalProperties":

```

```
{
  "type": "string",
  "type": "object",
  "replicaCount": {
    "type": "integer"
  },
  "resources": {
    "$ref": "#/definitions/Resources"
  },
  "title": "Coredns",
  "type": "object",
  "Limits": {
    "additionalProperties": false,
    "properties": {
      "cpu": {
        "type": "string"
      },
      "memory": {
        "type": "string"
      }
    },
    "title": "Limits"
  },
  "Resources": {
    "additionalProperties": false,
    "properties": {
      "limits": {
        "$ref": "#/definitions/Limits"
      },
      "requests": {
        "$ref": "#/definitions/Limits"
      }
    },
    "title": "Resources",
    "type": "object"
  }
}
```

Para obtener más información, consulta [Cómo crear o actualizar un archivo kubeconfig para un EKS clúster de Amazon](#) en Amazon. EKS

- Para API obtener más información, consulte la Referencia [DescribeAddonConfiguration](#) de AWS CLI comandos.

## describe-addon-versions

En el siguiente ejemplo de código, se muestra cómo usar `describe-addon-versions`.

### AWS CLI

Ejemplo 1: Listar todos los complementos disponibles para Cluster EKS

En el siguiente `describe-addon-versions` ejemplo, se enumeran todos los AWS complementos disponibles.

```
aws eks describe-addon-versions \
  --query 'sort_by(addons &owner)[].{publisher: publisher, owner: owner,
  addonName: addonName, type: type}' \
  --output table
```

Salida:

```
-----
|                                                                 DescribeAddonVersions
|                                                                 |
+-----+-----+-----+-----+
|          addonName          |          owner          |          publisher
|          |          type          |          |          |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
```

vpc-cni	aws	eks
networking		
snapshot-controller	aws	eks
storage		
kube-proxy	aws	eks
networking		
eks-pod-identity-agent	aws	eks
security		
coredns	aws	eks
networking		
aws-mountpoint-s3-csi-driver	aws	s3
storage		
aws-guardduty-agent	aws	eks
security		
aws-efs-csi-driver	aws	eks
storage		
aws-ebs-csi-driver	aws	eks
storage		
amazon-cloudwatch-observability	aws	eks
observability		
adot	aws	eks
observability		
upwind-security_upwind-operator	aws-marketplace	Upwind Security
security		
upbound_universal-crossplane	aws-marketplace	upbound
infra-management		
tetrade-io_istio-distro	aws-marketplace	tetrade-io
policy-management		
teleport_teleport	aws-marketplace	teleport
policy-management		
stormforge_optimize-live	aws-marketplace	StormForge
cost-management		
splunk_splunk-otel-collector-chart	aws-marketplace	Splunk
monitoring		
solo-io_istio-distro	aws-marketplace	Solo.io
service-mesh		
rafay-systems_rafay-operator	aws-marketplace	rafay-systems
kubernetes-management		
new-relic_kubernetes-operator	aws-marketplace	New Relic
observability		
netapp_trident-operator	aws-marketplace	NetApp Inc.
storage		
leaksignal_leakagent	aws-marketplace	leaksignal
monitoring		

```

| kubecost_kubecost | aws-marketplace | kubecost
|   | cost-management |
| kong_konnect-ri | aws-marketplace | kong
|   | ingress-service-type |
| kasten_k10 | aws-marketplace | Kasten by Veeam
|   | data-protection |
| haproxy-technologies_kubernetes-ingress-ee | aws-marketplace | HAProxy
Technologies | ingress-controller |
| groundcover_agent | aws-marketplace | groundcover
|   | monitoring |
| grafana-labs_kubernetes-monitoring | aws-marketplace | Grafana Labs
|   | monitoring |
| factorhouse_kpow | aws-marketplace | factorhouse
|   | monitoring |
| dynatrace_dynatrace-operator | aws-marketplace | dynatrace
|   | monitoring |
| datree_engine-pro | aws-marketplace | datree
|   | policy-management |
| datadog_operator | aws-marketplace | Datadog
|   | monitoring |
| cribl_cribledge | aws-marketplace | Cribl
|   | observability |
| calyptia_fluent-bit | aws-marketplace | Calyptia Inc
|   | observability |
| accuknox_kubearmor | aws-marketplace | AccuKnox
|   | security |
+-----+-----+
+-----+-----+

```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: creación de un complemento](#) en la Guía del EKS usuario de Amazon.

Ejemplo 2: Listar todos los complementos disponibles para la versión específica de Kubernetes compatible con EKS

En el siguiente `describe-addon-versions` ejemplo, se enumeran todos los complementos disponibles para la versión específica de Kubernetes compatible. EKS

```

aws eks describe-addon-versions \
  --kubernetes-version=1.26 \
  --query 'sort_by(addons &owner)[].{publisher: publisher, owner: owner,
addonName: addonName, type: type}' \
  --output table

```



## Salida:

```

-----
|                                     DescribeAddonVersions
|                                     |
+-----+-----+-----+-----+
|                                     |                                     |
|          addonName                 |          owner           |          publisher
|          |          type            |                                     |
+-----+-----+-----+-----+
| vpc-cni                             | aws                       | eks
|   | networking                      |                             |
| snapshot-controller                 | aws                       | eks
|   | storage                         |                             |
| kube-proxy                          | aws                       | eks
|   | networking                      |                             |
| eks-pod-identity-agent              | aws                       | eks
|   | security                       |                             |
| coredns                             | aws                       | eks
|   | networking                      |                             |
| aws-mountpoint-s3-csi-driver        | aws                       | s3
|   | storage                         |                             |
| aws-guardduty-agent                 | aws                       | eks
|   | security                       |                             |
| aws-efs-csi-driver                  | aws                       | eks
|   | storage                         |                             |
| aws-ebs-csi-driver                  | aws                       | eks
|   | storage                         |                             |
| amazon-cloudwatch-observability    | aws                       | eks
|   | observability                  |                             |
| adot                                | aws                       | eks
|   | observability                  |                             |
| upwind-security_upwind-operator    | aws-marketplace          | Upwind Security
|   | security                       |                             |
| tetrade-io_istio-distro             | aws-marketplace          | tetrade-io
|   | policy-management              |                             |
| stormforge_optimize-live           | aws-marketplace          | StormForge
|   | cost-management                |                             |
| splunk_splunk-otel-collector-chart | aws-marketplace          | Splunk
|   | monitoring                     |                             |
| solo-io_istio-distro                | aws-marketplace          | Solo.io
|   | service-mesh                   |

```

```

| rafay-systems_rafay-operator | aws-marketplace | rafay-systems
| | kubernetes-management |
| new-relic_kubernetes-operator | aws-marketplace | New Relic
| | observability |
| netapp_trident-operator | aws-marketplace | NetApp Inc.
| | storage |
| leaksignal_leakagent | aws-marketplace | leaksignal
| | monitoring |
| kubecost_kubecost | aws-marketplace | kubecost
| | cost-management |
| kong_konnect-ri | aws-marketplace | kong
| | ingress-service-type |
| haproxy-technologies_kubernetes-ingress-ee | aws-marketplace | HAProxy
Technologies | ingress-controller |
| groundcover_agent | aws-marketplace | groundcover
| | monitoring |
| grafana-labs_kubernetes-monitoring | aws-marketplace | Grafana Labs
| | monitoring |
| dynatrace_dynatrace-operator | aws-marketplace | dynatrace
| | monitoring |
| datadog_operator | aws-marketplace | Datadog
| | monitoring |
| cribl_cribledge | aws-marketplace | Cribl
| | observability |
| calyptia_fluent-bit | aws-marketplace | Calyptia Inc
| | observability |
| accuknox_kubearmor | aws-marketplace | AccuKnox
| | security |
+-----+-----+
+-----+-----+

```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: creación de un complemento](#) en la Guía del EKS usuario de Amazon.

Ejemplo 3: Enumere todas las versiones de complementos de vpc-cni disponibles para una versión específica de Kubernetes compatible con EKS

En el siguiente `describe-addon-versions` ejemplo, se enumeran todas las versiones de complementos de vpc-cni disponibles para la versión específica de Kubernetes compatible. EKS

```

aws eks describe-addon-versions \
  --kubernetes-version=1.26 \
  --addon-name=vpc-cni \

```

```
--query='addons[].addonVersions[].addonVersion'
```

Salida:

```
[  
  "v1.18.0-eksbuild.1",  
  "v1.17.1-eksbuild.1",  
  "v1.16.4-eksbuild.2",  
  "v1.16.3-eksbuild.2",  
  "v1.16.2-eksbuild.1",  
  "v1.16.0-eksbuild.1",  
  "v1.15.5-eksbuild.1",  
  "v1.15.4-eksbuild.1",  
  "v1.15.3-eksbuild.1",  
  "v1.15.1-eksbuild.1",  
  "v1.15.0-eksbuild.2",  
  "v1.14.1-eksbuild.1",  
  "v1.14.0-eksbuild.3",  
  "v1.13.4-eksbuild.1",  
  "v1.13.3-eksbuild.1",  
  "v1.13.2-eksbuild.1",  
  "v1.13.0-eksbuild.1",  
  "v1.12.6-eksbuild.2",  
  "v1.12.6-eksbuild.1",  
  "v1.12.5-eksbuild.2",  
  "v1.12.0-eksbuild.2"  
]
```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: creación de un complemento](#) en la Guía del EKS usuario de Amazon.

- Para API obtener más información, consulte [DescribeAddonVersions](#) la Referencia de AWS CLI comandos.

## describe-addon

En el siguiente ejemplo de código, se muestra cómo usar `describe-addon`.

### AWS CLI

Describe el EKS complemento que se está ejecutando activamente en tu clúster de Amazon EKS

El siguiente describe-addon ejemplo es la ejecución activa de un EKS complemento en tu EKS clúster de Amazon.

```
aws eks describe-addon \  
  --cluster-name my-eks-cluster \  
  --addon-name vpc-cni
```

Salida:

```
{  
  "addon": {  
    "addonName": "vpc-cni",  
    "clusterName": "my-eks-cluster",  
    "status": "ACTIVE",  
    "addonVersion": "v1.16.4-eksbuild.2",  
    "health": {  
      "issues": []  
    },  
    "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/vpc-cni/0ec71efc-98dd-3203-60b0-4b939b2a5e5f",  
    "createdAt": "2024-03-14T13:18:45.417000-04:00",  
    "modifiedAt": "2024-03-14T13:18:49.557000-04:00",  
    "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-addon-vpc-cni-Role1-Yfakrq0C1UTm",  
    "tags": {  
      "eks-addon-key-3": "value-3",  
      "eks-addon-key-4": "value-4"  
    },  
    "configurationValues": "resources:\n    limits:\n    cpu: '100m'\nenv:\n  AWS_VPC_K8S_CNI_LOGLEVEL: 'DEBUG'  
  }  
}
```

- Para API obtener más información, consulta [DescribeAddon](#) la Referencia de AWS CLI comandos.

## describe-cluster

En el siguiente ejemplo de código, se muestra cómo usar describe-cluster.

## AWS CLI

Describe el EKS complemento que se está ejecutando activamente en tu clúster de Amazon EKS

El siguiente describe-cluster ejemplo es la ejecución activa de un EKS complemento en tu EKS clúster de Amazon.

```
aws eks describe-cluster \
  --cluster-name my-eks-cluster
```

Salida:

```
{
  "cluster": {
    "name": "my-eks-cluster",
    "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster",
    "createdAt": "2024-03-14T11:31:44.348000-04:00",
    "version": "1.26",
    "endpoint": "https://JSA79429HJDASKJDJ8223829MNDNASW.y14.us-east-2.eks.amazonaws.com",
    "roleArn": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-cluster-ServiceRole-zMF6CBakwwbW",
    "resourcesVpcConfig": {
      "subnetIds": [
        "subnet-0fb75d2d8401716e7",
        "subnet-02184492f67a3d0f9",
        "subnet-04098063527aab776",
        "subnet-0e2907431c9988b72",
        "subnet-04ad87f71c6e5ab4d",
        "subnet-09d912bb63ef21b9a"
      ],
      "securityGroupIds": [
        "sg-0c1327f6270afbb36"
      ],
      "clusterSecurityGroupId": "sg-01c84d09d70f39a7f",
      "vpcId": "vpc-0012b8e1cc0abb17d",
      "endpointPublicAccess": true,
      "endpointPrivateAccess": true,
      "publicAccessCidrs": [
        "22.19.18.2/32"
      ]
    },
    "kubernetesNetworkConfig": {
```

```
    "serviceIpv4Cidr": "10.100.0.0/16",
    "ipFamily": "ipv4"
  },
  "logging": {
    "clusterLogging": [
      {
        "types": [
          "api",
          "audit",
          "authenticator",
          "controllerManager",
          "scheduler"
        ],
        "enabled": true
      }
    ]
  },
  "identity": {
    "oidc": {
      "issuer": "https://oidc.eks.us-east-2.amazonaws.com/id/
JSA79429HJDASKJDJ8223829MNDNASW"
    }
  },
  "status": "ACTIVE",
  "certificateAuthority": {
    "data": "CA_DATA_STRING..."
  },
  "platformVersion": "eks.14",
  "tags": {
    "aws:cloudformation:stack-name": "eksctl-my-eks-cluster-cluster",
    "alpha.eksctl.io/cluster-name": "my-eks-cluster",
    "karpenter.sh/discovery": "my-eks-cluster",
    "aws:cloudformation:stack-id": "arn:aws:cloudformation:us-
east-2:111122223333:stack/eksctl-my-eks-cluster-cluster/e752ea00-e217-11ee-
beae-0a9599c8c7ed",
    "auto-delete": "no",
    "eksctl.cluster.k8s.io/v1alpha1/cluster-name": "my-eks-cluster",
    "EKS-Cluster-Name": "my-eks-cluster",
    "alpha.eksctl.io/cluster-oidc-enabled": "true",
    "aws:cloudformation:logical-id": "ControlPlane",
    "alpha.eksctl.io/eksctl-version": "0.173.0-dev
+a7ee89342.2024-03-01T03:40:57Z",
    "Name": "eksctl-my-eks-cluster-cluster/ControlPlane"
  },
},
```

```
    "health": {
      "issues": []
    },
    "accessConfig": {
      "authenticationMode": "API_AND_CONFIG_MAP"
    }
  }
}
```

- Para API obtener más información, consulta [DescribeCluster](#) la Referencia de AWS CLI comandos.

## describe-fargate-profile

En el siguiente ejemplo de código, se muestra cómo usar `describe-fargate-profile`.

AWS CLI

Describe un perfil de Fargate

El siguiente `describe-fargate-profile` ejemplo describe un perfil de Fargate.

```
aws eks describe-fargate-profile \
  --cluster-name my-eks-cluster \
  --fargate-profile-name my-fargate-profile
```

Salida:

```
{
  "fargateProfile": {
    "fargateProfileName": "my-fargate-profile",
    "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-eks-cluster/my-fargate-profile/96c766ce-43d2-f9c9-954c-647334391198",
    "clusterName": "my-eks-cluster",
    "createdAt": "2024-04-11T10:42:52.486000-04:00",
    "podExecutionRoleArn": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-farga-FargatePodExecutionRole-1htfAaJdJUE0",
    "subnets": [
      "subnet-09d912bb63ef21b9a",
      "subnet-04ad87f71c6e5ab4d",
      "subnet-0e2907431c9988b72"
    ],
  },
}
```

```

    "selectors": [
      {
        "namespace": "prod*",
        "labels": {
          "labelname*?": "*value1"
        }
      },
      {
        "namespace": "*dev*",
        "labels": {
          "labelname*?": "*value*"
        }
      }
    ],
    "status": "ACTIVE",
    "tags": {
      "eks-fargate-profile-key-2": "value-2",
      "eks-fargate-profile-key-1": "value-1"
    }
  }
}

```

- Para API obtener más información, consulte [DescribeFargateProfile](#) la Referencia de AWS CLI comandos.

## describe-identity-provider-config

En el siguiente ejemplo de código, se muestra cómo usar `describe-identity-provider-config`.

### AWS CLI

Describe una configuración de proveedor de identidad asociada a tu Amazon EKS Cluster

El siguiente `describe-identity-provider-config` ejemplo describe una configuración de proveedor de identidad asociada a tu Amazon EKS Cluster.

```

aws eks describe-identity-provider-config \
  --cluster-name my-eks-cluster \
  --identity-provider-config type=oidc,name=my-identity-provider

```

Salida:



```
{
  "identityProviderConfig": {
    "oidc": {
      "identityProviderConfigName": "my-identity-provider",
      "identityProviderConfigArn": "arn:aws:eks:us-
east-2:111122223333:identityproviderconfig/my-eks-cluster/oidc/my-identity-
provider/8ac76722-78e4-cec1-ed76-d49eea058622",
      "clusterName": "my-eks-cluster",
      "issuerUrl": "https://oidc.eks.us-east-2.amazonaws.com/
id/38D6A4619A0A69E342B113ED7F1A7652",
      "clientId": "kubernetes",
      "usernameClaim": "email",
      "usernamePrefix": "my-username-prefix",
      "groupsClaim": "my-claim",
      "groupsPrefix": "my-groups-prefix",
      "requiredClaims": {
        "Claim1": "value1",
        "Claim2": "value2"
      },
      "tags": {
        "env": "dev"
      },
      "status": "ACTIVE"
    }
  }
}
```

Para obtener más información, consulte [Autenticar a los usuarios de su clúster desde un proveedor de identidad de OpenID Connect](#) en la Guía del usuario de EKS Amazon.

- Para API obtener más información, consulte la Referencia [DescribeIdentityProviderConfig](#) de AWS CLI comandos.

## describe-nodegroup

En el siguiente ejemplo de código, se muestra cómo usar describe-nodegroup.

### AWS CLI

Describa un grupo de nodos gestionado para un EKS clúster de Amazon

El siguiente describe-nodegroup ejemplo describe un grupo de nodos gestionado para un EKS clúster de Amazon.

```
aws eks describe-nodegroup \  
  --cluster-name my-eks-cluster \  
  --nodegroup-name my-eks-nodegroup
```

Salida:

```
{  
  "nodegroup": {  
    "nodegroupName": "my-eks-nodegroup",  
    "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-  
cluster/my-eks-nodegroup/a8c75f2f-df78-a72f-4063-4b69af3de5b1",  
    "clusterName": "my-eks-cluster",  
    "version": "1.26",  
    "releaseVersion": "1.26.12-20240329",  
    "createdAt": "2024-04-08T11:42:10.555000-04:00",  
    "modifiedAt": "2024-04-08T11:44:12.402000-04:00",  
    "status": "ACTIVE",  
    "capacityType": "ON_DEMAND",  
    "scalingConfig": {  
      "minSize": 1,  
      "maxSize": 3,  
      "desiredSize": 1  
    },  
    "instanceTypes": [  
      "t3.medium"  
    ],  
    "subnets": [  
      "subnet-0e2907431c9988b72",  
      "subnet-04ad87f71c6e5ab4d",  
      "subnet-09d912bb63ef21b9a"  
    ],  
    "amiType": "AL2_x86_64",  
    "nodeRole": "arn:aws:iam::111122223333:role/role-name",  
    "labels": {},  
    "resources": {  
      "autoScalingGroups": [  
        {  
          "name": "eks-my-eks-nodegroup-a8c75f2f-df78-  
a72f-4063-4b69af3de5b1"  
        }  
      ]  
    },  
    "diskSize": 20,  
  }  
}
```

```
    "health": {
      "issues": []
    },
    "updateConfig": {
      "maxUnavailable": 1
    },
    "tags": {}
  }
}
```

- Para API obtener más información, consulte [DescribeNodegroup](#) la Referencia de AWS CLI comandos.

## describe-update

En el siguiente ejemplo de código, se muestra cómo usar `describe-update`.

### AWS CLI

Ejemplo 1: Para describir una actualización de un clúster

En el siguiente `describe-update` ejemplo se describe una actualización para un clúster denominado.

```
aws eks describe-update \
  --name my-eks-cluster \
  --update-id 10bddb13-a71b-425a-b0a6-71cd03e59161
```

Salida:

```
{
  "update": {
    "id": "10bddb13-a71b-425a-b0a6-71cd03e59161",
    "status": "Successful",
    "type": "EndpointAccessUpdate",
    "params": [
      {
        "type": "EndpointPublicAccess",
        "value": "false"
      },
      {
        "type": "EndpointPrivateAccess",
```

```

        "value": "true"
      }
    ],
    "createdAt": "2024-03-14T10:01:26.297000-04:00",
    "errors": []
  }
}

```

Para obtener más información, consulte [Actualización de una versión de Kubernetes de un EKS clúster de Amazon en la Guía](#) del usuario de Amazon EKS.

Ejemplo 2: Para describir una actualización de un clúster

En el siguiente describe-update ejemplo se describe una actualización para un clúster denominado.

```

aws eks describe-update \
  --name my-eks-cluster \
  --update-id e4994991-4c0f-475a-a040-427e6da52966

```

Salida:

```

{
  "update": {
    "id": "e4994991-4c0f-475a-a040-427e6da52966",
    "status": "Successful",
    "type": "AssociateEncryptionConfig",
    "params": [
      {
        "type": "EncryptionConfig",
        "value": "[{\"resources\":[\"secrets\"],\"provider\":{\"keyArn\":
\\\"arn:aws:kms:region-code:account:key/key\\\"}]]"
      }
    ],
    "createdAt": "2024-03-14T11:01:26.297000-04:00",
    "errors": []
  }
}

```

Para obtener más información, consulte [Actualización de una versión de Kubernetes de un EKS clúster de Amazon en la Guía](#) del usuario de Amazon EKS.

### Ejemplo 3: Para describir una actualización de un clúster

En el siguiente `describe-update` ejemplo se describe una actualización para un clúster denominado.

```
aws eks describe-update \  
  --name my-eks-cluster \  
  --update-id b5f0ba18-9a87-4450-b5a0-825e6e84496f
```

Salida:

```
{  
  "update": {  
    "id": "b5f0ba18-9a87-4450-b5a0-825e6e84496f",  
    "status": "Successful",  
    "type": "VersionUpdate",  
    "params": [  
      {  
        "type": "Version",  
        "value": "1.29"  
      },  
      {  
        "type": "PlatformVersion",  
        "value": "eks.1"  
      }  
    ],  
    "createdAt": "2024-03-14T12:05:26.297000-04:00",  
    "errors": []  
  }  
}
```

Para obtener más información, consulte [Actualización de una versión de Kubernetes de un EKS clúster de Amazon en la Guía](#) del usuario de Amazon EKS.

- Para API obtener más información, consulte la Referencia de [DescribeUpdate](#) comandos AWS CLI .

### **disassociate-identity-provider-config**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-identity-provider-config`.

## AWS CLI

Desvincular el proveedor de identidad a tu Amazon Cluster EKS

En el siguiente `disassociate-identity-provider-config` ejemplo, se desvincula un proveedor de identidad de tu Amazon EKS Cluster.

```
aws eks disassociate-identity-provider-config \  
  --cluster-name my-eks-cluster \  
  --identity-provider-config 'type=oidc,name=my-identity-provider'
```

Salida:

```
{  
  "update": {  
    "id": "5f78d14e-c57b-4857-a3e4-cf664ae20949",  
    "status": "InProgress",  
    "type": "DisassociateIdentityProviderConfig",  
    "params": [  
      {  
        "type": "IdentityProviderConfig",  
        "value": "[]"  
      }  
    ],  
    "createdAt": "2024-04-11T13:53:43.314000-04:00",  
    "errors": []  
  }  
}
```

Para obtener más información, consulte [Autenticar a los usuarios de su clúster desde un proveedor de identidades de OpenID Connect: OIDC desasociar un proveedor de identidad de su clúster en la Guía del](#) usuario de EKS Amazon.

- Para API obtener más información, consulte la Referencia de [DisassociateIdentityProviderConfig](#) comandos AWS CLI .

## get-token

En el siguiente ejemplo de código, se muestra cómo usar `get-token`.

## AWS CLI

Ejemplo 1: Obtener un token de autenticación para un EKS clúster de Amazon llamado `my-eks-cluster`

En el siguiente `get-token` ejemplo, se obtiene un token de autenticación para un EKS clúster de Amazon denominado `my-eks-cluster`.

```
aws eks get-token \  
  --cluster-name my-eks-cluster
```

Salida:

```
{  
  "kind": "ExecCredential",  
  "apiVersion": "client.authentication.k8s.io/v1beta1",  
  "spec": {},  
  "status": {  
    "expirationTimestamp": "2024-04-11T20:59:56Z",  
    "token": "k8s-aws-v1.EXAMPLE_TOKEN_DATA_STRING..."  
  }  
}
```

Ejemplo 2: obtiene un token de autenticación para un EKS clúster de Amazon llamado `my-eks-cluster` asumiendo esta función ARN para las credenciales al firmar el token

El siguiente `get-token` ejemplo obtiene un token de autenticación para un EKS clúster de Amazon denominado `my-eks-cluster` asumiendo esta función ARN para las credenciales al firmar el token.

```
aws eks get-token \  
  --cluster-name my-eks-cluster \  
  --role-arn arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-cluster-ServiceRole-j1k7AfTIQtnM
```

Salida:

```
{  
  "kind": "ExecCredential",  
  "apiVersion": "client.authentication.k8s.io/v1beta1",  
  "spec": {},  
  "status": {
```

```
    "expirationTimestamp": "2024-04-11T21:05:26Z",
    "token": "k8s-aws-v1.EXAMPLE_TOKEN_DATA_STRING..."
  }
}
```

- Para API obtener más información, consulte [GetToken](#) la Referencia de AWS CLI comandos.

## list-addons

En el siguiente ejemplo de código, se muestra cómo usar `list-addons`.

### AWS CLI

Enumera todos los complementos instalados en tu EKS clúster de Amazon con el nombre ``my-eks-cluster``

El siguiente `list-addons` ejemplo muestra todos los complementos instalados en tu EKS clúster de Amazon denominados `my-eks-cluster`.

```
aws eks list-addons \
  --cluster-name my-eks-cluster
```

Salida:

```
{
  "addons": [
    "kube-proxy",
    "vpc-cni"
  ]
}
```

- Para API obtener más información, consulte [ListAddons](#) la Referencia de AWS CLI comandos.

## list-clusters

En el siguiente ejemplo de código, se muestra cómo usar `list-clusters`.

### AWS CLI

Para enumerar todos los complementos instalados en tu EKS clúster de Amazon con el nombre ``my-eks-cluster``



El siguiente `list-clusters` ejemplo muestra todos los complementos instalados en tu EKS clúster de Amazon denominados `my-eks-cluster`.

```
aws eks list-clusters
```

Salida:

```
{
  "clusters": [
    "prod",
    "qa",
    "stage",
    "my-eks-cluster"
  ]
}
```

- Para API obtener más información, consulte [ListClusters](#) la Referencia de AWS CLI comandos.

## `list-fargate-profiles`

En el siguiente ejemplo de código, se muestra cómo usar `list-fargate-profiles`.

### AWS CLI

Para enumerar todos los perfiles de Fargate de tu EKS clúster de Amazon llamado `my-eks-cluster`

En el siguiente `list-fargate-profiles` ejemplo, se enumeran todos los perfiles de Fargate de tu EKS clúster de Amazon denominados `my-eks-cluster`.

```
aws eks list-fargate-profiles \
  --cluster-name my-eks-cluster
```

Salida:

```
{
  "fargateProfileNames": [
    "my-fargate-profile"
  ]
}
```

- Para API obtener más información, consulte [ListFargateProfiles](#) la Referencia de AWS CLI comandos.

## **list-identity-provider-configs**

En el siguiente ejemplo de código, se muestra cómo usar `list-identity-provider-configs`.

### AWS CLI

Listar los proveedores de identidad asociados a un Amazon EKS Cluster

El siguiente `list-identity-provider-configs` ejemplo muestra el proveedor de identidad asociado a un Amazon EKS Cluster.

```
aws eks list-identity-provider-configs \  
  --cluster-name my-eks-cluster
```

Salida:

```
{  
  "identityProviderConfigs": [  
    {  
      "type": "oidc",  
      "name": "my-identity-provider"  
    }  
  ]  
}
```

Para obtener más información, consulte [Autenticar a los usuarios de su clúster desde un proveedor de identidad de OpenID Connect](#) en la Guía del usuario de EKS Amazon.

- Para API obtener más información, consulte la Referencia [ListIdentityProviderConfigs](#) de AWS CLI comandos.

## **list-nodegroups**

En el siguiente ejemplo de código, se muestra cómo usar `list-nodegroups`.

### AWS CLI

Listar todos los grupos de nodos de un EKS clúster de Amazon

En el siguiente `list-nodegroups` ejemplo, se enumeran todos los grupos de nodos de un EKS clúster de Amazon.

```
aws eks list-nodegroups \  
  --cluster-name my-eks-cluster
```

Salida:

```
{  
  "nodegroups": [  
    "my-eks-managed-node-group",  
    "my-eks-nodegroup"  
  ]  
}
```

- Para API obtener más información, consulte [ListNodegroups](#) la Referencia de AWS CLI comandos.

## **list-tags-for-resource**

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

AWS CLI

Ejemplo 1: Para enumerar todas las etiquetas de un Amazon EKS Cluster ARN

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas de un Amazon EKS ClusterARN.

```
aws eks list-tags-for-resource \  
  --resource-arn arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster
```

Salida:

```
{  
  "tags": {  
    "aws:cloudformation:stack-name": "eksctl-my-eks-cluster-cluster",  
    "alpha.eksctl.io/cluster-name": "my-eks-cluster",  
    "karpenter.sh/discovery": "my-eks-cluster",
```

```

    "aws:cloudformation:stack-id": "arn:aws:cloudformation:us-
east-2:111122223333:stack/eksctl-my-eks-cluster-cluster/e752ea00-e217-11ee-
beae-0a9599c8c7ed",
    "auto-delete": "no",
    "eksctl.cluster.k8s.io/v1alpha1/cluster-name": "my-eks-cluster",
    "EKS-Cluster-Name": "my-eks-cluster",
    "alpha.eksctl.io/cluster-oidc-enabled": "true",
    "aws:cloudformation:logical-id": "ControlPlane",
    "alpha.eksctl.io/eksctl-version": "0.173.0-dev
+a7ee89342.2024-03-01T03:40:57Z",
    "Name": "eksctl-my-eks-cluster-cluster/ControlPlane"
  }
}

```

Ejemplo 2: Para enumerar todas las etiquetas de un grupo de Amazon EKS Node ARN

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas de un grupo de Amazon EKS NodeARN.

```

aws eks list-tags-for-resource \
  --resource-arn arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-cluster/my-
eks-managed-node-group/60c71ed2-2cfb-020f-a5f4-ad32477f198c

```

Salida:

```

{
  "tags": {
    "aws:cloudformation:stack-name": "eksctl-my-eks-cluster-nodegroup-my-eks-
managed-node-group",
    "aws:cloudformation:stack-id": "arn:aws:cloudformation:us-
east-2:111122223333:stack/eksctl-my-eks-cluster-nodegroup-my-eks-managed-node-group/
eaa20310-e219-11ee-b851-0ab9ad8228ff",
    "eksctl.cluster.k8s.io/v1alpha1/cluster-name": "my-eks-cluster",
    "EKS-Cluster-Name": "my-eks-cluster",
    "alpha.eksctl.io/nodegroup-type": "managed",
    "NodeGroup Name 1": "my-eks-managed-node-group",
    "k8s.io/cluster-autoscaler/enabled": "true",
    "nodegroup-role": "worker",
    "alpha.eksctl.io/cluster-name": "my-eks-cluster",
    "alpha.eksctl.io/nodegroup-name": "my-eks-managed-node-group",
    "karpenter.sh/discovery": "my-eks-cluster",
    "NodeGroup Name 2": "AmazonLinux-Linux-Managed-NG-v1-26-v1",
    "auto-delete": "no",
  }
}

```

```

        "k8s.io/cluster-autoscaler/my-eks-cluster": "owned",
        "aws:cloudformation:logical-id": "ManagedNodeGroup",
        "alpha.eksctl.io/eksctl-version": "0.173.0-dev
+7ee89342.2024-03-01T03:40:57Z"
    }
}

```

Ejemplo 3: Para enumerar todas las etiquetas de un perfil de Amazon EKS Fargate ARN

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas de un perfil de Amazon EKS Fargate. ARN

```

aws eks list-tags-for-resource \
  --resource-arn arn:aws:eks:us-east-2:111122223333:fargateprofile/my-eks-cluster/my-fargate-profile/d6c76780-e541-0725-c816-36754cab734b

```

Salida:

```

{
  "tags": {
    "eks-fargate-profile-key-2": "value-2",
    "eks-fargate-profile-key-1": "value-1"
  }
}

```

Ejemplo 4: Para enumerar todas las etiquetas de un EKS complemento de Amazon ARN

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas de un EKS complemento de Amazon ARN.

```

aws eks list-tags-for-resource \
  --resource-arn arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/vpc-cni/0ec71efc-98dd-3203-60b0-4b939b2a5e5f

```

Salida:

```

{
  "tags": {
    "eks-addon-key-2": "value-2",
    "eks-addon-key-1": "value-1"
  }
}

```

```
}
```

Ejemplo 5: Para enumerar todas las etiquetas de un proveedor de EKS OIDC identidad de Amazon ARN

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas de un proveedor de EKS OIDC identidad de AmazonARN.

```
aws eks list-tags-for-resource \  
  --resource-arn arn:aws:eks:us-east-2:111122223333:identityproviderconfig/my-eks-  
cluster/oidc/my-identity-provider/8ac76722-78e4-cec1-ed76-d49eea058622
```

Salida:

```
{  
  "tags": {  
    "my-identity-provider": "test"  
  }  
}
```

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## list-update

En el siguiente ejemplo de código, se muestra cómo usar `list-update`.

AWS CLI

Ejemplo 1: Para enumerar las actualizaciones asociadas a un nombre de Amazon EKS Cluster

En el siguiente `list-updates` ejemplo, se enumeran todas las actualizaciones IDs de un nombre de Amazon EKS Cluster.

```
aws eks list-updates \  
  --name my-eks-cluster
```

Salida:

```
{  
  "updateIds": [  
    "
```

```
    "5f78d14e-c57b-4857-a3e4-cf664ae20949",  
    "760e5a3f-adad-48c7-88d3-7ac283c09c26",  
    "cd4ec863-bc55-47d5-a377-3971502f529b",  
    "f12657ce-e869-4f17-b158-a82ab8b7d937"  
  ]  
}
```

Ejemplo 2: Para enumerar todas las actualizaciones IDs de un grupo de Amazon EKS Node

En el siguiente `list-updates` ejemplo, se enumeran todas las actualizaciones IDs de un grupo de Amazon EKS Node.

```
aws eks list-updates \  
  --name my-eks-cluster \  
  --nodegroup-name my-eks-managed-node-group
```

Salida:

```
{  
  "updateIds": [  
    "8c6c1bef-61fe-42ac-a242-89412387b8e7"  
  ]  
}
```

Ejemplo 3: Para publicar todas las actualizaciones IDs en un Amazon EKS Add-one

En el siguiente `list-updates` ejemplo, se enumeran todas las actualizaciones IDs de un EKS complemento de Amazon.

```
aws eks list-updates \  
  --name my-eks-cluster \  
  --addon-name vpc-cni
```

Salida:

```
{  
  "updateIds": [  
    "9cdba8d4-79fb-3c83-afe8-00b508d33268"  
  ]  
}
```

- Para API obtener más información, consulte [ListUpdate](#) la Referencia de AWS CLI comandos.

## list-updates

En el siguiente ejemplo de código, se muestra cómo usar `list-updates`.

### AWS CLI

Para ver una lista de las actualizaciones de un clúster

Este comando de ejemplo muestra las actualizaciones actuales de un clúster nombrado `example` en la región predeterminada.

Comando:

```
aws eks list-updates --name example
```

Salida:

```
{
  "updateIds": [
    "10bddb13-a71b-425a-b0a6-71cd03e59161"
  ]
}
```

- Para API obtener más información, consulte [ListUpdates](#) la Referencia de AWS CLI comandos.

## register-cluster

En el siguiente ejemplo de código, se muestra cómo usar `register-cluster`.

### AWS CLI

Ejemplo 1: Registrar un clúster de EKS \_ ANYWHERE Kubernetes externo en Amazon EKS

En el siguiente `register-cluster` ejemplo, se registra un clúster de EKS \_ ANYWHERE Kubernetes externo en Amazon. EKS

```
aws eks register-cluster \
  --name my-eks-anywhere-cluster \
  --connector-config 'roleArn=arn:aws:iam::111122223333:role/AmazonEKSCollectorAgentRole,provider=EKS_ANYWHERE'
```

Salida:



```
{
  "cluster": {
    "name": "my-eks-anywhere-cluster",
    "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-eks-anywhere-cluster",
    "createdAt": "2024-04-12T12:38:37.561000-04:00",
    "status": "PENDING",
    "tags": {},
    "connectorConfig": {
      "activationId": "xxxxxxxxACTIVATION_IDxxxxxxxx",
      "activationCode": "xxxxxxxxACTIVATION_CODExxxxxxxx",
      "activationExpiry": "2024-04-15T12:38:37.082000-04:00",
      "provider": "EKS_ANYWHERE",
      "roleArn": "arn:aws:iam::111122223333:role/AmazonEKSCollectorAgentRole"
    }
  }
}
```

Para obtener más información, consulta [Cómo conectar un clúster externo](#) en la Guía del EKS usuario de Amazon.

Ejemplo 2: Registrar cualquier clúster de Kubernetes externo en Amazon EKS

En el siguiente `register-cluster` ejemplo, se registra un clúster de EKS \_ ANYWHERE Kubernetes externo en Amazon. EKS

```
aws eks register-cluster \
  --name my-eks-anywhere-cluster \
  --connector-config 'roleArn=arn:aws:iam::111122223333:role/AmazonEKSCollectorAgentRole,provider=OTHER'
```

Salida:

```
{
  "cluster": {
    "name": "my-onprem-k8s-cluster",
    "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-onprem-k8s-cluster",
    "createdAt": "2024-04-12T12:42:10.861000-04:00",
    "status": "PENDING",
    "tags": {},
    "connectorConfig": {
      "activationId": "xxxxxxxxACTIVATION_IDxxxxxxxx",
      "activationCode": "xxxxxxxxACTIVATION_CODExxxxxxxx",

```

```
        "activationExpiry": "2024-04-15T12:42:10.339000-04:00",
        "provider": "OTHER",
        "roleArn": "arn:aws:iam::111122223333:role/AmazonEKSCollectorAgentRole"
    }
}
}
```

Para obtener más información, consulta [Cómo conectar un clúster externo](#) en la Guía del EKS usuario de Amazon.

- Para API obtener más información, consulte [RegisterCluster](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Ejemplo 1: Para añadir las etiquetas especificadas a un Amazon EKS Cluster

En el siguiente `tag-resource` ejemplo, se añaden las etiquetas especificadas a un Amazon EKS Cluster.

```
aws eks tag-resource \
  --resource-arn arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster \
  --tag 'my-eks-cluster-test-1=test-value-1,my-eks-cluster-dev-1=dev-value-2'
```

Este comando no genera ninguna salida.

Ejemplo 2: Para añadir las etiquetas especificadas a un grupo de Amazon EKS Node

En el siguiente `tag-resource` ejemplo, se añaden las etiquetas especificadas a un grupo de Amazon EKS Node.

```
aws eks tag-resource \
  --resource-arn arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-cluster/my-eks-managed-node-group/60c71ed2-2cfb-020f-a5f4-ad32477f198c \
  --tag 'my-eks-nodegroup-test-1=test-value-1,my-eks-nodegroup-dev-1=dev-value-2'
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Ejemplo 1: Para eliminar las etiquetas especificadas de un Amazon Cluster EKS

En el siguiente `untag-resource` ejemplo, se eliminan las etiquetas especificadas de un Amazon EKS Cluster.

```
aws eks untag-resource \  
  --resource-arn arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster \  
  --tag-keys "my-eks-cluster-test-1" "my-eks-cluster-dev-1"
```

Este comando no genera ninguna salida.

Ejemplo 2: Para eliminar las etiquetas especificadas de un grupo de Amazon EKS Node

En el siguiente `untag-resource` ejemplo, se eliminan las etiquetas especificadas de un grupo de Amazon EKS Node.

```
aws eks untag-resource \  
  --resource-arn arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-cluster/my-eks-managed-node-group/60c71ed2-2cfb-020f-a5f4-ad32477f198c \  
  --tag-keys "my-eks-nodegroup-test-1" "my-eks-nodegroup-dev-1"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-addon

En el siguiente ejemplo de código, se muestra cómo usar `update-addon`.

### AWS CLI

Ejemplo 1. Para actualizar un EKS complemento de Amazon con el rol de cuenta de servicio ARN

El siguiente comando de `update-addon` ejemplo actualiza un EKS complemento de Amazon con el rol de cuenta de servicioARN.

```
aws eks update-addon \  
  --cluster-name my-eks-cluster \  
  --addon-name vpc-cni \  
  --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-  
addon-vpc-cni-Role1-Yfakrq0C1UTm
```

Salida:

```
{  
  "update": {  
    "id": "c00d2de2-c2e4-3d30-929e-46b8edec2ce4",  
    "status": "InProgress",  
    "type": "AddonUpdate",  
    "params": [  
      {  
        "type": "ServiceAccountRoleArn",  
        "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-  
addon-vpc-cni-Role1-Yfakrq0C1UTm"  
      }  
    ],  
    "updatedAt": "2024-04-12T16:04:55.614000-04:00",  
    "errors": []  
  }  
}
```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: actualización de un complemento](#) en la Guía del EKS usuario de Amazon.

Ejemplo 2. Para actualizar un EKS complemento de Amazon con una versión de complemento específica

El siguiente comando de `update-addon` ejemplo actualiza un EKS complemento de Amazon con una versión de complemento específica.

```
aws eks update-addon \  
  --cluster-name my-eks-cluster \  
  --addon-name vpc-cni \  
  --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-  
addon-vpc-cni-Role1-Yfakrq0C1UTm \  
  --version 1.10.0
```

```
--addon-version v1.16.4-eksbuild.2
```

Salida:

```
{
  "update": {
    "id": "f58dc0b0-2b18-34bd-bc6a-e4abc0011f36",
    "status": "InProgress",
    "type": "AddonUpdate",
    "params": [
      {
        "type": "AddonVersion",
        "value": "v1.16.4-eksbuild.2"
      },
      {
        "type": "ServiceAccountRoleArn",
        "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-Yfakrq0C1UTm"
      }
    ],
    "createdAt": "2024-04-12T16:07:16.550000-04:00",
    "errors": []
  }
}
```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: actualización de un complemento](#) en la Guía del EKS usuario de Amazon.

Ejemplo 3. Para actualizar un EKS complemento de Amazon con valores de configuración personalizados y resolver conflictos: detalles

El siguiente comando de `update-addon` ejemplo actualiza un EKS complemento de Amazon con valores de configuración personalizados y resuelve los detalles de los conflictos.

```
aws eks update-addon \
  --cluster-name my-eks-cluster \
  --addon-name vpc-cni \
  --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-Yfakrq0C1UTm \
  --addon-version v1.16.4-eksbuild.2 \
  --configuration-values '{"resources": {"limits":{"cpu":"100m"}, "requests":
{"cpu":"50m"}}}' \
```

**--resolve-conflicts** *PRESERVE*

Salida:

```
{
  "update": {
    "id": "cd9f2173-a8d8-3004-a90f-032f14326520",
    "status": "InProgress",
    "type": "AddonUpdate",
    "params": [
      {
        "type": "AddonVersion",
        "value": "v1.16.4-eksbuild.2"
      },
      {
        "type": "ServiceAccountRoleArn",
        "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-Yfakrq0C1UTm"
      },
      {
        "type": "ResolveConflicts",
        "value": "PRESERVE"
      },
      {
        "type": "ConfigurationValues",
        "value": "{\"resources\": {\"limits\": {\"cpu\": \"100m\"}, \"requests
\": {\"cpu\": \"50m\"}}}"
      }
    ],
    "createdAt": "2024-04-12T16:16:27.363000-04:00",
    "errors": []
  }
}
```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: actualización de un complemento](#) en la Guía del EKS usuario de Amazon.

Ejemplo 4. Para actualizar un EKS complemento de Amazon con un archivo de valores JSON de configuración personalizados

El siguiente comando de `update-addon` ejemplo actualiza un EKS complemento de Amazon con valores de JSON configuración personalizados y resuelve los detalles de los conflictos.

```
aws eks update-addon \  
  --cluster-name my-eks-cluster \  
  --addon-name vpc-cni \  
  --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-  
addon-vpc-cni-Role1-Yfakrq0C1UTm \  
  --addon-version v1.17.1-eksbuild.1 \  
  --configuration-values 'file://configuration-values.json' \  
  --resolve-conflicts PRESERVE
```

Contenidos de `configuration-values.json`:

```
{  
  "resources": {  
    "limits": {  
      "cpu": "100m"  
    },  
    "requests": {  
      "cpu": "50m"  
    }  
  },  
  "env": {  
    "AWS_VPC_K8S_CNI_LOGLEVEL": "ERROR"  
  }  
}
```

Salida:

```
{  
  "update": {  
    "id": "6881a437-174f-346b-9a63-6e91763507cc",  
    "status": "InProgress",  
    "type": "AddonUpdate",  
    "params": [  
      {  
        "type": "AddonVersion",  
        "value": "v1.17.1-eksbuild.1"  
      },  
      {  
        "type": "ServiceAccountRoleArn",  
        "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-  
addon-vpc-cni-Role1-Yfakrq0C1UTm"  
      }  
    ]  
  }  
}
```

```

    {
      "type": "ResolveConflicts",
      "value": "PRESERVE"
    },
    {
      "type": "ConfigurationValues",
      "value": "{\n  \"resources\": {\n    \"limits\": {\n\n      \"cpu\": \"100m\"\n    },\n    \"requests\": {\n      \"cpu\": \"50m\"\n    }\n  },\n  \"env\": {\n    \"AWS_VPC_K8S_CNI_LOGLEVEL\": \"ERROR\"\n  }\n}"
    }
  ],
  "createdAt": "2024-04-12T16:22:55.519000-04:00",
  "errors": []
}
}

```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: actualización de un complemento](#) en la Guía del EKS usuario de Amazon.

Ejemplo 5. Para actualizar un EKS complemento de Amazon con un archivo de valores YAML de configuración personalizados

El siguiente comando de `update-addon` ejemplo actualiza un EKS complemento de Amazon con valores de YAML configuración personalizados y resuelve los detalles de los conflictos.

```

aws eks update-addon \
  --cluster-name my-eks-cluster \
  --addon-name vpc-cni \
  --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-addon-vpc-cni-Role1-Yfakrq0C1UTm \
  --addon-version v1.18.0-eksbuild.1 \
  --configuration-values 'file://configuration-values.yaml' \
  --resolve-conflicts PRESERVE

```

Contenidos de `configuration-values.yaml`:

```

resources:
  limits:
    cpu: '100m'
  requests:
    cpu: '50m'

```



```
env:
  AWS_VPC_K8S_CNI_LOGLEVEL: 'DEBUG'
```

**Salida:**

```
{
  "update": {
    "id": "a067a4c9-69d0-3769-ace9-d235c5b16701",
    "status": "InProgress",
    "type": "AddonUpdate",
    "params": [
      {
        "type": "AddonVersion",
        "value": "v1.18.0-eksbuild.1"
      },
      {
        "type": "ServiceAccountRoleArn",
        "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-Yfakrq0C1UTm"
      },
      {
        "type": "ResolveConflicts",
        "value": "PRESERVE"
      },
      {
        "type": "ConfigurationValues",
        "value": "resources:\n      limits:\n          cpu: '100m'\n
requests:\n      cpu: '50m'\nenv:\n      AWS_VPC_K8S_CNI_LOGLEVEL: 'DEBUG'"
      }
    ],
    "createdAt": "2024-04-12T16:25:07.212000-04:00",
    "errors": []
  }
}
```

Para obtener más información, consulta [Gestión de EKS complementos de Amazon: actualización de un complemento](#) en la Guía del EKS usuario de Amazon.

- Para API obtener más información, consulte [UpdateAddon](#) la Referencia de AWS CLI comandos.

## update-cluster-config

En el siguiente ejemplo de código, se muestra cómo usar `update-cluster-config`.

### AWS CLI

Para actualizar el acceso al punto final del clúster

Este comando de ejemplo actualiza un clúster para inhabilitar el acceso público a los puntos finales y habilitar el acceso a los puntos finales privados.

Comando:

```
aws eks update-cluster-config --name example \  
--resources-vpc-config endpointPublicAccess=false,endpointPrivateAccess=true
```

Salida:

```
{  
  "update": {  
    "id": "ec883c93-2e9e-407c-a22f-8f6fa6e67d4f",  
    "status": "InProgress",  
    "type": "EndpointAccessUpdate",  
    "params": [  
      {  
        "type": "EndpointPublicAccess",  
        "value": "false"  
      },  
      {  
        "type": "EndpointPrivateAccess",  
        "value": "true"  
      }  
    ],  
    "createdAt": 1565806986.506,  
    "errors": []  
  }  
}
```

Para habilitar el registro de un clúster

Este comando de ejemplo habilita todos los tipos de registro del plano de control del clúster para un clúster denominado `example`.

Comando:

```
aws eks update-cluster-config --name example \
--logging '{"clusterLogging":[{"types":
["api","audit","authenticator","controllerManager","scheduler"],"enabled":true}]}'
```

Salida:

```
{
  "update": {
    "id": "7551c64b-1d27-4b1e-9f8e-c45f056eb6fd",
    "status": "InProgress",
    "type": "LoggingUpdate",
    "params": [
      {
        "type": "ClusterLogging",
        "value": "{\"clusterLogging\":{\"types\":[\"api\",\"audit\",
\"authenticator\",\"controllerManager\",\"scheduler\"],\"enabled\":true}}"
      }
    ],
    "createdAt": 1565807210.37,
    "errors": []
  }
}
```

- Para API obtener más información, consulte [UpdateClusterConfig](#) la Referencia de AWS CLI comandos.

## update-cluster-version

En el siguiente ejemplo de código, se muestra cómo usar `update-cluster-version`.

AWS CLI

Para actualizar un EKS clúster de Amazon denominado `my-eks-cluster` a la versión de Kubernetes especificada

El siguiente `update-cluster-version` ejemplo actualiza un EKS clúster de Amazon a la versión de Kubernetes especificada.

```
aws eks update-cluster-version \
```

```
--name my-eks-cluster \  
--kubernetes-version 1.27
```

Salida:

```
{  
  "update": {  
    "id": "e4091a28-ea14-48fd-a8c7-975aeb469e8a",  
    "status": "InProgress",  
    "type": "VersionUpdate",  
    "params": [  
      {  
        "type": "Version",  
        "value": "1.27"  
      },  
      {  
        "type": "PlatformVersion",  
        "value": "eks.16"  
      }  
    ],  
    "createdAt": "2024-04-12T16:56:01.082000-04:00",  
    "errors": []  
  }  
}
```

Para obtener más información, consulte [Actualización de una versión de Kubernetes de un EKS clúster de Amazon en la Guía](#) del usuario de Amazon EKS.

- Para API obtener más información, consulte la Referencia de [UpdateClusterVersion](#) comandos AWS CLI .

## update-kubeconfig

En el siguiente ejemplo de código, se muestra cómo usar `update-kubeconfig`.

### AWS CLI

Ejemplo 1: configura tu `kubectl` creando o actualizando `kubeconfig` para que puedas conectarte a un Amazon Cluster llamado `EKS my-eks-cluster`

El siguiente `update-kubeconfig` ejemplo configura tu `kubectl` creando o actualizando el `kubeconfig` para que puedas conectarte a un Amazon Cluster llamado. EKS `my-eks-cluster`

```
aws eks update-kubeconfig \  
  --name my-eks-cluster
```

Salida:

```
Updated context arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster in /Users/  
xxx/.kube/config
```

Para obtener más información, consulta [Crear o actualizar un archivo kubeconfig para un EKS clúster de Amazon](#) en la Guía del EKSusuario de Amazon.

Ejemplo 2: configura tu kubectl creando o actualizando kubeconfig (con la opción role-arn para asumir un rol en la autenticación de clústeres) para que puedas conectarte a un Amazon Cluster llamado `` EKS my-eks-cluster

El siguiente update-kubeconfig ejemplo configura tu kubectl creando o actualizando el kubeconfig (con la opción role-arn para que asuma un rol en la autenticación de clústeres) para que puedas conectarte a un Amazon Cluster denominado. EKS my-eks-cluster

```
aws eks update-kubeconfig \  
  --name my-eks-cluster \  
  --role-arn arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-  
cluster-ServiceRole-j1k7AfTIQtnM
```

Salida:

```
Updated context arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster in /Users/  
xxx/.kube/config
```

Para obtener más información, consulta [Crear o actualizar un archivo kubeconfig para un EKS clúster de Amazon](#) en la Guía del EKSusuario de Amazon.

Ejemplo 3: configura tu kubectl creando o actualizando kubeconfig (con la opción role-arn para asumir un rol en la autenticación del clúster junto con un alias de clúster y un alias de usuario personalizados) para que puedas conectarte a un Amazon Cluster llamado `` EKS my-eks-cluster

El siguiente update-kubeconfig ejemplo configura tu kubectl creando o actualizando kubeconfig (con la opción role-arn para asumir un rol en la autenticación del clúster junto con un alias de clúster y un alias de usuario personalizados) para que puedas conectarte a un Amazon Cluster llamado. EKS my-eks-cluster

```
aws eks update-kubeconfig \  
  --name my-eks-cluster \  
  --role-arn arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-  
cluster-ServiceRole-j1k7AfTIQtnM \  
  --alias stage-eks-cluster \  
  --user-alias john
```

Salida:

```
Updated context stage-eks-cluster in /Users/dubaria/.kube/config
```

Para obtener más información, consulta [Crear o actualizar un archivo kubeconfig para un EKS clúster de Amazon](#) en la Guía del EKSusuario de Amazon.

Ejemplo 4: Imprime las entradas del archivo kubeconfig para revisarlas y configura tu kubectl para que puedas conectarte a un Amazon Cluster llamado `` EKS my-eks-cluster

El siguiente update-kubeconfig ejemplo configura tu kubectl creando o actualizando kubeconfig (con la opción role-arn para asumir un rol en la autenticación del clúster junto con un alias de clúster y un alias de usuario personalizados) para que puedas conectarte a un Amazon Cluster llamado. EKS my-eks-cluster

```
aws eks update-kubeconfig \  
  --name my-eks-cluster \  
  --role-arn arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-  
cluster-ServiceRole-j1k7AfTIQtnM \  
  --alias stage-eks-cluster \  
  --user-alias john \  
  --verbose
```

Salida:

```
Updated context stage-eks-cluster in /Users/dubaria/.kube/config  
Entries:  
  
context:  
cluster: arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster  
user: john  
name: stage-eks-cluster  
  
name: john
```

```

user:
exec:
  apiVersion: client.authentication.k8s.io/v1beta1
  args:
  - --region
  - us-east-2
  - eks
  - get-token
  - --cluster-name
  - my-eks-cluster
  - --output
  - json
  - --role
  - arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-cluster-
ServiceRole-j1k7AfTIQtnM
  command: aws

cluster:
certificate-authority-data: xxx_CA_DATA_xxx
server: https://DALSJ343KE23J3RN45653DSKJTT647TYD.y14.us-east-2.eks.amazonaws.com
name: arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster

```

Para obtener más información, consulta [Crear o actualizar un archivo kubeconfig para un EKS clúster de Amazon](#) en la Guía del EKSusuario de Amazon.

- Para API obtener más información, consulte la Referencia [UpdateKubeconfig](#) de AWS CLI comandos.

## update-nodegroup-config

En el siguiente ejemplo de código, se muestra cómo usar `update-nodegroup-config`.

### AWS CLI

Ejemplo 1: Actualizar un grupo de nodos gestionado para añadir nuevas etiquetas y contaminar el nodo de EKS trabajo de un clúster de Amazon EKS

En el siguiente `update-nodegroup-config` ejemplo, se actualiza un grupo de nodos gestionado para añadir nuevas etiquetas y contaminar el nodo de EKS trabajo de un EKS clúster de Amazon.

```

aws eks update-nodegroup-config \
  --cluster-name my-eks-cluster \

```

```
--nodegroup-name my-eks-nodegroup \
--labels 'addOrUpdateLabels={my-eks-nodegroup-label-1=value-1,my-eks-nodegroup-label-2=value-2}' \
--taints 'addOrUpdateTaints=[{key=taint-key-1,value=taint-value-1,effect=NO_EXECUTE}]'
```

Salida:

```
{
  "update": {
    "id": "e66d21d3-bd8b-3ad1-a5aa-b196dc08c7c1",
    "status": "InProgress",
    "type": "ConfigUpdate",
    "params": [
      {
        "type": "LabelsToAdd",
        "value": "{\"my-eks-nodegroup-label-2\":\"value-2\",\"my-eks-nodegroup-label-1\":\"value-1\"}"
      },
      {
        "type": "TaintsToAdd",
        "value": "[{\"effect\":\"NO_EXECUTE\",\"value\":\"taint-value-1\",\"key\":\"taint-key-1\"}]"
      }
    ],
    "createdAt": "2024-04-08T12:05:19.161000-04:00",
    "errors": []
  }
}
```

Para obtener más información, consulta [Actualización de un grupo de nodos gestionado](#) en la Guía del EKS usuario de Amazon.

Ejemplo 2: Actualizar un grupo de nodos gestionado para eliminar las etiquetas y la contaminación del nodo de EKS trabajo de un clúster de Amazon EKS

En el siguiente update-nodegroup-config ejemplo, se actualiza un grupo de nodos gestionado para eliminar las etiquetas y las impurezas del nodo de EKS trabajo de un EKS clúster de Amazon.

```
aws eks update-nodegroup-config \
  --cluster-name my-eks-cluster \
```



```
--nodegroup-name my-eks-nodegroup \  
--labels 'removeLabels=my-eks-nodegroup-label-1, my-eks-nodegroup-label-2' \  
--taints 'removeTaints=[{key=taint-key-1,value=taint-value-1,effect=NO_EXECUTE}]'
```

Salida:

```
{  
  "update": {  
    "id": "67a08692-9e59-3ace-a916-13929f44cec3",  
    "status": "InProgress",  
    "type": "ConfigUpdate",  
    "params": [  
      {  
        "type": "LabelsToRemove",  
        "value": "[\"my-eks-nodegroup-label-1\", \"my-eks-nodegroup-label-2\"]"  
      },  
      {  
        "type": "TaintsToRemove",  
        "value": "[{\"effect\": \"NO_EXECUTE\", \"value\": \"taint-value-1\",  
        \"key\": \"taint-key-1\"}]"  
      }  
    ],  
    "createdAt": "2024-04-08T12:17:31.817000-04:00",  
    "errors": []  
  }  
}
```

Para obtener más información, consulta [Actualización de un grupo de nodos gestionado](#) en la Guía del EKS usuario de Amazon.

Ejemplo 3: Actualizar un grupo de nodos gestionado para eliminar y añadir etiquetas y contaminar el nodo de EKS trabajo de un clúster de Amazon EKS

En el siguiente `update-nodegroup-config` ejemplo, se actualiza un grupo de nodos gestionado para eliminar y añadir etiquetas y manchas al nodo de EKS trabajo de un EKS clúster de Amazon.

```
aws eks update-nodegroup-config \  
--cluster-name my-eks-cluster \  
--nodegroup-name my-eks-nodegroup \  

```

```
--labels 'add0rUpdateLabels={my-eks-nodegroup-new-label-1=new-value-1,my-eks-nodegroup-new-label-2=new-value-2},removeLabels=my-eks-nodegroup-label-1, my-eks-nodegroup-label-2' \
--taints 'add0rUpdateTaints=[{key=taint-new-key-1,value=taint-new-value-1,effect=PREFER_NO_SCHEDULE}],removeTaints=[{key=taint-key-1,value=taint-value-1,effect=NO_EXECUTE}]'
```

Salida:

```
{
  "update": {
    "id": "4a9c8c45-6ac7-3115-be71-d6412a2339b7",
    "status": "InProgress",
    "type": "ConfigUpdate",
    "params": [
      {
        "type": "LabelsToAdd",
        "value": "{\"my-eks-nodegroup-new-label-1\":\"new-value-1\",\"my-eks-nodegroup-new-label-2\":\"new-value-2\"}"
      },
      {
        "type": "LabelsToRemove",
        "value": "[\"my-eks-nodegroup-label-1\",\"my-eks-nodegroup-label-2\"]"
      },
      {
        "type": "TaintsToAdd",
        "value": "[{\"effect\":\"PREFER_NO_SCHEDULE\",\"value\":\"taint-new-value-1\",\"key\":\"taint-new-key-1\"}]"
      },
      {
        "type": "TaintsToRemove",
        "value": "[{\"effect\":\"NO_EXECUTE\",\"value\":\"taint-value-1\",\"key\":\"taint-key-1\"}]"
      }
    ],
    "createdAt": "2024-04-08T12:30:55.486000-04:00",
    "errors": []
  }
}
```

Para obtener más información, consulta [Actualización de un grupo de nodos gestionado](#) en la Guía del EKS usuario de Amazon.

Ejemplo 4: Actualizar un grupo de nodos gestionado para actualizar `scaling-config` y `update-config` para el nodo de EKS trabajo de un clúster de Amazon EKS

El siguiente `update-nodegroup-config` ejemplo actualiza un grupo de nodos gestionado para actualizar `scaling-config` y `update-config` para el nodo de EKS trabajo de un clúster de Amazon EKS

```
aws eks update-nodegroup-config \  
  --cluster-name my-eks-cluster \  
  --nodegroup-name my-eks-nodegroup \  
  --scaling-config minSize=1,maxSize=5,desiredSize=2 \  
  --update-config maxUnavailable=2
```

Salida:

```
{  
  "update": {  
    "id": "a977160f-59bf-3023-805d-c9826e460aea",  
    "status": "InProgress",  
    "type": "ConfigUpdate",  
    "params": [  
      {  
        "type": "MinSize",  
        "value": "1"  
      },  
      {  
        "type": "MaxSize",  
        "value": "5"  
      },  
      {  
        "type": "DesiredSize",  
        "value": "2"  
      },  
      {  
        "type": "MaxUnavailable",  
        "value": "2"  
      }  
    ],  
    "createdAt": "2024-04-08T12:35:17.036000-04:00",  
    "errors": []  
  }  
}
```

Para obtener más información, consulta [Actualización de un grupo de nodos gestionado](#) en la Guía del EKS usuario de Amazon.

- Para API obtener más información, consulte [UpdateNodegroupConfig](#) la Referencia de AWS CLI comandos.

## update-nodegroup-version

En el siguiente ejemplo de código, se muestra cómo usar `update-nodegroup-version`.

### AWS CLI

Ejemplo 1: Actualizar la versión de Kubernetes o la versión de un grupo de AMI nodos gestionado por Amazon EKS

El siguiente `update-nodegroup-version` ejemplo actualiza la versión de Kubernetes o AMI la versión de un grupo de nodos EKS gestionado por Amazon a la última versión disponible para tu clúster de Kubernetes.

```
aws eks update-nodegroup-version \  
  --cluster-name my-eks-cluster \  
  --nodegroup-name my-eks-nodegroup \  
  --no-force
```

Salida:

```
{  
  "update": {  
    "id": "a94ebfc3-6bf8-307a-89e6-7dbaa36421f7",  
    "status": "InProgress",  
    "type": "VersionUpdate",  
    "params": [  
      {  
        "type": "Version",  
        "value": "1.26"  
      },  
      {  
        "type": "ReleaseVersion",  
        "value": "1.26.12-20240329"  
      }  
    ],  
  },  
}
```

```
    "createdAt": "2024-04-08T13:16:00.724000-04:00",
    "errors": []
  }
}
```

Para obtener más información, consulta [Actualización de un grupo de nodos gestionado](#) en la Guía del EKS usuario de Amazon.

Ejemplo 2: Actualizar la versión de Kubernetes o la versión de un grupo de AMI nodos gestionado por Amazon EKS

El siguiente `update-nodegroup-version` ejemplo actualiza la versión de Kubernetes o la versión de un grupo de nodos EKS gestionado por Amazon a la AMI versión de lanzamiento especificadaAMI.

```
aws eks update-nodegroup-version \
  --cluster-name my-eks-cluster \
  --nodegroup-name my-eks-nodegroup \
  --kubernetes-version '1.26' \
  --release-version '1.26.12-20240307' \
  --no-force
```

Salida:

```
{
  "update": {
    "id": "4db06fe1-088d-336b-bdcd-3fdb94995fb7",
    "status": "InProgress",
    "type": "VersionUpdate",
    "params": [
      {
        "type": "Version",
        "value": "1.26"
      },
      {
        "type": "ReleaseVersion",
        "value": "1.26.12-20240307"
      }
    ],
    "createdAt": "2024-04-08T13:13:58.595000-04:00",
    "errors": []
  }
}
```

```
}
```

Para obtener más información, consulta Actualización de un grupo de nodos gestionado: < <https://docs.aws.amazon.com/eks/latest/userguide/update-managed-node-group.html>>` en la Guía del usuario de Amazon. EKS

- Para obtener más información, consulte la Referencia de comandos. API [UpdateNodegroupVersion](#) AWS CLI

## Ejemplos de Elastic Beanstalk usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Elastic Beanstalk.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

#### **abort-environment-update**

En el siguiente ejemplo de código, se muestra cómo usar `abort-environment-update`.

#### AWS CLI

Para abortar una implementación

El siguiente comando anula el despliegue de una versión de aplicación en ejecución para un entorno denominado: `my-env`

```
aws elasticbeanstalk abort-environment-update --environment-name my-env
```

- Para API obtener más información, consulte [AbortEnvironmentUpdate](#) la Referencia de AWS CLI comandos.

## check-dns-availability

En el siguiente ejemplo de código, se muestra cómo usar `check-dns-availability`.

### AWS CLI

Para comprobar la disponibilidad de un CNAME

El siguiente comando comprueba la disponibilidad del subdominio: `my-cname.elasticbeanstalk.com`

```
aws elasticbeanstalk check-dns-availability --cname-prefix my-cname
```

Salida:

```
{
  "Available": true,
  "FullyQualifiedCNAME": "my-cname.elasticbeanstalk.com"
}
```

- Para API obtener más información, consulte [CheckDnsAvailability](#) la Referencia de AWS CLI comandos.

## create-application-version

En el siguiente ejemplo de código, se muestra cómo usar `create-application-version`.

### AWS CLI

Para crear una nueva versión de la aplicación

El siguiente comando crea una nueva versión, la «v1», de una aplicación denominada " MyApp «:

```
aws elasticbeanstalk create-application-version --application-name MyApp
--version-label v1 --description MyAppv1 --source-bundle S3Bucket="my-
bucket",S3Key="sample.war" --auto-create-application
```

La aplicación se creará automáticamente si aún no existe, gracias a la `auto-create-application` opción. El paquete fuente es un archivo `.war` almacenado en un bucket de S3 denominado «`my-bucket`» que contiene la aplicación de ejemplo de Apache Tomcat.

Salida:

```
{
  "ApplicationVersion": {
    "ApplicationName": "MyApp",
    "VersionLabel": "v1",
    "Description": "MyAppv1",
    "DateCreated": "2015-02-03T23:01:25.412Z",
    "DateUpdated": "2015-02-03T23:01:25.412Z",
    "SourceBundle": {
      "S3Bucket": "my-bucket",
      "S3Key": "sample.war"
    }
  }
}
```

- Para API obtener más información, consulte la Referencia de [CreateApplicationVersion](#) comandos AWS CLI .

## create-application

En el siguiente ejemplo de código, se muestra cómo usar `create-application`.

AWS CLI

Para crear una nueva aplicación

El siguiente comando crea una nueva aplicación denominada "MyApp«:

```
aws elasticbeanstalk create-application --application-name MyApp --description "my application"
```

El `create-application` comando solo configura el nombre y la descripción de la aplicación. Para cargar el código fuente de la aplicación, cree una versión inicial de la aplicación utilizando `create-application-version`. `create-application-version` también tiene una `auto-create-application` opción que te permite crear la aplicación y la versión de la aplicación en un solo paso.



Salida:

```
{
  "Application": {
    "ApplicationName": "MyApp",
    "ConfigurationTemplates": [],
    "DateUpdated": "2015-02-12T18:32:21.181Z",
    "Description": "my application",
    "DateCreated": "2015-02-12T18:32:21.181Z"
  }
}
```

- Para API obtener más información, consulte [CreateApplication](#) la Referencia de AWS CLI comandos.

## create-configuration-template

En el siguiente ejemplo de código, se muestra cómo usar `create-configuration-template`.

AWS CLI

Para crear una plantilla de configuración

El siguiente comando crea una plantilla de configuración denominada a `my-app-v1` partir de los ajustes aplicados a un entorno con el identificador `e-1pqsewtp2j`:

```
aws elasticbeanstalk create-configuration-template --application-name my-app --
template-name my-app-v1 --environment-id e-1pqsewtp2j
```

Salida:

```
{
  "ApplicationName": "my-app",
  "TemplateName": "my-app-v1",
  "DateCreated": "2015-08-12T18:40:39Z",
  "DateUpdated": "2015-08-12T18:40:39Z",
  "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8"
}
```

- Para API obtener más información, consulte [CreateConfigurationTemplate](#) la Referencia de AWS CLI comandos.

## create-environment

En el siguiente ejemplo de código, se muestra cómo usar `create-environment`.

### AWS CLI

Para crear un nuevo entorno para una aplicación

El siguiente comando crea un nuevo entorno para la versión «v1» de una aplicación java denominada «my-app»:

```
aws elasticbeanstalk create-environment --application-name my-app --environment-name my-env --cname-prefix my-app --version-label v1 --solution-stack-name "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8"
```

Salida:

```
{
  "ApplicationName": "my-app",
  "EnvironmentName": "my-env",
  "VersionLabel": "v1",
  "Status": "Launching",
  "EnvironmentId": "e-izqpassy4h",
  "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8",
  "CNAME": "my-app.elasticbeanstalk.com",
  "Health": "Grey",
  "Tier": {
    "Type": "Standard",
    "Name": "WebServer",
    "Version": " "
  },
  "DateUpdated": "2015-02-03T23:04:54.479Z",
  "DateCreated": "2015-02-03T23:04:54.479Z"
}
```

`v1` es la etiqueta de una versión de la aplicación con la que se cargó anteriormente. `create-application-version`

Para especificar un JSON archivo para definir las opciones de configuración del entorno

El siguiente `create-environment` comando especifica que se `myoptions.json` debe usar un JSON archivo con ese nombre para anular los valores obtenidos de la pila de soluciones o de la plantilla de configuración:

```
aws elasticbeanstalk create-environment --environment-name sample-env --application-name sampleapp --option-settings file://myoptions.json
```

`myoptions.json` es un JSON objeto que define varios ajustes:

```
[
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "Interval",
    "Value": "15"
  },
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "Timeout",
    "Value": "8"
  },
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "HealthyThreshold",
    "Value": "2"
  },
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "UnhealthyThreshold",
    "Value": "3"
  }
]
```

Para obtener más información, consulte Valores de opciones en la Guía para desarrolladores de AWS Elastic Beanstalk.

- Para API obtener más información, consulte la Referencia de [CreateEnvironment](#) comandos AWS CLI .

## **create-storage-location**

En el siguiente ejemplo de código, se muestra cómo usar `create-storage-location`.

### AWS CLI

Para crear una ubicación de almacenamiento

El siguiente comando crea una ubicación de almacenamiento en Amazon S3:

```
aws elasticbeanstalk create-storage-location
```

Salida:

```
{
  "S3Bucket": "elasticbeanstalk-us-west-2-0123456789012"
}
```

- Para API obtener más información, consulte [CreateStorageLocation](#) la Referencia de AWS CLI comandos.

## delete-application-version

En el siguiente ejemplo de código, se muestra cómo usar `delete-application-version`.

AWS CLI

Para eliminar una versión de la aplicación

El siguiente comando elimina una versión de la aplicación que tenga el nombre `22a0-stage-150819_182129` de una aplicación denominada `my-app`:

```
aws elasticbeanstalk delete-application-version --version-label 22a0-stage-150819_182129 --application-name my-app
```

- Para API obtener más información, consulte [DeleteApplicationVersion](#) la Referencia de AWS CLI comandos.

## delete-application

En el siguiente ejemplo de código, se muestra cómo usar `delete-application`.

AWS CLI

Eliminación de una aplicación

El siguiente comando elimina una aplicación denominada `my-app`:

```
aws elasticbeanstalk delete-application --application-name my-app
```

- Para API obtener más información, consulte [DeleteApplication](#) la Referencia de AWS CLI comandos.

## delete-configuration-template

En el siguiente ejemplo de código, se muestra cómo usar `delete-configuration-template`.

### AWS CLI

Para eliminar una plantilla de configuración

El siguiente comando elimina una plantilla de configuración con el nombre `my-template` de una aplicación denominada `my-app`:

```
aws elasticbeanstalk delete-configuration-template --template-name my-template --application-name my-app
```

- Para API obtener más información, consulte [DeleteConfigurationTemplate](#) la Referencia de AWS CLI comandos.

## delete-environment-configuration

En el siguiente ejemplo de código, se muestra cómo usar `delete-environment-configuration`.

### AWS CLI

Para eliminar un borrador de configuración

El siguiente comando elimina un borrador de configuración para un entorno denominado `my-env`:

```
aws elasticbeanstalk delete-environment-configuration --environment-name my-env --application-name my-app
```

- Para API obtener más información, consulte [DeleteEnvironmentConfiguration](#) la Referencia de AWS CLI comandos.

## describe-application-versions

En el siguiente ejemplo de código, se muestra cómo usar `describe-application-versions`.

### AWS CLI

Para ver información sobre la versión de una aplicación

El siguiente comando recupera información sobre una versión de la aplicación etiquetada `comov2`:

```
aws elasticbeanstalk describe-application-versions --application-name my-app --  
version-label "v2"
```

Salida:

```
{  
  "ApplicationVersions": [  
    {  
      "ApplicationName": "my-app",  
      "VersionLabel": "v2",  
      "Description": "update cover page",  
      "DateCreated": "2015-07-23T01:32:26.079Z",  
      "DateUpdated": "2015-07-23T01:32:26.079Z",  
      "SourceBundle": {  
        "S3Bucket": "elasticbeanstalk-us-west-2-015321684451",  
        "S3Key": "my-app/5026-stage-150723_224258.war"  
      }  
    },  
    {  
      "ApplicationName": "my-app",  
      "VersionLabel": "v1",  
      "Description": "initial version",  
      "DateCreated": "2015-07-23T22:26:10.816Z",  
      "DateUpdated": "2015-07-23T22:26:10.816Z",  
      "SourceBundle": {  
        "S3Bucket": "elasticbeanstalk-us-west-2-015321684451",  
        "S3Key": "my-app/5026-stage-150723_222618.war"  
      }  
    }  
  ]  
}
```

- Para API obtener más información, consulte [DescribeApplicationVersions](#) la Referencia de AWS CLI comandos.

## describe-applications

En el siguiente ejemplo de código, se muestra cómo usar `describe-applications`.

### AWS CLI

Para ver una lista de aplicaciones

El siguiente comando recupera información sobre las aplicaciones de la región actual:

```
aws elasticbeanstalk describe-applications
```

Salida:

```
{
  "Applications": [
    {
      "ApplicationName": "ruby",
      "ConfigurationTemplates": [],
      "DateUpdated": "2015-08-13T21:05:44.376Z",
      "Versions": [
        "Sample Application"
      ],
      "DateCreated": "2015-08-13T21:05:44.376Z"
    },
    {
      "ApplicationName": "pythonsample",
      "Description": "Application created from the EB CLI using \"eb init\"",
      "Versions": [
        "Sample Application"
      ],
      "DateCreated": "2015-08-13T19:05:43.637Z",
      "ConfigurationTemplates": [],
      "DateUpdated": "2015-08-13T19:05:43.637Z"
    },
    {
      "ApplicationName": "nodejs-example",
      "ConfigurationTemplates": [],
      "DateUpdated": "2015-08-06T17:50:02.486Z",
```

```

        "Versions": [
            "add elasticache",
            "First Release"
        ],
        "DateCreated": "2015-08-06T17:50:02.486Z"
    }
]
}

```

- Para API obtener más información, consulte [DescribeApplications](#) la Referencia de AWS CLI comandos.

## describe-configuration-options

En el siguiente ejemplo de código, se muestra cómo usar `describe-configuration-options`.

### AWS CLI

Para ver las opciones de configuración de un entorno

El siguiente comando recupera las descripciones de todas las opciones de configuración disponibles para un entorno denominado `my-env`:

```
aws elasticbeanstalk describe-configuration-options --environment-name my-env --
application-name my-app
```

Salida (abreviada):

```

{
  "Options": [
    {
      "Name": "JVMOptions",
      "UserDefined": false,
      "DefaultValue": "Xms=256m,Xmx=256m,XX:MaxPermSize=64m,JVM Options=",
      "ChangeSeverity": "RestartApplicationServer",
      "Namespace": "aws:cloudformation:template:parameter",
      "ValueType": "KeyValueList"
    },
    {
      "Name": "Interval",
      "UserDefined": false,

```



```

        "DefaultValue": "30",
        "ChangeSeverity": "NoInterruption",
        "Namespace": "aws:elb:healthcheck",
        "MaxValue": 300,
        "MinValue": 5,
        "ValueType": "Scalar"
    },
    ...
    {
        "Name": "LowerThreshold",
        "UserDefined": false,
        "DefaultValue": "2000000",
        "ChangeSeverity": "NoInterruption",
        "Namespace": "aws:autoscaling:trigger",
        "MinValue": 0,
        "ValueType": "Scalar"
    },
    {
        "Name": "ListenerEnabled",
        "UserDefined": false,
        "DefaultValue": "true",
        "ChangeSeverity": "Unknown",
        "Namespace": "aws:elb:listener",
        "ValueType": "Boolean"
    }
]
}

```

Las opciones de configuración disponibles varían según la plataforma y la versión de configuración. Para obtener más información sobre los espacios de nombres y las opciones compatibles, consulte los valores de las opciones en la Guía para desarrolladores de Elastic AWS Beanstalk.

- Para obtener API más información, consulte la Referencia de comandos.

[DescribeConfigurationOptions](#) AWS CLI

## describe-configuration-settings

En el siguiente ejemplo de código, se muestra cómo usar `describe-configuration-settings`.

### AWS CLI

Para ver los ajustes de configuración de un entorno

El siguiente comando recupera los parámetros de configuración de un entorno denominado `my-env`:

```
aws elasticbeanstalk describe-configuration-settings --environment-name my-env --  
application-name my-app
```

Salida (abreviada):

```
{  
  "ConfigurationSettings": [  
    {  
      "ApplicationName": "my-app",  
      "EnvironmentName": "my-env",  
      "Description": "Environment created from the EB CLI using \"eb create  
\\\"\",  
      "DeploymentStatus": "deployed",  
      "DateCreated": "2015-08-13T19:16:25Z",  
      "OptionSettings": [  
        {  
          "OptionName": "Availability Zones",  
          "ResourceName": "AWSEBAutoScalingGroup",  
          "Namespace": "aws:autoscaling:asg",  
          "Value": "Any"  
        },  
        {  
          "OptionName": "Cooldown",  
          "ResourceName": "AWSEBAutoScalingGroup",  
          "Namespace": "aws:autoscaling:asg",  
          "Value": "360"  
        },  
        ...  
        {  
          "OptionName": "ConnectionDrainingTimeout",  
          "ResourceName": "AWSEBLoadBalancer",  
          "Namespace": "aws:elb:policies",  
          "Value": "20"  
        },  
        {  
          "OptionName": "ConnectionSettingIdleTimeout",  
          "ResourceName": "AWSEBLoadBalancer",  
          "Namespace": "aws:elb:policies",  
          "Value": "60"  
        }  
      ]  
    }  
  ]  
}
```

```
    ],
    "DateUpdated": "2015-08-13T23:30:07Z",
    "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8
Java 8"
  }
]
}
```

Para obtener más información sobre los espacios de nombres y las opciones compatibles, consulte los valores de las opciones en la Guía para desarrolladores de Elastic AWS Beanstalk.

- Para obtener API más información, consulte la Referencia de comandos.

[DescribeConfigurationSettings](#) AWS CLI

## describe-environment-health

En el siguiente ejemplo de código, se muestra cómo usar `describe-environment-health`.

### AWS CLI

Para ver el estado del medio ambiente

El siguiente comando recupera la información de estado general de un entorno denominado `my-env`:

```
aws elasticbeanstalk describe-environment-health --environment-name my-env --
attribute-names ALL
```

Salida:

```
{
  "Status": "Ready",
  "EnvironmentName": "my-env",
  "Color": "Green",
  "ApplicationMetrics": {
    "Duration": 10,
    "Latency": {
      "P99": 0.004,
      "P75": 0.002,
      "P90": 0.003,
      "P95": 0.004,
      "P85": 0.003,
      "P10": 0.001,
```

```

        "P999": 0.004,
        "P50": 0.001
    },
    "RequestCount": 45,
    "StatusCodes": {
        "Status3xx": 0,
        "Status2xx": 45,
        "Status5xx": 0,
        "Status4xx": 0
    }
},
"RefreshedAt": "2015-08-20T21:09:18Z",
"HealthStatus": "Ok",
"InstancesHealth": {
    "Info": 0,
    "Ok": 1,
    "Unknown": 0,
    "Severe": 0,
    "Warning": 0,
    "Degraded": 0,
    "NoData": 0,
    "Pending": 0
},
"Causes": []
}

```

La información de salud solo está disponible para entornos con informes de salud mejorados habilitados. Para obtener más información, consulte [Enhanced Health Reporting and Monitoring](#) en la Guía para desarrolladores de AWS Elastic Beanstalk.

- Para API obtener más información, consulte la Referencia de [DescribeEnvironmentHealth](#) comandos AWS CLI .

## **describe-environment-resources**

En el siguiente ejemplo de código, se muestra cómo usar `describe-environment-resources`.

### AWS CLI

Para ver información sobre los AWS recursos de su entorno

El siguiente comando recupera información sobre los recursos de un entorno denominado `my-env`:

```
aws elasticbeanstalk describe-environment-resources --environment-name my-env
```

Salida:

```
{
  "EnvironmentResources": {
    "EnvironmentName": "my-env",
    "AutoScalingGroups": [
      {
        "Name": "awseb-e-qu3fyyjyjs-stack-AWSEBAutoScalingGroup-
QSB2Z088SXZT"
      }
    ],
    "Triggers": [],
    "LoadBalancers": [
      {
        "Name": "awseb-e-q-AWSEBLoa-1EEPZ0K98BIF0"
      }
    ],
    "Queues": [],
    "Instances": [
      {
        "Id": "i-0c91c786"
      }
    ],
    "LaunchConfigurations": [
      {
        "Name": "awseb-e-qu3fyyjyjs-stack-
AWSEBAutoScalingLaunchConfiguration-1UUVQIBC96TQ2"
      }
    ]
  }
}
```

- Para API obtener más información, consulte [DescribeEnvironmentResources](#) la Referencia de AWS CLI comandos.

## describe-environments

En el siguiente ejemplo de código, se muestra cómo usar `describe-environments`.

## AWS CLI

Para ver información sobre un entorno

El siguiente comando recupera información sobre un entorno denominado `my-env`:

```
aws elasticbeanstalk describe-environments --environment-names my-env
```

Salida:

```
{
  "Environments": [
    {
      "ApplicationName": "my-app",
      "EnvironmentName": "my-env",
      "VersionLabel": "7f58-stage-150812_025409",
      "Status": "Ready",
      "EnvironmentId": "e-rpqsewtp2j",
      "EndpointURL": "awseb-e-w-AWSEBLoa-1483140XB0Q4L-109QXY8121.us-
west-2.elb.amazonaws.com",
      "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8
Java 8",
      "CNAME": "my-env.elasticbeanstalk.com",
      "Health": "Green",
      "AbortableOperationInProgress": false,
      "Tier": {
        "Version": " ",
        "Type": "Standard",
        "Name": "WebServer"
      },
      "DateUpdated": "2015-08-12T18:16:55.019Z",
      "DateCreated": "2015-08-07T20:48:49.599Z"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeEnvironments](#) la Referencia de AWS CLI comandos.

## describe-events

En el siguiente ejemplo de código, se muestra cómo usar `describe-events`.

## AWS CLI

Para ver los eventos de un entorno

El siguiente comando recupera los eventos de un entorno denominado `my-env`:

```
aws elasticbeanstalk describe-events --environment-name my-env
```

Salida (abreviada):

```
{
  "Events": [
    {
      "ApplicationName": "my-app",
      "EnvironmentName": "my-env",
      "Message": "Environment health has transitioned from Info to Ok.",
      "EventDate": "2015-08-20T07:06:53.535Z",
      "Severity": "INFO"
    },
    {
      "ApplicationName": "my-app",
      "EnvironmentName": "my-env",
      "Severity": "INFO",
      "RequestId": "b7f3960b-4709-11e5-ba1e-07e16200da41",
      "Message": "Environment update completed successfully.",
      "EventDate": "2015-08-20T07:06:02.049Z"
    },
    ...
    {
      "ApplicationName": "my-app",
      "EnvironmentName": "my-env",
      "Severity": "INFO",
      "RequestId": "ca8dfbf6-41ef-11e5-988b-651aa638f46b",
      "Message": "Using elasticbeanstalk-us-west-2-012445113685 as Amazon S3
storage bucket for environment data.",
      "EventDate": "2015-08-13T19:16:27.561Z"
    },
    {
      "ApplicationName": "my-app",
      "EnvironmentName": "my-env",
      "Severity": "INFO",
      "RequestId": "cdfba8f6-41ef-11e5-988b-65638f41aa6b",
      "Message": "createEnvironment is starting.",

```

```

    "EventDate": "2015-08-13T19:16:26.581Z"
  }
]
}

```

- Para API obtener más información, consulte [DescribeEvents](#) la Referencia de AWS CLI comandos.

## describe-instances-health

En el siguiente ejemplo de código, se muestra cómo usar `describe-instances-health`.

### AWS CLI

Para ver el estado del medio ambiente

El siguiente comando recupera la información de estado de las instancias de un entorno denominado `my-env`:

```
aws elasticbeanstalk describe-instances-health --environment-name my-env --attribute-names ALL
```

Salida:

```

{
  "InstanceHealthList": [
    {
      "InstanceId": "i-08691cc7",
      "ApplicationMetrics": {
        "Duration": 10,
        "Latency": {
          "P99": 0.006,
          "P75": 0.002,
          "P90": 0.004,
          "P95": 0.005,
          "P85": 0.003,
          "P10": 0.0,
          "P999": 0.006,
          "P50": 0.001
        },
        "RequestCount": 48,

```



```
    "StatusCodes": {
      "Status3xx": 0,
      "Status2xx": 47,
      "Status5xx": 0,
      "Status4xx": 1
    }
  },
  "System": {
    "LoadAverage": [
      0.0,
      0.02,
      0.05
    ],
    "CPUUtilization": {
      "SoftIRQ": 0.1,
      "IOWait": 0.2,
      "System": 0.3,
      "Idle": 97.8,
      "User": 1.5,
      "IRQ": 0.0,
      "Nice": 0.1
    }
  },
  "Color": "Green",
  "HealthStatus": "Ok",
  "LaunchedAt": "2015-08-13T19:17:09Z",
  "Causes": []
}
],
"RefreshedAt": "2015-08-20T21:09:08Z"
}
```

La información de salud solo está disponible para entornos con informes de salud mejorados habilitados. Para obtener más información, consulte Enhanced Health Reporting and Monitoring en la Guía para desarrolladores de AWS Elastic Beanstalk.

- Para API obtener más información, consulte la Referencia de [DescribeInstancesHealth](#) comandos AWS CLI .

## **list-available-solution-stacks**

En el siguiente ejemplo de código, se muestra cómo usar `list-available-solution-stacks`.

## AWS CLI

Para ver las pilas de soluciones

El siguiente comando muestra las pilas de soluciones para todas las configuraciones de plataforma disponibles actualmente y para todas las que haya utilizado en el pasado:

```
aws elasticbeanstalk list-available-solution-stacks
```

Salida (abreviada):

```
{
  "SolutionStacks": [
    "64bit Amazon Linux 2015.03 v2.0.0 running Node.js",
    "64bit Amazon Linux 2015.03 v2.0.0 running PHP 5.6",
    "64bit Amazon Linux 2015.03 v2.0.0 running PHP 5.5",
    "64bit Amazon Linux 2015.03 v2.0.0 running PHP 5.4",
    "64bit Amazon Linux 2015.03 v2.0.0 running Python 3.4",
    "64bit Amazon Linux 2015.03 v2.0.0 running Python 2.7",
    "64bit Amazon Linux 2015.03 v2.0.0 running Python",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.2 (Puma)",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.2 (Passenger Standalone)",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.1 (Puma)",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.1 (Passenger Standalone)",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.0 (Puma)",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.0 (Passenger Standalone)",
    "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 1.9.3",
    "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8",
    "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 7 Java 7",
    "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 7 Java 6",
    "64bit Windows Server Core 2012 R2 running IIS 8.5",
    "64bit Windows Server 2012 R2 running IIS 8.5",
    "64bit Windows Server 2012 running IIS 8",
    "64bit Windows Server 2008 R2 running IIS 7.5",
    "64bit Amazon Linux 2015.03 v2.0.0 running Docker 1.6.2",
    "64bit Amazon Linux 2015.03 v2.0.0 running Multi-container Docker 1.6.2
    (Generic)",
    "64bit Debian jessie v2.0.0 running GlassFish 4.1 Java 8 (Preconfigured -
    Docker)",
    "64bit Debian jessie v2.0.0 running GlassFish 4.0 Java 7 (Preconfigured -
    Docker)",
    "64bit Debian jessie v2.0.0 running Go 1.4 (Preconfigured - Docker)",
    "64bit Debian jessie v2.0.0 running Go 1.3 (Preconfigured - Docker)",
```

```
    "64bit Debian jessie v2.0.0 running Python 3.4 (Preconfigured - Docker)",
  ],
  "SolutionStackDetails": [
    {
      "PermittedFileTypes": [
        "zip"
      ],
      "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Node.js"
    },
    ...
  ]
}
```

- Para API obtener más información, consulte [ListAvailableSolutionStacks](#) la Referencia de AWS CLI comandos.

## rebuild-environment

En el siguiente ejemplo de código, se muestra cómo usar `rebuild-environment`.

### AWS CLI

Para reconstruir un entorno

El siguiente comando finaliza y vuelve a crear los recursos en un entorno denominado: `my-env`

```
aws elasticbeanstalk rebuild-environment --environment-name my-env
```

- Para API obtener más información, consulte la Referencia [RebuildEnvironment](#) de AWS CLI comandos.

## request-environment-info

En el siguiente ejemplo de código, se muestra cómo usar `request-environment-info`.

### AWS CLI

Para solicitar registros con cola

El siguiente comando solicita registros de un entorno denominado `my-env`:

```
aws elasticbeanstalk request-environment-info --environment-name my-env --info-type tail
```

Tras solicitar los registros, recupere su ubicación con `retrieve-environment-info`.

- Para API obtener más información, consulte [RequestEnvironmentInfo](#) la Referencia de AWS CLI comandos.

## restart-app-server

En el siguiente ejemplo de código, se muestra cómo usar `restart-app-server`.

### AWS CLI

Para reiniciar los servidores de aplicaciones

El siguiente comando reinicia los servidores de aplicaciones en todas las instancias de un entorno denominado `my-env`:

```
aws elasticbeanstalk restart-app-server --environment-name my-env
```

- Para API obtener más información, consulte [RestartAppServer](#) la Referencia de AWS CLI comandos.

## retrieve-environment-info

En el siguiente ejemplo de código, se muestra cómo usar `retrieve-environment-info`.

### AWS CLI

Para recuperar troncos con cola

El siguiente comando recupera un enlace a los registros de un entorno denominado: `my-env`

```
aws elasticbeanstalk retrieve-environment-info --environment-name my-env --info-type tail
```

Salida:

```
{
  "EnvironmentInfo": [
```

```
{
  "SampleTimestamp": "2015-08-20T22:23:17.703Z",
  "Message": "https://elasticbeanstalk-us-
west-2-0123456789012.s3.amazonaws.com/resources/environments/
logs/tail/e-fyqyju3yjs/i-09c1c867/TailLogs-1440109397703.out?
AWSAccessKeyId=AKGPT4J56IAJ2EUBL5CQ&Expires=1440195891&Signature=n
%2BEa10V6A2HI0x4Rcfb7LT16bBM%3D",
  "InfoType": "tail",
  "Ec2InstanceId": "i-09c1c867"
}
]
```

Vea el enlace en un navegador. Antes de la recuperación, los registros deben solicitarse a request-environment-info.

- Para API obtener más información, consulte [RetrieveEnvironmentInfo](#) la Referencia de AWS CLI comandos.

## swap-environment-cnames

En el siguiente ejemplo de código, se muestra cómo usar swap-environment-cnames.

### AWS CLI

Para cambiar el entorno CNAMEs

El siguiente comando intercambia los subdominios asignados de dos entornos:

```
aws elasticbeanstalk swap-environment-cnames --source-environment-name my-env-blue
--destination-environment-name my-env-green
```

- Para API obtener más información, consulte la Referencia [SwapEnvironmentCnames](#) de AWS CLI comandos.

## terminate-environment

En el siguiente ejemplo de código, se muestra cómo usar terminate-environment.

### AWS CLI

Para terminar un entorno

El siguiente comando finaliza un entorno de Elastic Beanstalk denominado: `my-env`

```
aws elasticbeanstalk terminate-environment --environment-name my-env
```

Salida:

```
{
  "ApplicationName": "my-app",
  "EnvironmentName": "my-env",
  "Status": "Terminating",
  "EnvironmentId": "e-fh2eravpns",
  "EndpointURL": "awseb-e-f-AWSEBLoa-1I9XUMP4-8492WNUP202574.us-
west-2.elb.amazonaws.com",
  "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java
8",
  "CNAME": "my-env.elasticbeanstalk.com",
  "Health": "Grey",
  "AbortableOperationInProgress": false,
  "Tier": {
    "Version": " ",
    "Type": "Standard",
    "Name": "WebServer"
  },
  "DateUpdated": "2015-08-12T19:05:54.744Z",
  "DateCreated": "2015-08-12T18:52:53.622Z"
}
```

- Para obtener API más información, consulte la Referencia de comandos [TerminateEnvironment](#).AWS CLI

## update-application-version

En el siguiente ejemplo de código, se muestra cómo usar `update-application-version`.

AWS CLI

Para cambiar la descripción de una versión de la aplicación

El siguiente comando actualiza la descripción de una versión de la aplicación denominada `22a0-stage-150819_185942`:

```
aws elasticbeanstalk update-application-version --version-label 22a0-stage-150819_185942 --application-name my-app --description "new description"
```

Salida:

```
{
  "ApplicationVersion": {
    "ApplicationName": "my-app",
    "VersionLabel": "22a0-stage-150819_185942",
    "Description": "new description",
    "DateCreated": "2015-08-19T18:59:17.646Z",
    "DateUpdated": "2015-08-20T22:53:28.871Z",
    "SourceBundle": {
      "S3Bucket": "elasticbeanstalk-us-west-2-0123456789012",
      "S3Key": "my-app/22a0-stage-150819_185942.war"
    }
  }
}
```

- Para API obtener más información, consulte [UpdateApplicationVersion](#) la Referencia de AWS CLI comandos.

## update-application

En el siguiente ejemplo de código, se muestra cómo usar `update-application`.

AWS CLI

Para cambiar la descripción de una aplicación

El siguiente comando actualiza la descripción de una aplicación denominada `my-app`:

```
aws elasticbeanstalk update-application --application-name my-app --description "my Elastic Beanstalk application"
```

Salida:

```
{
  "Application": {
    "ApplicationName": "my-app",
    "Description": "my Elastic Beanstalk application",
  }
}
```

```

    "Versions": [
      "2fba-stage-150819_234450",
      "bf07-stage-150820_214945",
      "93f8",
      "fd7c-stage-150820_000431",
      "22a0-stage-150819_185942"
    ],
    "DateCreated": "2015-08-13T19:15:50.449Z",
    "ConfigurationTemplates": [],
    "DateUpdated": "2015-08-20T22:34:56.195Z"
  }
}

```

- Para API obtener más información, consulte [UpdateApplication](#) la Referencia de AWS CLI comandos.

## update-configuration-template

En el siguiente ejemplo de código, se muestra cómo usar `update-configuration-template`.

### AWS CLI

Para actualizar una plantilla de configuración

El siguiente comando elimina la configuración de métricas de estado CloudWatch personalizadas configurada ConfigDocument de una plantilla de configuración guardada denominada `my-template`:

```

aws elasticbeanstalk update-configuration-template --template-
name my-template --application-name my-app --options-to-
remove Namespace=aws:elasticbeanstalk:healthreporting:system,OptionName=ConfigDocument

```

Salida:

```

{
  "ApplicationName": "my-app",
  "TemplateName": "my-template",
  "DateCreated": "2015-08-20T22:39:31Z",
  "DateUpdated": "2015-08-20T22:43:11Z",
  "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8"
}

```



Para obtener más información sobre los espacios de nombres y las opciones compatibles, consulte los valores de las opciones en la Guía para desarrolladores de Elastic AWS Beanstalk.

- Para obtener API más información, consulte la Referencia de comandos.

[UpdateConfigurationTemplate](#) AWS CLI

## update-environment

En el siguiente ejemplo de código, se muestra cómo usar `update-environment`.

### AWS CLI

Para actualizar un entorno a una nueva versión

El siguiente comando actualiza un entorno denominado «my-env» a la versión «v2» de la aplicación a la que pertenece:

```
aws elasticbeanstalk update-environment --environment-name my-env --version-label v2
```

Este comando requiere que el entorno «my-env» ya exista y pertenezca a una aplicación que tenga una versión de aplicación válida con la etiqueta «v2».

Salida:

```
{
  "ApplicationName": "my-app",
  "EnvironmentName": "my-env",
  "VersionLabel": "v2",
  "Status": "Updating",
  "EnvironmentId": "e-szqipays4h",
  "EndpointURL": "awseb-e-i-AWSEBLoa-1RDLX6TC9VUA0-0123456789.us-
west-2.elb.amazonaws.com",
  "SolutionStackName": "64bit Amazon Linux running Tomcat 7",
  "CNAME": "my-env.elasticbeanstalk.com",
  "Health": "Grey",
  "Tier": {
    "Version": " ",
    "Type": "Standard",
    "Name": "WebServer"
  },
  "DateUpdated": "2015-02-03T23:12:29.119Z",
  "DateCreated": "2015-02-03T23:04:54.453Z"
```

```
}
```

Para configurar una variable de entorno

El siguiente comando establece el valor de la variable "PARAM1" en el entorno «my-env» en "«: ParamValue

```
aws elasticbeanstalk update-environment --environment-name my-env --option-  
settings Namespace=aws:elasticbeanstalk:application:environment,OptionName=PARAM1,Value=Para
```

El `option-settings` parámetro ocupa un espacio de nombres además del nombre y el valor de la variable. Elastic Beanstalk admite varios espacios de nombres para las opciones, además de las variables de entorno.

Para configurar los ajustes de las opciones desde un archivo

El siguiente comando configura varias opciones en el espacio de `aws:elb:loadbalancer` nombres de un archivo:

```
aws elasticbeanstalk update-environment --environment-name my-env --option-  
settings file://options.json
```

`options.json` es un JSON objeto que define varios ajustes:

```
[  
  {  
    "Namespace": "aws:elb:healthcheck",  
    "OptionName": "Interval",  
    "Value": "15"  
  },  
  {  
    "Namespace": "aws:elb:healthcheck",  
    "OptionName": "Timeout",  
    "Value": "8"  
  },  
  {  
    "Namespace": "aws:elb:healthcheck",  
    "OptionName": "HealthyThreshold",  
    "Value": "2"  
  },  
  {  
    "Namespace": "aws:elb:healthcheck",
```

```
    "OptionName": "UnhealthyThreshold",
    "Value": "3"
  }
]
```

Salida:

```
{
  "ApplicationName": "my-app",
  "EnvironmentName": "my-env",
  "VersionLabel": "7f58-stage-150812_025409",
  "Status": "Updating",
  "EnvironmentId": "e-wtp2rqpsej",
  "EndpointURL": "awseb-e-w-AWSEBLoa-14XB83101Q4L-104QXY80921.sa-
east-1.elb.amazonaws.com",
  "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java
8",
  "CNAME": "my-env.elasticbeanstalk.com",
  "Health": "Grey",
  "AbortableOperationInProgress": true,
  "Tier": {
    "Version": " ",
    "Type": "Standard",
    "Name": "WebServer"
  },
  "DateUpdated": "2015-08-12T18:15:23.804Z",
  "DateCreated": "2015-08-07T20:48:49.599Z"
}
```

Para obtener más información sobre los espacios de nombres y las opciones compatibles, consulte los valores de las opciones en la Guía para desarrolladores de Elastic AWS Beanstalk.

- Para obtener API más información, consulte la Referencia de comandos.

[UpdateEnvironment](#) AWS CLI

## **validate-configuration-settings**

En el siguiente ejemplo de código, se muestra cómo usar `validate-configuration-settings`.

AWS CLI

Para validar los ajustes de configuración

El siguiente comando valida un documento de configuración de métricas CloudWatch personalizado:

```
aws elasticbeanstalk validate-configuration-settings --application-name my-app --
environment-name my-env --option-settings file://options.json
```

`options.json` es un JSON documento que incluye una o más opciones de configuración para validar:

```
[
  {
    "Namespace": "aws:elasticbeanstalk:healthreporting:system",
    "OptionName": "ConfigDocument",
    "Value": "{\"CloudWatchMetrics\": {\"Environment\":
    {\"ApplicationLatencyP99.9\": null,\"InstancesSevere\": 60,
    \"ApplicationLatencyP90\": 60,\"ApplicationLatencyP99\": null,
    \"ApplicationLatencyP95\": 60,\"InstancesUnknown\": 60,\"ApplicationLatencyP85\":
    60,\"InstancesInfo\": null,\"ApplicationRequests2xx\": null,\"InstancesDegraded
    \": null,\"InstancesWarning\": 60,\"ApplicationLatencyP50\": 60,
    \"ApplicationRequestsTotal\": null,\"InstancesNoData\": null,\"InstancesPending
    \": 60,\"ApplicationLatencyP10\": null,\"ApplicationRequests5xx\": null,
    \"ApplicationLatencyP75\": null,\"InstancesOk\": 60,\"ApplicationRequests3xx\":
    null,\"ApplicationRequests4xx\": null},\"Instance\": {\"ApplicationLatencyP99.9\":
    null,\"ApplicationLatencyP90\": 60,\"ApplicationLatencyP99\": null,
    \"ApplicationLatencyP95\": null,\"ApplicationLatencyP85\": null,\"CPUUser\": 60,
    \"ApplicationRequests2xx\": null,\"CPUIdle\": null,\"ApplicationLatencyP50\":
    null,\"ApplicationRequestsTotal\": 60,\"RootFilesystemUtil\": null,
    \"LoadAverage1min\": null,\"CPUIrq\": null,\"CPUNice\": 60,\"CPUiowait\": 60,
    \"ApplicationLatencyP10\": null,\"LoadAverage5min\": null,\"ApplicationRequests5xx
    \": null,\"ApplicationLatencyP75\": 60,\"CPUSystem\": 60,\"ApplicationRequests3xx\":
    60,\"ApplicationRequests4xx\": null,\"InstanceHealth\": null,\"CPUsoftirq\": 60}},
    \"Version\": 1}"
  }
]
```

Si las opciones que especifique son válidas para el entorno especificado, Elastic Beanstalk devuelve una matriz Messages vacía:

```
{
  "Messages": []
}
```

Si la validación falla, la respuesta incluirá información sobre el error:

```
{
  "Messages": [
    {
      "OptionName": "ConfigDocumet",
      "Message": "Invalid option specification (Namespace:
'aws:elasticbeanstalk:healthreporting:system', OptionName: 'ConfigDocumet'):
Unknown configuration setting.",
      "Namespace": "aws:elasticbeanstalk:healthreporting:system",
      "Severity": "error"
    }
  ]
}
```

Para obtener más información sobre los espacios de nombres y las opciones compatibles, consulte los valores de las opciones en la Guía para desarrolladores de Elastic AWS Beanstalk.

- Para obtener API más información, consulte la Referencia de comandos.

[ValidateConfigurationSettings](#) AWS CLI

## Ejemplos de Elastic Load Balancing: versión 1 utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Elastic Load Balancing, versión 1.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **add-tags**

En el siguiente ejemplo de código, se muestra cómo usar add-tags.

## AWS CLI

Para añadir una etiqueta a un balanceador de cargas

En este ejemplo, se agregan etiquetas al balanceador de cargas especificado.

Comando:

```
aws elb add-tags --load-balancer-name my-load-balancer --  
tags "Key=project,Value=Lima" "Key=department,Value=digital-media"
```

- Para API obtener más información, consulta [AddTags](#) la Referencia de AWS CLI comandos.

## apply-security-groups-to-load-balancer

En el siguiente ejemplo de código, se muestra cómo usar `apply-security-groups-to-load-balancer`.

## AWS CLI

Para asociar un grupo de seguridad a un balanceador de cargas en un VPC

En este ejemplo, se asocia un grupo de seguridad al balanceador de cargas especificado en un VPC

Comando:

```
aws elb apply-security-groups-to-load-balancer --load-balancer-name my-load-balancer  
--security-groups sg-fc448899
```

Salida:

```
{  
  "SecurityGroups": [  
    "sg-fc448899"  
  ]  
}
```

- Para API obtener más información, consulte [ApplySecurityGroupsToLoadBalancer](#) la Referencia de AWS CLI comandos.

## attach-load-balancer-to-subnets

En el siguiente ejemplo de código, se muestra cómo usar `attach-load-balancer-to-subnets`.

### AWS CLI

Para adjuntar subredes a un balanceador de cargas

En este ejemplo, se agrega la subred especificada al conjunto de subredes configuradas para el balanceador de cargas especificado.

Comando:

```
aws elb attach-load-balancer-to-subnets --load-balancer-name my-load-balancer --  
subnets subnet-0ecac448
```

Salida:

```
{  
  "Subnets": [  
    "subnet-15aaab61",  
    "subnet-0ecac448"  
  ]  
}
```

- Para API obtener más información, consulta la Referencia de [AttachLoadBalancerToSubnets](#) comandos AWS CLI .

## configure-health-check

En el siguiente ejemplo de código, se muestra cómo usar `configure-health-check`.

### AWS CLI

Para especificar la configuración de comprobación de estado de las instancias de backend EC2

En este ejemplo, se especifican los ajustes de comprobación de estado que se utilizan para evaluar el estado de las instancias de backend EC2.

Comando:

```
aws elb configure-health-check --load-balancer-name my-load-balancer --health-check Target=HTTP:80/png,Interval=30,UnhealthyThreshold=2,HealthyThreshold=2,Timeout=3
```

Salida:

```
{
  "HealthCheck": {
    "HealthyThreshold": 2,
    "Interval": 30,
    "Target": "HTTP:80/png",
    "Timeout": 3,
    "UnhealthyThreshold": 2
  }
}
```

- Para API obtener más información, consulta [ConfigureHealthCheck](#) la Referencia de AWS CLI comandos.

## create-app-cookie-stickiness-policy

En el siguiente ejemplo de código, se muestra cómo usar `create-app-cookie-stickiness-policy`.

AWS CLI

Para generar una política de adherencia para tu HTTPS balanceador de cargas

En este ejemplo, se genera una política de adherencia que sigue la duración de las sesiones fijas de la cookie generada por la aplicación.

Comando:

```
aws elb create-app-cookie-stickiness-policy --load-balancer-name my-load-balancer --policy-name my-app-cookie-policy --cookie-name my-app-cookie
```

- Para obtener API más información, consulte la Referencia de comandos. [CreateAppCookieStickinessPolicy](#) AWS CLI



## create-lb-cookie-stickness-policy

En el siguiente ejemplo de código, se muestra cómo usar `create-lb-cookie-stickness-policy`.

### AWS CLI

Para generar una política de adherencia basada en la duración para tu balanceador de cargas HTTPS

En este ejemplo, se genera una política de rigidez con una duración de sesión fija controlada en función del período de caducidad especificado.

Comando:

```
aws elb create-lb-cookie-stickness-policy --load-balancer-name my-load-balancer --policy-name my-duration-cookie-policy --cookie-expiration-period 60
```

- Para API obtener más información, consulte la Referencia de [CreateLbCookieStickinessPolicy](#) comandos AWS CLI .

## create-load-balancer-listeners

En el siguiente ejemplo de código, se muestra cómo usar `create-load-balancer-listeners`.

### AWS CLI

Para crear HTTP oyentes para un balanceador de cargas

En este ejemplo, se crea un agente de escucha para el balanceador de cargas en el puerto 80 mediante el protocolo. HTTP

Comando:

```
aws elb create-load-balancer-listeners --load-balancer-name my-load-balancer --listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80"
```

Para crear HTTPS oyentes para un balanceador de cargas

En este ejemplo, se crea un agente de escucha para el balanceador de cargas en el puerto 443 mediante el protocolo. HTTPS

Comando:

```
aws elb create-load-balancer-listeners --load-balancer-name my-load-balancer --  
listeners "Protocol=HTTPS,LoadBalancerPort=443,InstanceProtocol=HTTP,InstancePort=80"
```

- Para API obtener más información, consulta la Referencia [CreateLoadBalancerListeners](#) de AWS CLI comandos.

## create-load-balancer-policy

En el siguiente ejemplo de código, se muestra cómo usar `create-load-balancer-policy`.

### AWS CLI

Para crear una política que habilite el protocolo proxy en un balanceador de cargas

En este ejemplo, se crea una política que habilita el protocolo proxy en el balanceador de cargas especificado.

Comando:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-balancer --policy-  
name my-ProxyProtocol-policy --policy-type-name ProxyProtocolPolicyType --policy-  
attributes AttributeName=ProxyProtocol,AttributeValue=true
```

Para crear una política de SSL negociación mediante la política de seguridad recomendada

En este ejemplo, se crea una política de SSL negociación para el equilibrador de HTTPS cargas especificado mediante la política de seguridad recomendada.

Comando:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-  
balancer --policy-name my-SSLNegotiation-policy --policy-type-  
name SSLNegotiationPolicyType --policy-attributes AttributeName=Reference-Security-  
Policy,AttributeValue=ELBSecurityPolicy-2015-03
```

Para crear una política de SSL negociación mediante una política de seguridad personalizada

En este ejemplo, se crea una política de SSL negociación para HTTPS el equilibrador de cargas mediante una política de seguridad personalizada que habilita los protocolos y los cifrados.

Comando:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-balancer --policy-name my-SSLNegotiation-policy --policy-type-name SSLNegotiationPolicyType --policy-attributes AttributeName=Protocol-SSLv3,AttributeValue=true AttributeName=Protocol-TLSv1.1,AttributeValue=true AttributeName=DHE-RSA-AES256-SHA256,AttributeValue=true AttributeName=Server-Defined-Cipher-Order,AttributeValue=true
```

Para crear una política de clave pública

En este ejemplo, se crea una política de clave pública.

Comando:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-balancer --policy-name my-PublicKey-policy --policy-type-name PublicKeyPolicyType --policy-attributes AttributeName=PublicKey,AttributeValue=MIIBIjANBgkqhkiG9w0BAQEFAAAOCAQ8AMIIBCgKCAQ+dS74kj//c6x7R0tusUaeQCTgIUkayttRDWchuqo1pHC1u+n5xxXnBBE2ejbb2WRsKIQ5rXEeixsjFpFsojpSQKkzhVGI6mJVZBJDVKSHmswnwLBdofLhzvllpovBPTHe+o4haAWvDBALJU0pkSI1FecPHcs2hwx14zHoXy1e2k36A64nXW43wtfx5qcVSIxtCE0jnYRg7RPvybaGfQ+v6Iaxb/+7J5kEvZhTFQId+bSiJImF1FSUT1W1xwzBZPubcUkkXDj45vC2s3Z8E+Lk7a3uZhvsQHLZnrfuWjBWGWvZ/MhZYgEXAMPLE
```

Para crear una política de autenticación del servidor back-end

En este ejemplo, se crea una política de autenticación del servidor backend que permite la autenticación en la instancia de backend mediante una política de clave pública.

Comando:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-balancer --policy-name my-authentication-policy --policy-type-name BackendServerAuthenticationPolicyType --policy-attributes AttributeName=PublicKeyPolicyName,AttributeValue=my-PublicKey-policy
```

- Para API obtener más información, consulta la Referencia [CreateLoadBalancerPolicy](#) de AWS CLI comandos.

## create-load-balancer

En el siguiente ejemplo de código, se muestra cómo usar `create-load-balancer`.

## AWS CLI

Para crear un HTTP balanceador de cargas

En este ejemplo, se crea un equilibrador de cargas con un agente de HTTP escucha en un VPC

Comando:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --  
listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80"  
--subnets subnet-15aaab61 --security-groups sg-a61988c3
```

Salida:

```
{  
  "DNSName": "my-load-balancer-1234567890.us-west-2.elb.amazonaws.com"  
}
```

En este ejemplo, se crea un balanceador de cargas con un agente de HTTP escucha en -Classic.  
EC2

Comando:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --  
listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80"  
--availability-zones us-west-2a us-west-2b
```

Salida:

```
{  
  "DNSName": "my-load-balancer-123456789.us-west-2.elb.amazonaws.com"  
}
```

Para crear un balanceador de cargas HTTPS

En este ejemplo, se crea un equilibrador de cargas con un agente de HTTPS escucha en un VPC

Comando:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --  
listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80" "Protocol=HTTPS,LoadBalancerPort=443,InstanceProtocol=HTTPS,InstancePort=443,certificate/my-server-cert" --subnets subnet-15aaab61 --security-groups sg-a61988c3
```

Salida:

```
{
  "DNSName": "my-load-balancer-1234567890.us-west-2.elb.amazonaws.com"
}
```

En este ejemplo, se crea un balanceador de cargas con un agente de HTTPS escucha en - Classic. EC2

Comando:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --
listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80" "Protoco
certificate/my-server-cert" --availability-zones us-west-2a us-west-2b
```

Salida:

```
{
  "DNSName": "my-load-balancer-123456789.us-west-2.elb.amazonaws.com"
}
```

Para crear un balanceador de cargas interno

En este ejemplo, se crea un balanceador de cargas interno con un HTTP oyente en un. VPC

Comando:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --
listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80"
--scheme internal --subnets subnet-a85db0df --security-groups sg-a61988c3
```

Salida:

```
{
  "DNSName": "internal-my-load-balancer-123456789.us-west-2.elb.amazonaws.com"
}
```

- Para API obtener más información, consulte la Referencia [CreateLoadBalancer](#) de AWS CLI comandos.

## delete-load-balancer-listeners

En el siguiente ejemplo de código, se muestra cómo usar `delete-load-balancer-listeners`.

### AWS CLI

Para eliminar un agente de escucha de tu balanceador de cargas

En este ejemplo, se elimina el agente de escucha del puerto especificado del balanceador de cargas especificado.

Comando:

```
aws elb delete-load-balancer-listeners --load-balancer-name my-load-balancer --load-balancer-ports 80
```

- Para API obtener más información, consulta la Referencia de [DeleteLoadBalancerListeners](#) comandos AWS CLI .

## delete-load-balancer-policy

En el siguiente ejemplo de código, se muestra cómo usar `delete-load-balancer-policy`.

### AWS CLI

Para eliminar una política de tu balanceador de cargas

En este ejemplo, se elimina la política especificada del equilibrador de cargas especificado. La política no debe estar habilitada en ningún agente de escucha.

Comando:

```
aws elb delete-load-balancer-policy --load-balancer-name my-load-balancer --policy-name my-duration-cookie-policy
```

- Para API obtener más información, consulte [DeleteLoadBalancerPolicy](#) la Referencia de AWS CLI comandos.

## delete-load-balancer

En el siguiente ejemplo de código, se muestra cómo usar `delete-load-balancer`.

## AWS CLI

### Eliminación de un equilibrador de carga

En este ejemplo, se elimina el balanceador de cargas especificado.

Comando:

```
aws elb delete-load-balancer --load-balancer-name my-load-balancer
```

- Para API obtener más información, consulte la Referencia [DeleteLoadBalancer](#) de AWS CLI comandos.

## deregister-instances-from-load-balancer

En el siguiente ejemplo de código, se muestra cómo usar `deregister-instances-from-load-balancer`.

## AWS CLI

Para anular el registro de instancias de un balanceador de cargas

En este ejemplo, se anula el registro de la instancia especificada en el balanceador de cargas especificado.

Comando:

```
aws elb deregister-instances-from-load-balancer --load-balancer-name my-load-balancer --instances i-d6f6fae3
```

Salida:

```
{
  "Instances": [
    {
      "InstanceId": "i-207d9717"
    },
    {
      "InstanceId": "i-afefb49b"
    }
  ]
}
```

- Para API obtener más información, consulta la Referencia de [DeregisterInstancesFromLoadBalancer](#) comandos AWS CLI .

## describe-account-limits

En el siguiente ejemplo de código, se muestra cómo usar `describe-account-limits`.

### AWS CLI

Para describir tus límites de Classic Load Balancer

En el siguiente `describe-account-limits` ejemplo, se muestran detalles sobre los límites de Classic Load Balancer de tu AWS cuenta.

```
aws elb describe-account-limits
```

Salida:

```
{
  "Limits": [
    {
      "Name": "classic-load-balancers",
      "Max": "20"
    },
    {
      "Name": "classic-listeners",
      "Max": "100"
    },
    {
      "Name": "classic-registered-instances",
      "Max": "1000"
    }
  ]
}
```

- Para API obtener más información, consulta [DescribeAccountLimits](#) la Referencia de AWS CLI comandos.

## describe-instance-health

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-health`.



## AWS CLI

Para describir el estado de las instancias de un balanceador de cargas

En este ejemplo, se describe el estado de las instancias del balanceador de cargas especificado.

Comando:

```
aws elb describe-instance-health --load-balancer-name my-load-balancer
```

Salida:

```
{
  "InstanceStates": [
    {
      "InstanceId": "i-207d9717",
      "ReasonCode": "N/A",
      "State": "InService",
      "Description": "N/A"
    },
    {
      "InstanceId": "i-afefb49b",
      "ReasonCode": "N/A",
      "State": "InService",
      "Description": "N/A"
    }
  ]
}
```

Para describir el estado de una instancia de un balanceador de cargas

En este ejemplo, se describe el estado de la instancia especificada para el balanceador de cargas especificado.

Comando:

```
aws elb describe-instance-health --load-balancer-name my-load-balancer --
instances i-7299c809
```

A continuación, se muestra un ejemplo de respuesta para una instancia que se está registrando.

Salida:

```
{
  "InstanceStates": [
    {
      "InstanceId": "i-7299c809",
      "ReasonCode": "ELB",
      "State": "OutOfService",
      "Description": "Instance registration is still in progress."
    }
  ]
}
```

A continuación, se muestra un ejemplo de respuesta para una instancia en mal estado.

Salida:

```
{
  "InstanceStates": [
    {
      "InstanceId": "i-7299c809",
      "ReasonCode": "Instance",
      "State": "OutOfService",
      "Description": "Instance has failed at least the UnhealthyThreshold number
of health checks consecutively."
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeInstanceHealth](#) la Referencia de AWS CLI comandos.

## **describe-load-balancer-attributes**

En el siguiente ejemplo de código, se muestra cómo usar `describe-load-balancer-attributes`.

AWS CLI

Para describir los atributos de un balanceador de cargas

En este ejemplo, se describen los atributos del balanceador de cargas especificado.

Comando:

```
aws elb describe-load-balancer-attributes --load-balancer-name my-load-balancer
```

Salida:

```
{
  "LoadBalancerAttributes": {
    "ConnectionDraining": {
      "Enabled": false,
      "Timeout": 300
    },
    "CrossZoneLoadBalancing": {
      "Enabled": true
    },
    "ConnectionSettings": {
      "IdleTimeout": 30
    },
    "AccessLog": {
      "Enabled": false
    }
  }
}
```

- Para API obtener más información, consulta [DescribeLoadBalancerAttributes](#) la Referencia de AWS CLI comandos.

## **describe-load-balancer-policies**

En el siguiente ejemplo de código, se muestra cómo usar `describe-load-balancer-policies`.

AWS CLI

Para describir todas las políticas asociadas a un balanceador de cargas

En este ejemplo, se describen todas las políticas asociadas al balanceador de cargas especificado.

Comando:

```
aws elb describe-load-balancer-policies --load-balancer-name my-load-balancer
```

Salida:

```
{
  "PolicyDescriptions": [
    {
      "PolicyAttributeDescriptions": [
        {
          "AttributeName": "ProxyProtocol",
          "AttributeValue": "true"
        }
      ],
      "PolicyName": "my-ProxyProtocol-policy",
      "PolicyTypeName": "ProxyProtocolPolicyType"
    },
    {
      "PolicyAttributeDescriptions": [
        {
          "AttributeName": "CookieName",
          "AttributeValue": "my-app-cookie"
        }
      ],
      "PolicyName": "my-app-cookie-policy",
      "PolicyTypeName": "AppCookieStickinessPolicyType"
    },
    {
      "PolicyAttributeDescriptions": [
        {
          "AttributeName": "CookieExpirationPeriod",
          "AttributeValue": "60"
        }
      ],
      "PolicyName": "my-duration-cookie-policy",
      "PolicyTypeName": "LBCookieStickinessPolicyType"
    },
    .
    .
    .
  ]
}
```

Para describir una política específica asociada a un balanceador de cargas

En este ejemplo, se describe la política especificada asociada al balanceador de cargas especificado.

Comando:

```
aws elb describe-load-balancer-policies --load-balancer-name my-load-balancer --  
policy-name my-authentication-policy
```

Salida:

```
{  
  "PolicyDescriptions": [  
    {  
      "PolicyAttributeDescriptions": [  
        {  
          "AttributeName": "PublicKeyPolicyName",  
          "AttributeValue": "my-PublicKey-policy"  
        }  
      ],  
      "PolicyName": "my-authentication-policy",  
      "PolicyTypeName": "BackendServerAuthenticationPolicyType"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [DescribeLoadBalancerPolicies](#) la Referencia de AWS CLI comandos.

## describe-load-balancer-policy-types

En el siguiente ejemplo de código, se muestra cómo usar `describe-load-balancer-policy-types`.

AWS CLI

Para describir los tipos de políticas de balanceador de carga definidos por Elastic Load Balancing

En este ejemplo, se describen los tipos de políticas de balanceador de carga que puede usar para crear configuraciones de políticas para su balanceador de carga.

Comando:

```
aws elb describe-load-balancer-policy-types
```

**Salida:**

```
{
  "PolicyTypeDescriptions": [
    {
      "PolicyAttributeTypeDescriptions": [
        {
          "Cardinality": "ONE",
          "AttributeName": "ProxyProtocol",
          "AttributeType": "Boolean"
        }
      ],
      "PolicyTypeName": "ProxyProtocolPolicyType",
      "Description": "Policy that controls whether to include the IP address and
port of the originating request for TCP messages. This policy operates on TCP/SSL
listeners only"
    },
    {
      "PolicyAttributeTypeDescriptions": [
        {
          "Cardinality": "ONE",
          "AttributeName": "PublicKey",
          "AttributeType": "String"
        }
      ],
      "PolicyTypeName": "PublicKeyPolicyType",
      "Description": "Policy containing a list of public keys to
accept when authenticating the back-end server(s). This policy cannot be
applied directly to back-end servers or listeners but must be part of a
BackendServerAuthenticationPolicyType."
    },
    {
      "PolicyAttributeTypeDescriptions": [
        {
          "Cardinality": "ONE",
          "AttributeName": "CookieName",
          "AttributeType": "String"
        }
      ],
      "PolicyTypeName": "AppCookieStickinessPolicyType",
      "Description": "Stickiness policy with session lifetimes controlled by the
lifetime of the application-generated cookie. This policy can be associated only
with HTTP/HTTPS listeners."
    },
  ],
}
```

```

{
  "PolicyAttributeTypeDescriptions": [
    {
      "Cardinality": "ZERO_OR_ONE",
      "AttributeName": "CookieExpirationPeriod",
      "AttributeType": "Long"
    }
  ],
  "PolicyTypeName": "LBCookieStickinessPolicyType",
  "Description": "Stickiness policy with session lifetimes controlled by
the browser (user-agent) or a specified expiration period. This policy can be
associated only with HTTP/HTTPS listeners."
},
{
  "PolicyAttributeTypeDescriptions": [
    .
    .
    .
  ],
  "PolicyTypeName": "SSLNegotiationPolicyType",
  "Description": "Listener policy that defines the ciphers and protocols
that will be accepted by the load balancer. This policy can be associated only with
HTTPS/SSL listeners."
},
{
  "PolicyAttributeTypeDescriptions": [
    {
      "Cardinality": "ONE_OR_MORE",
      "AttributeName": "PublicKeyPolicyName",
      "AttributeType": "PolicyName"
    }
  ],
  "PolicyTypeName": "BackendServerAuthenticationPolicyType",
  "Description": "Policy that controls authentication to back-end server(s)
and contains one or more policies, such as an instance of a PublicKeyPolicyType.
This policy can be associated only with back-end servers that are using HTTPS/SSL."
}
]
}

```

- Para API obtener más información, consulta la Referencia [DescribeLoadBalancerPolicyTypes](#) de AWS CLI comandos.

## describe-load-balancers

En el siguiente ejemplo de código, se muestra cómo usar `describe-load-balancers`.

### AWS CLI

Para describir sus balanceadores de carga

En este ejemplo, se describen todos los equilibradores de carga.

Comando:

```
aws elb describe-load-balancers
```

Para describir uno de tus balanceadores de carga

En este ejemplo se describe el equilibrador de carga especificado.

Comando:

```
aws elb describe-load-balancers --load-balancer-name my-load-balancer
```

El siguiente ejemplo de respuesta es para un equilibrador de HTTPS carga en un VPC

Salida:

```
{
  "LoadBalancerDescriptions": [
    {
      "Subnets": [
        "subnet-15aaab61"
      ],
      "CanonicalHostedZoneNameID": "Z3DZXE0EXAMPLE",
      "CanonicalHostedZoneName": "my-load-balancer-1234567890.us-west-2.elb.amazonaws.com",
      "ListenerDescriptions": [
        {
          "Listener": {
            "InstancePort": 80,
            "LoadBalancerPort": 80,
            "Protocol": "HTTP",
            "InstanceProtocol": "HTTP"
          },
          "PolicyNames": []
        }
      ]
    }
  ]
}
```



```
    },
    {
      "Listener": {
        "InstancePort": 443,
        "SSLCertificateId": "arn:aws:iam::123456789012:server-certificate/
my-server-cert",
        "LoadBalancerPort": 443,
        "Protocol": "HTTPS",
        "InstanceProtocol": "HTTPS"
      },
      "PolicyNames": [
        "ELBSecurityPolicy-2015-03"
      ]
    }
  ],
  "HealthCheck": {
    "HealthyThreshold": 2,
    "Interval": 30,
    "Target": "HTTP:80/png",
    "Timeout": 3,
    "UnhealthyThreshold": 2
  },
  "VPCId": "vpc-a01106c2",
  "BackendServerDescriptions": [
    {
      "InstancePort": 80,
      "PolicyNames": [
        "my-ProxyProtocol-policy"
      ]
    }
  ],
  "Instances": [
    {
      "InstanceId": "i-207d9717"
    },
    {
      "InstanceId": "i-afefb49b"
    }
  ],
  "DNSName": "my-load-balancer-1234567890.us-west-2.elb.amazonaws.com",
  "SecurityGroups": [
    "sg-a61988c3"
  ],
  "Policies": {
```

```

    "LBCookieStickinessPolicies": [
      {
        "PolicyName": "my-duration-cookie-policy",
        "CookieExpirationPeriod": 60
      }
    ],
    "AppCookieStickinessPolicies": [],
    "OtherPolicies": [
      "my-PublicKey-policy",
      "my-authentication-policy",
      "my-SSLNegotiation-policy",
      "my-ProxyProtocol-policy",
      "ELBSecurityPolicy-2015-03"
    ]
  },
  "LoadBalancerName": "my-load-balancer",
  "CreatedTime": "2015-03-19T03:24:02.650Z",
  "AvailabilityZones": [
    "us-west-2a"
  ],
  "Scheme": "internet-facing",
  "SourceSecurityGroup": {
    "OwnerAlias": "123456789012",
    "GroupName": "my-elb-sg"
  }
}
]
}

```

- Para API obtener más información, consulte [DescribeLoadBalancers](#) la Referencia de AWS CLI comandos.

## describe-tags

En el siguiente ejemplo de código, se muestra cómo usar `describe-tags`.

### AWS CLI

Para describir las etiquetas asignadas a un balanceador de cargas

En este ejemplo, se describen las etiquetas asignadas al balanceador de cargas especificado.

Comando:

```
aws elb describe-tags --load-balancer-name my-load-balancer
```

Salida:

```
{
  "TagDescriptions": [
    {
      "Tags": [
        {
          "Value": "lima",
          "Key": "project"
        },
        {
          "Value": "digital-media",
          "Key": "department"
        }
      ],
      "LoadBalancerName": "my-load-balancer"
    }
  ]
}
```

- Para API obtener más información, consulta [DescribeTags](#) la Referencia de AWS CLI comandos.

## detach-load-balancer-from-subnets

En el siguiente ejemplo de código, se muestra cómo usar `detach-load-balancer-from-subnets`.

AWS CLI

Para separar los balanceadores de carga de las subredes

En este ejemplo, se separa el balanceador de cargas especificado de la subred especificada.

Comando:

```
aws elb detach-load-balancer-from-subnets --load-balancer-name my-load-balancer --
subnets subnet-0ecac448
```

Salida:

```
{
  "Subnets": [
    "subnet-15aaab61"
  ]
}
```

- Para API obtener más información, consulta la Referencia de [DetachLoadBalancerFromSubnets](#) comandos AWS CLI .

## **disable-availability-zones-for-load-balancer**

En el siguiente ejemplo de código, se muestra cómo usar `disable-availability-zones-for-load-balancer`.

AWS CLI

Para deshabilitar las zonas de disponibilidad de un balanceador de cargas

En este ejemplo, se elimina la zona de disponibilidad especificada del conjunto de zonas de disponibilidad del equilibrador de cargas especificado.

Comando:

```
aws elb disable-availability-zones-for-load-balancer --load-balancer-name my-Load-balancer --availability-zones us-west-2a
```

Salida:

```
{
  "AvailabilityZones": [
    "us-west-2b"
  ]
}
```

- Para API obtener más información, consulte [DisableAvailabilityZonesForLoadBalancer](#) la Referencia de AWS CLI comandos.

## **enable-availability-zones-for-load-balancer**

En el siguiente ejemplo de código, se muestra cómo usar `enable-availability-zones-for-load-balancer`.

### AWS CLI

Para habilitar las zonas de disponibilidad para un balanceador de cargas

En este ejemplo, se agrega la zona de disponibilidad especificada al balanceador de cargas especificado.

Comando:

```
aws elb enable-availability-zones-for-load-balancer --load-balancer-name my-Load-balancer --availability-zones us-west-2b
```

Salida:

```
{
  "AvailabilityZones": [
    "us-west-2a",
    "us-west-2b"
  ]
}
```

- Para API obtener más información, consulte [EnableAvailabilityZonesForLoadBalancer](#) la Referencia de AWS CLI comandos.

## **modify-load-balancer-attributes**

En el siguiente ejemplo de código, se muestra cómo usar `modify-load-balancer-attributes`.

### AWS CLI

Para modificar los atributos de un balanceador de cargas

En este ejemplo, se modifica el `CrossZoneLoadBalancing` atributo del equilibrador de cargas especificado.

Comando:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-load-balancer --  
load-balancer-attributes "{\"CrossZoneLoadBalancing\":{\"Enabled\":true}}"
```

Salida:

```
{  
  "LoadBalancerAttributes": {  
    "CrossZoneLoadBalancing": {  
      "Enabled": true  
    }  
  },  
  "LoadBalancerName": "my-load-balancer"  
}
```

En este ejemplo, se modifica el `ConnectionDraining` atributo del equilibrador de cargas especificado.

Comando:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-load-balancer  
--load-balancer-attributes "{\"ConnectionDraining\":{\"Enabled\":true,\"Timeout  
\":300}}"
```

Salida:

```
{  
  "LoadBalancerAttributes": {  
    "ConnectionDraining": {  
      "Enabled": true,  
      "Timeout": 300  
    }  
  },  
  "LoadBalancerName": "my-load-balancer"  
}
```

- Para API obtener más información, consulte la Referencia [ModifyLoadBalancerAttributes](#) de AWS CLI comandos.

## register-instances-with-load-balancer

En el siguiente ejemplo de código, se muestra cómo usar `register-instances-with-load-balancer`.

### AWS CLI

Para registrar instancias con un balanceador de carga

En este ejemplo, se registra la instancia especificada con el balanceador de cargas especificado.

Comando:

```
aws elb register-instances-with-load-balancer --load-balancer-name my-load-balancer
--instances i-d6f6fae3
```

Salida:

```
{
  "Instances": [
    {
      "InstanceId": "i-d6f6fae3"
    },
    {
      "InstanceId": "i-207d9717"
    },
    {
      "InstanceId": "i-afefb49b"
    }
  ]
}
```

- Para API obtener más información, consulta [RegisterInstancesWithLoadBalancer](#) la Referencia de AWS CLI comandos.

## remove-tags

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags`.

### AWS CLI

Para eliminar etiquetas de un balanceador de cargas

En este ejemplo, se elimina una etiqueta del balanceador de cargas especificado.

Comando:

```
aws elb remove-tags --load-balancer-name my-load-balancer --tags project
```

- Para API obtener más información, consulta [RemoveTags](#) la Referencia de AWS CLI comandos.

## set-load-balancer-listener-ssl-certificate

En el siguiente ejemplo de código, se muestra cómo usar `set-load-balancer-listener-ssl-certificate`.

AWS CLI

Para actualizar el SSL certificado de un balanceador HTTPS de cargas

Este ejemplo reemplaza el SSL certificado existente para el balanceador de HTTPS cargas especificado.

Comando:

```
aws elb set-load-balancer-listener-ssl-certificate --load-balancer-name my-load-balancer --load-balancer-port 443 --ssl-certificate-id arn:aws:iam::123456789012:server-certificate/new-server-cert
```

- Para API obtener más información, consulte [SetLoadBalancerListenerSslCertificate](#) la Referencia de AWS CLI comandos.

## set-load-balancer-policies-for-backend-server

En el siguiente ejemplo de código, se muestra cómo usar `set-load-balancer-policies-for-backend-server`.

AWS CLI

Para reemplazar las políticas asociadas a un puerto por una instancia de backend

En este ejemplo, se sustituyen las políticas que están asociadas actualmente al puerto especificado.



Comando:

```
aws elb set-load-balancer-policies-for-backend-server --load-balancer-name my-load-balancer --instance-port 80 --policy-names my-ProxyProtocol-policy
```

Para eliminar todas las políticas que están asociadas actualmente a un puerto en tu instancia de backend

En este ejemplo, se eliminan todas las políticas asociadas al puerto especificado.

Comando:

```
aws elb set-load-balancer-policies-for-backend-server --load-balancer-name my-load-balancer --instance-port 80 --policy-names []
```

Para confirmar que se han eliminado las políticas, utilice el `describe-load-balancer-policies` comando.

- Para API obtener más información, consulte [SetLoadBalancerPoliciesForBackendServer](#) la Referencia de AWS CLI comandos.

## set-load-balancer-policies-of-listener

En el siguiente ejemplo de código, se muestra cómo usar `set-load-balancer-policies-of-listener`.

AWS CLI

Para reemplazar las políticas asociadas a un oyente

En este ejemplo, se sustituyen las políticas que están asociadas actualmente al agente de escucha especificado.

Comando:

```
aws elb set-load-balancer-policies-of-listener --load-balancer-name my-load-balancer --load-balancer-port 443 --policy-names my-SSLNegotiation-policy
```

Para eliminar todas las políticas asociadas a su oyente

En este ejemplo, se eliminan todas las políticas que están asociadas actualmente al agente de escucha especificado.

Comando:

```
aws elb set-load-balancer-policies-of-listener --load-balancer-name my-load-balancer
--load-balancer-port 443 --policy-names []
```

Para confirmar que las políticas se han eliminado del balanceador de cargas, usa el `describe-load-balancer-policies` comando.

- Para API obtener más información, consulte [SetLoadBalancerPoliciesOfListener](#) la Referencia de AWS CLI comandos.

## Ejemplos de Elastic Load Balancing: versión 2 utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Elastic Load Balancing, versión 2.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

Acciones

### **add-listener-certificates**

En el siguiente ejemplo de código, se muestra cómo usar `add-listener-certificates`.

AWS CLI

Para añadir un certificado a un agente de escucha seguro

En este ejemplo, se agrega el certificado especificado al agente de escucha seguro especificado.

Comando:

```
aws elbv2 add-listener-certificates --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 --certificates CertificateArn=arn:aws:acm:us-west-2:123456789012:certificate/5cc54884-f4a3-4072-80be-05b9ba72f705
```

Salida:

```
{
  "Certificates": [
    {
      "CertificateArn": "arn:aws:acm:us-west-2:123456789012:certificate/5cc54884-f4a3-4072-80be-05b9ba72f705",
      "IsDefault": false
    }
  ]
}
```

- Para API obtener más información, consulte [AddListenerCertificates](#) la Referencia de AWS CLI comandos.

## add-tags

En el siguiente ejemplo de código, se muestra cómo usar add-tags.

AWS CLI

Para añadir etiquetas a un balanceador de cargas

En el siguiente add-tags ejemplo, se agregan las department etiquetas project y al balanceador de cargas especificado.

```
aws elbv2 add-tags \
  --resource-arns arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \
  --tags "Key=project, Value=Lima" "Key=department, Value=digital-media"
```

- Para API obtener más información, consulte [AddTags](#) la Referencia de AWS CLI comandos.

## create-listener

En el siguiente ejemplo de código, se muestra cómo usar create-listener.

## AWS CLI

Ejemplo 1: Para crear un HTTP oyente

En el siguiente `create-listener` ejemplo, se crea un agente de HTTP escucha para el Application Load Balancer especificado que reenvía las solicitudes al grupo de destino especificado.

```
aws elbv2 create-listener \  
  --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \  
  --protocol HTTP \  
  --port 80 \  
  --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Para obtener más información, consulte el [tutorial: Creación de un balanceador de carga de aplicaciones mediante la AWS CLI](#) Guía del usuario de balanceadores de carga de aplicaciones.

Ejemplo 2: Para crear un oyente HTTPS

En el siguiente `create-listener` ejemplo, se crea un agente de HTTPS escucha para el Application Load Balancer especificado que reenvía las solicitudes al grupo de destino especificado. Debe especificar un SSL certificado para un listener. HTTPS Puede crear y administrar certificados mediante AWS Certificate Manager (ACM). Como alternativa, puede crear un certificado conSSL/TLS tools, conseguir que una entidad emisora de certificados (CA) firme el certificado y cargarlo en AWS Identity and Access Management (IAM).

```
aws elbv2 create-listener \  
  --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \  
  --protocol HTTPS \  
  --port 443 \  
  --certificates CertificateArn=arn:aws:acm:us-west-2:123456789012:certificate/3dcb0a41-bd72-4774-9ad9-756919c40557 \  
  --ssl-policy ELBSecurityPolicy-2016-08 \  
  --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Para obtener más información, consulte [Añadir un agente de HTTPS escucha](#) en la Guía del usuario de los balanceadores de carga de aplicaciones.

### Ejemplo 3: Para crear un oyente TCP

En el siguiente `create-listener` ejemplo, se crea un agente de TCP escucha para el Network Load Balancer especificado que reenvía las solicitudes al grupo de destino especificado.

```
aws elbv2 create-listener \  
  --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/net/my-network-load-balancer/5d1b75f4f1cee11e \  
  --protocol TCP \  
  --port 80 \  
  --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-tcp-targets/b6bba954d1361c78
```

Para obtener más información, consulte el [tutorial: Creación de un balanceador de carga de red mediante la AWS CLI](#) Guía del usuario para balanceadores de carga de red.

### Ejemplo 4: Para crear un oyente TLS

En el siguiente `create-listener` ejemplo, se crea un agente de TLS escucha para el Network Load Balancer especificado que reenvía las solicitudes al grupo de destino especificado. Debe especificar un SSL certificado para un listener. TLS

```
aws elbv2 create-listener \  
  --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \  
  --protocol TLS \  
  --port 443 \  
  --certificates CertificateArn=arn:aws:acm:us-west-2:123456789012:certificate/3dcb0a41-bd72-4774-9ad9-756919c40557 \  
  --ssl-policy ELBSecurityPolicy-2016-08 \  
  --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Para obtener más información, consulte [TLSlos oyentes de su Network Load Balancer](#) en la Guía del usuario de Network Load Balancers.

### Ejemplo 5: Para crear un oyente UDP

En el siguiente `create-listener` ejemplo, se crea un agente de UDP escucha para el Network Load Balancer especificado que reenvía las solicitudes al grupo de destino especificado.

```
aws elbv2 create-listener \  
  --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/net/my-network-load-balancer/5d1b75f4f1cee11e \  
  --protocol UDP \  
  --port 80 \  
  --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-tcp-targets/b6bba954d1361c78
```

```

--load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/net/my-network-load-balancer/5d1b75f4f1cee11e \
--protocol UDP \
--port 53 \
--default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-tcp-targets/b6bba954d1361c78

```

Para obtener más información, consulte el [tutorial: Creación de un balanceador de carga de red mediante la AWS CLI](#) Guía del usuario para balanceadores de carga de red.

Ejemplo 6: Creación de un oyente para la puerta de enlace y el reenvío especificados

En el siguiente ejemplo de `create-listener` se crea un oyente para el Equilibrador de carga de puerta de enlace especificado que reenvía las solicitudes al grupo de destino especificado.

```

aws elbv2 create-listener \
--load-balancer-arn arn:aws:elasticloadbalancing:us-
east-1:850631746142:loadbalancer/gwy/my-gateway-load-balancer/e0f9b3d5c7f7d3d6 \
--default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
east-1:850631746142:targetgroup/my-glb-targets/007ca469fae3bb1615

```

Salida:

```

{
  "Listeners": [
    {
      "ListenerArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:listener/gwy/my-agw-lb-example2/e0f9b3d5c7f7d3d6/
afc127db15f925de",
      "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:loadbalancer/gwy/my-agw-lb-example2/e0f9b3d5c7f7d3d6",
      "DefaultActions": [
        {
          "Type": "forward",
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:targetgroup/test-tg-agw-2/007ca469fae3bb1615",
          "ForwardConfig": {
            "TargetGroups": [
              {
                "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:targetgroup/test-tg-agw-2/007ca469fae3bb1615"
              }
            ]
          }
        }
      ]
    }
  ]
}

```

```
}
  ]
}
  ]
}
  ]
}
```

Para obtener más información, consulte [Introducción a los balanceadores de carga de puerta de enlace utilizando la Guía del usuario de los AWS CLI balanceadores de carga de puerta de enlace](#).

- Para API obtener más información, consulte la Referencia [CreateListener](#) de AWS CLI comandos.

## create-load-balancer

En el siguiente ejemplo de código, se muestra cómo usar `create-load-balancer`.

### AWS CLI

Ejemplo 1: creación de un equilibrador de carga expuesto a Internet

En el siguiente ejemplo de `create-load-balancer` se crea un equilibrador de carga de aplicación expuesto a Internet y se habilitan las zonas de disponibilidad para las subredes especificadas.

```
aws elbv2 create-load-balancer \
  --name my-load-balancer \
  --subnets subnet-b7d581c0 subnet-8360a9e7
```

Salida:

```
{
  "LoadBalancers": [
    {
      "Type": "application",
      "Scheme": "internet-facing",
      "IpAddressType": "ipv4",
      "VpcId": "vpc-3ac0fb5f",
      "AvailabilityZones": [
```

```

        {
            "ZoneName": "us-west-2a",
            "SubnetId": "subnet-8360a9e7"
        },
        {
            "ZoneName": "us-west-2b",
            "SubnetId": "subnet-b7d581c0"
        }
    ],
    "CreatedTime": "2017-08-25T21:26:12.920Z",
    "CanonicalHostedZoneId": "Z2P70J7EXAMPLE",
    "DNSName": "my-load-balancer-424835706.us-west-2.elb.amazonaws.com",
    "SecurityGroups": [
        "sg-5943793c"
    ],
    "LoadBalancerName": "my-load-balancer",
    "State": {
        "Code": "provisioning"
    },
    "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188"
    }
]
}

```

Para obtener más información, consulte el [tutorial: Creación de un balanceador de carga de aplicaciones mediante la AWS CLI](#) Guía del usuario de balanceadores de carga de aplicaciones.

Ejemplo 2: Creación de un equilibrador de carga interno

En el siguiente ejemplo de `create-load-balancer` se crea un equilibrador de carga de aplicación interno y se habilitan las zonas de disponibilidad para las subredes especificadas.

```

aws elbv2 create-load-balancer \
  --name my-internal-load-balancer \
  --scheme internal \
  --subnets subnet-b7d581c0 subnet-8360a9e7

```

Salida:

```

{
  "LoadBalancers": [

```



```

    {
      "Type": "application",
      "Scheme": "internal",
      "IpAddressType": "ipv4",
      "VpcId": "vpc-3ac0fb5f",
      "AvailabilityZones": [
        {
          "ZoneName": "us-west-2a",
          "SubnetId": "subnet-8360a9e7"
        },
        {
          "ZoneName": "us-west-2b",
          "SubnetId": "subnet-b7d581c0"
        }
      ],
      "CreatedTime": "2016-03-25T21:29:48.850Z",
      "CanonicalHostedZoneId": "Z2P70J7EXAMPLE",
      "DNSName": "internal-my-internal-load-balancer-1529930873.us-
west-2.elb.amazonaws.com",
      "SecurityGroups": [
        "sg-5943793c"
      ],
      "LoadBalancerName": "my-internal-load-balancer",
      "State": {
        "Code": "provisioning"
      },
      "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-internal-load-balancer/5b49b8d4303115c2"
    }
  ]
}

```

Para obtener más información, consulte el [tutorial: Creación de un balanceador de carga de aplicaciones mediante la AWS CLI](#) Guía del usuario de balanceadores de carga de aplicaciones.

### Ejemplo 3: Creación de un equilibrador de carga de red

En el siguiente ejemplo de `create-load-balancer` se crea un equilibrador de carga de red expuesto a Internet y se habilitan las zonas de disponibilidad para la subred especificada. Utiliza un mapeo de subred para asociar la dirección IP elástica especificada con la interfaz de red utilizada por los nodos del equilibrador de carga de la zona de disponibilidad.

```
aws elbv2 create-load-balancer \
```

```
--name my-network-load-balancer \
--type network \
--subnet-mappings SubnetId=subnet-b7d581c0,AllocationId=eipalloc-64d5890a
```

Salida:

```
{
  "LoadBalancers": [
    {
      "Type": "network",
      "Scheme": "internet-facing",
      "IpAddressType": "ipv4",
      "VpcId": "vpc-3ac0fb5f",
      "AvailabilityZones": [
        {
          "LoadBalancerAddresses": [
            {
              "IpAddress": "35.161.207.171",
              "AllocationId": "eipalloc-64d5890a"
            }
          ],
          "ZoneName": "us-west-2b",
          "SubnetId": "subnet-5264e837"
        }
      ],
      "CreatedTime": "2017-10-15T22:41:25.657Z",
      "CanonicalHostedZoneId": "Z2P70J7EXAMPLE",
      "DNSName": "my-network-load-balancer-5d1b75f4f1cee11e.elb.us-
west-2.amazonaws.com",
      "LoadBalancerName": "my-network-load-balancer",
      "State": {
        "Code": "provisioning"
      },
      "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/net/my-network-load-balancer/5d1b75f4f1cee11e"
    }
  ]
}
```

Para obtener más información, consulte el [tutorial: Creación de un balanceador de carga de red mediante la AWS CLI](#) Guía del usuario para balanceadores de carga de red.

Ejemplo 4: Creación de un equilibrador de carga de puerta de enlace

En el siguiente ejemplo de `create-load-balancer` se crea un equilibrador de carga de puerta de enlace y se habilitan las zonas de disponibilidad para las subredes especificadas.

```
aws elbv2 create-load-balancer \  
  --name my-gateway-load-balancer \  
  --type gateway \  
  --subnets subnet-dc83f691 subnet-a62583f9
```

Salida:

```
{  
  "LoadBalancers": [  
    {  
      "Type": "gateway",  
      "VpcId": "vpc-838475fe",  
      "AvailabilityZones": [  
        {  
          "ZoneName": "us-east-1b",  
          "SubnetId": "subnet-a62583f9"  
        },  
        {  
          "ZoneName": "us-east-1a",  
          "SubnetId": "subnet-dc83f691"  
        }  
      ],  
      "CreatedTime": "2021-07-14T19:33:43.324000+00:00",  
      "LoadBalancerName": "my-gateway-load-balancer",  
      "State": {  
        "Code": "provisioning"  
      },  
      "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-  
east-1:850631746142:loadbalancer/gwy/my-gateway-load-balancer/dfbb5a7d32cdee79"  
    }  
  ]  
}
```

Para obtener más información, consulte [Introducción a los balanceadores de carga de puerta de enlace utilizando la Guía del usuario de los AWS CLI balanceadores](#) de carga de puerta de enlace.

- Para API obtener más información, consulte la Referencia [CreateLoadBalancer](#) de AWS CLI comandos.

## create-rule

En el siguiente ejemplo de código, se muestra cómo usar `create-rule`.

### AWS CLI

Ejemplo 1: Para crear una regla mediante una condición de ruta y una acción de reenvío

En el siguiente `create-rule` ejemplo, se crea una regla que reenvía las solicitudes al grupo objetivo especificado si URL contiene el patrón especificado.

```
aws elbv2 create-rule \  
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \  
  --priority 5 \  
  --conditions file://conditions-pattern.json \  
  --actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Contenidos de `conditions-pattern.json`:

```
[  
  {  
    "Field": "path-pattern",  
    "PathPatternConfig": {  
      "Values": ["/images/*"]  
    }  
  }  
]
```

Ejemplo 2: Para crear una regla mediante una condición de host y una respuesta fija

El siguiente `create-rule` ejemplo crea una regla que proporciona una respuesta fija si el nombre de host del encabezado del host coincide con el nombre de host especificado.

```
aws elbv2 create-rule \  
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \  
  --priority 10 \  
  --conditions file://conditions-host.json \  
  --actions file://actions-fixed-response.json
```

## Contenido de conditions-host.json

```
[
  {
    "Field": "host-header",
    "HostHeaderConfig": {
      "Values": ["*.example.com"]
    }
  }
]
```

## Contenido de actions-fixed-response.json

```
[
  {
    "Type": "fixed-response",
    "FixedResponseConfig": {
      "MessageBody": "Hello world",
      "StatusCode": "200",
      "ContentType": "text/plain"
    }
  }
]
```

Ejemplo 3: Para crear una regla mediante una condición de dirección IP de origen, una acción de autenticación y una acción de reenvío

El siguiente `create-rule` ejemplo crea una regla que autentica al usuario si la dirección IP de origen coincide con la dirección IP especificada y reenvía la solicitud al grupo de destino especificado si la autenticación se realiza correctamente.

```
aws elbv2 create-rule \
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
  --priority 20 \
  --conditions file://conditions-source-ip.json \
  --actions file://actions-authenticate.json
```

## Contenido de conditions-source-ip.json

```
[
```

```

    {
      "Field": "source-ip",
      "SourceIpConfig": {
        "Values": ["192.0.2.0/24", "198.51.100.10/32"]
      }
    }
  ]

```

### Contenido de actions-authenticate.json

```

[
  {
    "Type": "authenticate-oidc",
    "AuthenticateOidcConfig": {
      "Issuer": "https://idp-issuer.com",
      "AuthorizationEndpoint": "https://authorization-endpoint.com",
      "TokenEndpoint": "https://token-endpoint.com",
      "UserInfoEndpoint": "https://user-info-endpoint.com",
      "ClientId": "abcdefghijklmnopqrstuvwxy123456789",
      "ClientSecret": "123456789012345678901234567890",
      "SessionCookieName": "my-cookie",
      "SessionTimeout": 3600,
      "Scope": "email",
      "AuthenticationRequestExtraParams": {
        "display": "page",
        "prompt": "login"
      },
      "OnUnauthenticatedRequest": "deny"
    },
    "Order": 1
  },
  {
    "Type": "forward",
    "TargetGroupArn": "arn:aws:elasticloadbalancing:us-east-1:880185128111:targetgroup/cli-test/642a97ecb0e0f26b",
    "Order": 2
  }
]

```

- Para API obtener más información, consulte la Referencia [CreateRule](#) de AWS CLI comandos.

## create-target-group

En el siguiente ejemplo de código, se muestra cómo usar `create-target-group`.

### AWS CLI

Ejemplo 1: Para crear un grupo objetivo para un Application Load Balancer

En el siguiente ejemplo de `create-target-group` se crea un grupo de destino para un Equilibrador de carga de aplicación en el que se registran los destinos por ID de instancia (el tipo de destino es `instance`). Este grupo objetivo utiliza el HTTP protocolo, el puerto 80 y la configuración de comprobación de estado predeterminada para un grupo HTTP objetivo.

```
aws elbv2 create-target-group \  
  --name my-targets \  
  --protocol HTTP \  
  --port 80 \  
  --target-type instance \  
  --vpc-id vpc-3ac0fb5f
```

Salida:

```
{  
  "TargetGroups": [  
    {  
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",  
      "TargetGroupName": "my-targets",  
      "Protocol": "HTTP",  
      "Port": 80,  
      "VpcId": "vpc-3ac0fb5f",  
      "HealthCheckProtocol": "HTTP",  
      "HealthCheckPort": "traffic-port",  
      "HealthCheckEnabled": true,  
      "HealthCheckIntervalSeconds": 30,  
      "HealthCheckTimeoutSeconds": 5,  
      "HealthyThresholdCount": 5,  
      "UnhealthyThresholdCount": 2,  
      "HealthCheckPath": "/",  
      "Matcher": {  
        "HttpCode": "200"  
      },  
    },  
  ],  
}
```

```

        "TargetType": "instance",
        "ProtocolVersion": "HTTP1",
        "IpAddressType": "ipv4"
    }
]
}

```

Para obtener más información, consulte [Crear un grupo de destino](#) en la Guía del usuario para Equilibradores de carga de aplicación.

Ejemplo 2: Para crear un grupo objetivo para enrutar el tráfico desde un Application Load Balancer a una función Lambda

En el siguiente ejemplo de `create-target-group` se crea un grupo de destino para un Equilibrador de carga de aplicación donde el destino es una función de Lambda (el tipo de destino es `lambda`). De forma predeterminada, las comprobaciones de estado están deshabilitadas para este grupo de destino.

```

aws elbv2 create-target-group \
  --name my-lambda-target \
  --target-type lambda

```

Salida:

```

{
  "TargetGroups": [
    {
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-lambda-target/a3003e085dbb8ddc",
      "TargetGroupName": "my-lambda-target",
      "HealthCheckEnabled": false,
      "HealthCheckIntervalSeconds": 35,
      "HealthCheckTimeoutSeconds": 30,
      "HealthyThresholdCount": 5,
      "UnhealthyThresholdCount": 2,
      "HealthCheckPath": "/",
      "Matcher": {
        "HttpCode": "200"
      },
      "TargetType": "lambda",
      "IpAddressType": "ipv4"
    }
  ]
}

```



```
]
}
```

Para obtener más información, consulte [Funciones de Lambda como destinos](#) en la Guía del usuario de Equilibradores de carga de aplicación.

Ejemplo 3: Para crear un grupo objetivo para un Network Load Balancer

En el siguiente ejemplo de `create-target-group` se crea un grupo de destino para un Equilibrador de carga de red en el que se registran los destinos por dirección IP (el tipo de destino es `ip`). Este grupo objetivo usa el TCP protocolo, el puerto 80 y la configuración de comprobación de estado predeterminada para un grupo TCP objetivo.

```
aws elbv2 create-target-group \
  --name my-ip-targets \
  --protocol TCP \
  --port 80 \
  --target-type ip \
  --vpc-id vpc-3ac0fb5f
```

Salida:

```
{
  "TargetGroups": [
    {
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-ip-targets/b6bba954d1361c78",
      "TargetGroupName": "my-ip-targets",
      "Protocol": "TCP",
      "Port": 80,
      "VpcId": "vpc-3ac0fb5f",
      "HealthCheckEnabled": true,
      "HealthCheckProtocol": "TCP",
      "HealthCheckPort": "traffic-port",
      "HealthCheckIntervalSeconds": 30,
      "HealthCheckTimeoutSeconds": 10,
      "HealthyThresholdCount": 5,
      "UnhealthyThresholdCount": 2,
      "TargetType": "ip",
      "IpAddressType": "ipv4"
    }
  ]
}
```

```
}
```

Para obtener más información, consulte [Crear un grupo objetivo](#) en la Guía del usuario de los balanceadores de carga de red.

Ejemplo 4: Para crear un grupo objetivo para enrutar el tráfico de un Network Load Balancer a un Application Load Balancer

En el siguiente `create-target-group` ejemplo, se crea un grupo objetivo para un Network Load Balancer en el que se registra un Application Load Balancer como destino (el tipo de destino es). `alb`

```
aws elbv2 create-target-group --name my-alb-target --protocol --port 80 --target-type alb --vpc-id vpc-3ac0fb5f TCP
```

Salida:

```
{
  "TargetGroups": [
    {
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-alb-target/a3003e085dbb8ddc",
      "TargetGroupName": "my-alb-target",
      "Protocol": "TCP",
      "Port": 80,
      "VpcId": "vpc-838475fe",
      "HealthCheckProtocol": "HTTP",
      "HealthCheckPort": "traffic-port",
      "HealthCheckEnabled": true,
      "HealthCheckIntervalSeconds": 30,
      "HealthCheckTimeoutSeconds": 6,
      "HealthyThresholdCount": 5,
      "UnhealthyThresholdCount": 2,
      "HealthCheckPath": "/",
      "Matcher": {
        "HttpCode": "200-399"
      },
      "TargetType": "alb",
      "IpAddressType": "ipv4"
    }
  ]
}
```

Para obtener más información, consulte [Crear un grupo objetivo con un Application Load Balancer como destino en la Guía del usuario de Network Load Balancers](#).

Ejemplo 5: Para crear un grupo objetivo para un Gateway Load Balancer

El siguiente `create-target-group` ejemplo crea un grupo objetivo para un Gateway Load Balancer en el que el destino es una instancia y el protocolo del grupo objetivo es. GENEVE

```
aws elbv2 create-target-group \  
  --name my-glb-targetgroup \  
  --protocol GENEVE \  
  --port 6081 \  
  --target-type instance \  
  --vpc-id vpc-838475fe
```

Salida:

```
{  
  "TargetGroups": [  
    {  
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-glb-targetgroup/00c3d57eacd6f40b6f",  
      "TargetGroupName": "my-glb-targetgroup",  
      "Protocol": "GENEVE",  
      "Port": 6081,  
      "VpcId": "vpc-838475fe",  
      "HealthCheckProtocol": "TCP",  
      "HealthCheckPort": "80",  
      "HealthCheckEnabled": true,  
      "HealthCheckIntervalSeconds": 10,  
      "HealthCheckTimeoutSeconds": 5,  
      "HealthyThresholdCount": 5,  
      "UnhealthyThresholdCount": 2,  
      "TargetType": "instance"  
    }  
  ]  
}
```

Para obtener más información, consulte [Crear un grupo objetivo < https://docs.aws.amazon.com/elasticloadbalancing/latest/gateway/create-target-group .html>](https://docs.aws.amazon.com/elasticloadbalancing/latest/gateway/create-target-group.html) en la Guía del usuario del Gateway Load Balancer.

- Para obtener API más información, [CreateTargetGroup](#) consulte AWS CLI la Referencia de comandos.

## delete-listener

En el siguiente ejemplo de código, se muestra cómo usar `delete-listener`.

### AWS CLI

Para eliminar un oyente

En el siguiente `delete-listener` ejemplo, se elimina el oyente especificado.

```
aws elbv2 delete-listener \  
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2
```

- Para API obtener más información, consulte la Referencia [DeleteListener](#) de AWS CLI comandos.

## delete-load-balancer

En el siguiente ejemplo de código, se muestra cómo usar `delete-load-balancer`.

### AWS CLI

Eliminación de un equilibrador de carga

En el siguiente ejemplo de `delete-load-balancer` se elimina el equilibrador de carga especificado.

```
aws elbv2 delete-load-balancer \  
  --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188
```

- Para API obtener más información, consulte [DeleteLoadBalancer](#) la Referencia de AWS CLI comandos.

## delete-rule

En el siguiente ejemplo de código, se muestra cómo usar `delete-rule`.

## AWS CLI

Para eliminar una regla

En el siguiente `delete-rule` ejemplo, se elimina la regla especificada.

```
aws elbv2 delete-rule \  
  --rule-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-rule/  
app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/1291d13826f405c3
```

- Para API obtener más información, consulte [DeleteRule](#) la Referencia de AWS CLI comandos.

## `delete-target-group`

En el siguiente ejemplo de código, se muestra cómo usar `delete-target-group`.

## AWS CLI

Eliminación de un grupo de destino

En el siguiente ejemplo de `delete-target-group` se elimina el grupo de destino especificado.

```
aws elbv2 delete-target-group \  
  --target-group-arn arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un balanceador de carga en la Guía Application Load Balancer](#).

- Para API obtener más información, consulte la Referencia [DeleteTargetGroup](#) de AWS CLI comandos.

## `deregister-targets`

En el siguiente ejemplo de código, se muestra cómo usar `deregister-targets`.

## AWS CLI

Ejemplo 1: Para anular el registro de un objetivo de un grupo objetivo

En el siguiente `deregister-targets` ejemplo, se elimina la instancia especificada del grupo de destino especificado.

```
aws elbv2 deregister-targets \  
  --target-group-arn arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067 \  
  --targets Id=i-1234567890abcdef0
```

Ejemplo 2: Para anular el registro de un objetivo registrado mediante anulaciones de puertos

En el siguiente `deregister-targets` ejemplo, se elimina una instancia de un grupo de destino que se registró mediante anulaciones de puertos.

```
aws elbv2 deregister-targets \  
  --target-group-arn arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-internal-targets/3bb63f11dfb0faf9 \  
  --targets Id=i-1234567890abcdef0,Port=80 Id=i-1234567890abcdef0,Port=766
```

- Para API obtener más información, consulte [DeregisterTargets](#) la Referencia de AWS CLI comandos.

## **describe-account-limits**

En el siguiente ejemplo de código, se muestra cómo usar `describe-account-limits`.

### AWS CLI

Para describir sus límites de Elastic Load Balancing

En el siguiente `describe-account-limits` ejemplo, se muestran los límites de Elastic Load Balancing para su AWS cuenta en la región actual.

```
aws elbv2 describe-account-limits
```

Salida:

```
{  
  "Limits": [  
    {  
      "Name": "target-groups",  
      "Max": "3000"    }  
  ]  
}
```

```
  },
  {
    "Name": "targets-per-application-load-balancer",
    "Max": "1000"
  },
  {
    "Name": "listeners-per-application-load-balancer",
    "Max": "50"
  },
  {
    "Name": "rules-per-application-load-balancer",
    "Max": "100"
  },
  {
    "Name": "network-load-balancers",
    "Max": "50"
  },
  {
    "Name": "targets-per-network-load-balancer",
    "Max": "3000"
  },
  {
    "Name": "targets-per-availability-zone-per-network-load-balancer",
    "Max": "500"
  },
  {
    "Name": "listeners-per-network-load-balancer",
    "Max": "50"
  },
  {
    "Name": "condition-values-per-alb-rule",
    "Max": "5"
  },
  {
    "Name": "condition-wildcards-per-alb-rule",
    "Max": "5"
  },
  {
    "Name": "target-groups-per-application-load-balancer",
    "Max": "100"
  },
  {
    "Name": "target-groups-per-action-on-application-load-balancer",
    "Max": "5"
  }
```

```
  },
  {
    "Name": "target-groups-per-action-on-network-load-balancer",
    "Max": "1"
  },
  {
    "Name": "certificates-per-application-load-balancer",
    "Max": "25"
  },
  {
    "Name": "certificates-per-network-load-balancer",
    "Max": "25"
  },
  {
    "Name": "targets-per-target-group",
    "Max": "1000"
  },
  {
    "Name": "target-id-registrations-per-application-load-balancer",
    "Max": "1000"
  },
  {
    "Name": "network-load-balancer-enis-per-vpc",
    "Max": "1200"
  },
  {
    "Name": "application-load-balancers",
    "Max": "50"
  },
  {
    "Name": "gateway-load-balancers",
    "Max": "100"
  },
  {
    "Name": "gateway-load-balancers-per-vpc",
    "Max": "100"
  },
  {
    "Name": "geneve-target-groups",
    "Max": "100"
  },
  {
    "Name": "targets-per-availability-zone-per-gateway-load-balancer",
    "Max": "300"
  }
```



```

    }
  ]
}

```

Para obtener más información, consulte [Cuotas](#) en la referencia AWS general.

- Para API obtener más información, consulte [DescribeAccountLimits](#) la Referencia de AWS CLI comandos.

## describe-listener-certificates

En el siguiente ejemplo de código, se muestra cómo usar `describe-listener-certificates`.

### AWS CLI

Para describir los certificados de un agente de escucha seguro

En este ejemplo, se describen los certificados del agente de escucha seguro especificado.

Comando:

```
aws elbv2 describe-listener-certificates --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2
```

Salida:

```
{
  "Certificates": [
    {
      "CertificateArn": "arn:aws:acm:us-west-2:123456789012:certificate/5cc54884-f4a3-4072-80be-05b9ba72f705",
      "IsDefault": false
    },
    {
      "CertificateArn": "arn:aws:acm:us-west-2:123456789012:certificate/3dcb0a41-bd72-4774-9ad9-756919c40557",
      "IsDefault": false
    },
    {
      "CertificateArn": "arn:aws:acm:us-west-2:123456789012:certificate/fe59da96-6f58-4a22-8eed-6d0d50477e1d",

```

```

        "IsDefault": true
    }
]
}

```

- Para API obtener más información, consulte [DescribeListenerCertificates](#) la Referencia de AWS CLI comandos.

## describe-listeners

En el siguiente ejemplo de código, se muestra cómo usar `describe-listeners`.

### AWS CLI

Para describir a un oyente

En este ejemplo se describe el oyente especificado.

Comando:

```
aws elbv2 describe-listeners --listener-arns arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2
```

Salida:

```

{
  "Listeners": [
    {
      "Port": 80,
      "Protocol": "HTTP",
      "DefaultActions": [
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
          "Type": "forward"
        }
      ],
      "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
      "ListenerArn": "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2"
    }
  ]
}

```

```
]
}
```

Para describir los oyentes de un balanceador de cargas

En este ejemplo, se describen los oyentes del balanceador de cargas especificado.

Comando:

```
aws elbv2 describe-listeners --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188
```

Salida:

```
{
  "Listeners": [
    {
      "Port": 443,
      "Protocol": "HTTPS",
      "DefaultActions": [
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
          "Type": "forward"
        }
      ],
      "SslPolicy": "ELBSecurityPolicy-2015-05",
      "Certificates": [
        {
          "CertificateArn": "arn:aws:iam::123456789012:server-certificate/my-server-cert"
        }
      ],
      "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
      "ListenerArn": "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/0467ef3c8400ae65"
    },
    {
      "Port": 80,
      "Protocol": "HTTP",
      "DefaultActions": [
        {
```

```

        "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
        "Type": "forward"
    }
],
    "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
    "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2"
}
]
}

```

- Para API obtener más información, consulta la Referencia [DescribeListeners](#) de AWS CLI comandos.

## describe-load-balancer-attributes

En el siguiente ejemplo de código, se muestra cómo usar `describe-load-balancer-attributes`.

### AWS CLI

Para describir los atributos del equilibrador de carga

En el siguiente `describe-load-balancer-attributes` ejemplo, se muestran los atributos del equilibrador de cargas especificado.

```

aws elbv2 describe-load-balancer-attributes \
  --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188

```

El siguiente resultado de ejemplo muestra los atributos de un Application Load Balancer.

```

{
  "Attributes": [
    {
      "Value": "false",
      "Key": "access_logs.s3.enabled"
    },
    {
      "Value": "",

```

```
    "Key": "access_logs.s3.bucket"
  },
  {
    "Value": "",
    "Key": "access_logs.s3.prefix"
  },
  {
    "Value": "60",
    "Key": "idle_timeout.timeout_seconds"
  },
  {
    "Value": "false",
    "Key": "deletion_protection.enabled"
  },
  {
    "Value": "true",
    "Key": "routing.http2.enabled"
  }
]
}
```

El siguiente ejemplo de salida incluye los atributos de un Network Load Balancer.

```
{
  "Attributes": [
    {
      "Value": "false",
      "Key": "access_logs.s3.enabled"
    },
    {
      "Value": "",
      "Key": "access_logs.s3.bucket"
    },
    {
      "Value": "",
      "Key": "access_logs.s3.prefix"
    },
    {
      "Value": "false",
      "Key": "deletion_protection.enabled"
    },
    {
      "Value": "false",
```

```

        "Key": "load_balancing.cross_zone.enabled"
      }
    ]
  }
}

```

- Para API obtener más información, consulte [DescribeLoadBalancerAttributes](#) la Referencia de AWS CLI comandos.

## describe-load-balancers

En el siguiente ejemplo de código, se muestra cómo usar `describe-load-balancers`.

### AWS CLI

Descripción de un equilibrador de cargas

En este ejemplo se describe el equilibrador de carga especificado.

Comando:

```

aws elbv2 describe-load-balancers --load-balancer-
arns arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-
balancer/50dc6c495c0c9188

```

Salida:

```

{
  "LoadBalancers": [
    {
      "Type": "application",
      "Scheme": "internet-facing",
      "IpAddressType": "ipv4",
      "VpcId": "vpc-3ac0fb5f",
      "AvailabilityZones": [
        {
          "ZoneName": "us-west-2a",
          "SubnetId": "subnet-8360a9e7"
        },
        {
          "ZoneName": "us-west-2b",
          "SubnetId": "subnet-b7d581c0"
        }
      ]
    }
  ],
}

```

```

    "CreatedTime": "2016-03-25T21:26:12.920Z",
    "CanonicalHostedZoneId": "Z2P70J7EXAMPLE",
    "DNSName": "my-load-balancer-424835706.us-west-2.elb.amazonaws.com",
    "SecurityGroups": [
      "sg-5943793c"
    ],
    "LoadBalancerName": "my-load-balancer",
    "State": {
      "Code": "active"
    },
    "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188"
  }
]
}

```

Descripción de todos los equilibradores de carga

En este ejemplo, se describen todos los equilibradores de carga.

Comando:

```
aws elbv2 describe-load-balancers
```

- Para API obtener más información, consulte [DescribeLoadBalancers](#) la Referencia de AWS CLI comandos.

## describe-rules

En el siguiente ejemplo de código, se muestra cómo usar `describe-rules`.

AWS CLI

Ejemplo 1: Para describir una regla

En el siguiente `describe-rules` ejemplo, se muestran los detalles de la regla especificada.

```
aws elbv2 describe-rules \
  --rule-arns arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-rule/
app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/9683b2d02a6cabee
```

Ejemplo 2: Para describir las reglas para un oyente

En el siguiente `describe-rules` ejemplo, se muestran los detalles de las reglas del oyente especificado. El resultado incluye la regla predeterminada y cualquier otra regla que haya agregado.

```
aws elbv2 describe-rules \  
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2
```

- Para API obtener más información, consulte [DescribeRules](#) la Referencia de AWS CLI comandos.

## `describe-ssl-policies`

En el siguiente ejemplo de código, se muestra cómo usar `describe-ssl-policies`.

### AWS CLI

Ejemplo 1: Para enumerar las políticas utilizadas para la SSL negociación por tipo de balanceador de carga

En el siguiente `describe-ssl-policies` ejemplo, se muestran los nombres de las políticas que puede usar para SSL negociar con un Application Load Balancer. En el ejemplo, se utiliza el `--query` parámetro para mostrar solo los nombres de las políticas.

```
aws elbv2 describe-ssl-policies \  
  --load-balancer-type application \  
  --query SslPolicies[*].Name
```

Salida:

```
[  
  "ELBSecurityPolicy-2016-08",  
  "ELBSecurityPolicy-TLS13-1-2-2021-06",  
  "ELBSecurityPolicy-TLS13-1-2-Res-2021-06",  
  "ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06",  
  "ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06",  
  "ELBSecurityPolicy-TLS13-1-1-2021-06",  
  "ELBSecurityPolicy-TLS13-1-0-2021-06",  
  "ELBSecurityPolicy-TLS13-1-3-2021-06",  
  "ELBSecurityPolicy-TLS-1-2-2017-01",  
  "ELBSecurityPolicy-TLS-1-1-2017-01",
```



```

"ELBSecurityPolicy-TLS-1-2-Ext-2018-06",
"ELBSecurityPolicy-FS-2018-06",
"ELBSecurityPolicy-2015-05",
"ELBSecurityPolicy-TLS-1-0-2015-04",
"ELBSecurityPolicy-FS-1-2-Res-2019-08",
"ELBSecurityPolicy-FS-1-1-2019-08",
"ELBSecurityPolicy-FS-1-2-2019-08",
"ELBSecurityPolicy-FS-1-2-Res-2020-10"
]

```

Ejemplo 2: Para enumerar las políticas que admiten un protocolo específico

En el siguiente `describe-ssl-policies` ejemplo, se muestran los nombres de las políticas que admiten el protocolo TLS 1.3. En el ejemplo, se utiliza el `--query` parámetro para mostrar únicamente los nombres de las políticas.

```

aws elbv2 describe-ssl-policies \
  --load-balancer-type application \
  --query SslPolicies[?contains(SslProtocols, 'TLSv1.3')].Name

```

Salida:

```

[
  "ELBSecurityPolicy-TLS13-1-2-2021-06",
  "ELBSecurityPolicy-TLS13-1-2-Res-2021-06",
  "ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06",
  "ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06",
  "ELBSecurityPolicy-TLS13-1-1-2021-06",
  "ELBSecurityPolicy-TLS13-1-0-2021-06",
  "ELBSecurityPolicy-TLS13-1-3-2021-06"
]

```

Ejemplo 3: Para mostrar los cifrados de una política

En el siguiente `describe-ssl-policies` ejemplo, se muestran los nombres de los cifrados de la política especificada. En el ejemplo, se utiliza el `--query` parámetro para mostrar solo los nombres de los cifrados. El primer cifrado de la lista tiene la prioridad 1 y los demás cifrados están en orden de prioridad.

```

aws elbv2 describe-ssl-policies \
  --names ELBSecurityPolicy-TLS13-1-2-2021-06 \

```

```
--query SslPolicies[*].Ciphers[*].Name
```

Salida:

```
[  
  "TLS_AES_128_GCM_SHA256",  
  "TLS_AES_256_GCM_SHA384",  
  "TLS_CHACHA20_POLY1305_SHA256",  
  "ECDHE-ECDSA-AES128-GCM-SHA256",  
  "ECDHE-RSA-AES128-GCM-SHA256",  
  "ECDHE-ECDSA-AES128-SHA256",  
  "ECDHE-RSA-AES128-SHA256",  
  "ECDHE-ECDSA-AES256-GCM-SHA384",  
  "ECDHE-RSA-AES256-GCM-SHA384",  
  "ECDHE-ECDSA-AES256-SHA384",  
  "ECDHE-RSA-AES256-SHA384"  
]
```

Para obtener más información, consulte [las políticas de seguridad](#) en la Guía del usuario de los balanceadores de carga de aplicaciones.

- Para API obtener más información, consulte [DescribeSslPolicies](#) la Referencia de AWS CLI comandos.

## describe-tags

En el siguiente ejemplo de código, se muestra cómo usar `describe-tags`.

AWS CLI

Para describir las etiquetas asignadas a un balanceador de cargas

En este ejemplo, se describen las etiquetas asignadas al balanceador de cargas especificado.

Comando:

```
aws elbv2 describe-tags --resource-arns arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188
```

Salida:

```
{
```

```
"TagDescriptions": [
  {
    "ResourceArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
    "Tags": [
      {
        "Value": "lima",
        "Key": "project"
      },
      {
        "Value": "digital-media",
        "Key": "department"
      }
    ]
  }
]
```

- Para API obtener más información, consulta [DescribeTags](#) la Referencia de AWS CLI comandos.

## describe-target-group-attributes

En el siguiente ejemplo de código, se muestra cómo usar `describe-target-group-attributes`.

### AWS CLI

Para describir los atributos del grupo objetivo

En el siguiente `describe-target-group-attributes` ejemplo, se muestran los atributos del grupo objetivo especificado.

```
aws elbv2 describe-target-group-attributes \
  --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

El resultado incluye los atributos si el protocolo es HTTP o HTTPS y el tipo de destino es `instance` o `ip`.

```
{
```

```

    "Attributes": [
      {
        "Value": "false",
        "Key": "stickiness.enabled"
      },
      {
        "Value": "300",
        "Key": "deregistration_delay.timeout_seconds"
      },
      {
        "Value": "lb_cookie",
        "Key": "stickiness.type"
      },
      {
        "Value": "86400",
        "Key": "stickiness.lb_cookie.duration_seconds"
      },
      {
        "Value": "0",
        "Key": "slow_start.duration_seconds"
      }
    ]
  }

```

El siguiente resultado incluye los atributos si el protocolo es HTTP o HTTPS y el tipo de destino es lambda.

```

{
  "Attributes": [
    {
      "Value": "false",
      "Key": "lambda.multi_value_headers.enabled"
    }
  ]
}

```

El siguiente resultado incluye los atributos si el protocolo es TCPTLS,UDP, o TCP\_UDP.

```

{
  "Attributes": [
    {
      "Value": "false",
      "Key": "proxy_protocol_v2.enabled"
    }
  ]
}

```

```

    },
    {
      "Value": "300",
      "Key": "deregistration_delay.timeout_seconds"
    }
  ]
}

```

- Para API obtener más información, consulte [DescribeTargetGroupAttributes](#) la Referencia de AWS CLI comandos.

## describe-target-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-target-groups`.

### AWS CLI

Ejemplo 1: Descripción de un grupo de destino

En el siguiente ejemplo de `describe-target-groups` se muestran los detalles del grupo de destino especificado.

```

aws elbv2 describe-target-groups \
  --target-group-arns arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067

```

Salida:

```

{
  "TargetGroups": [
    {
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
      "TargetGroupName": "my-targets",
      "Protocol": "HTTP",
      "Port": 80,
      "VpcId": "vpc-3ac0fb5f",
      "HealthCheckProtocol": "HTTP",
      "HealthCheckPort": "traffic-port",
      "HealthCheckEnabled": true,
      "HealthCheckIntervalSeconds": 30,

```

```

    "HealthCheckTimeoutSeconds": 5,
    "HealthyThresholdCount": 5,
    "UnhealthyThresholdCount": 2,
    "HealthCheckPath": "/",
    "Matcher": {
      "HttpCode": "200"
    },
    "LoadBalancerArns": [
      "arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/
app/my-load-balancer/50dc6c495c0c9188"
    ],
    "TargetType": "instance",
    "ProtocolVersion": "HTTP1",
    "IpAddressType": "ipv4"
  }
]
}

```

Ejemplo 2: Descripción de todos los grupos de destino de un equilibrador de carga

En el siguiente ejemplo de `describe-target-groups` se muestran los detalles de todos los grupos de destino del equilibrador de cargas especificado. En el ejemplo, se utiliza el `--query` parámetro para mostrar solo los nombres de los grupos de destino.

```

aws elbv2 describe-target-groups \
  --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \
  --query TargetGroups[*].TargetGroupName

```

Salida:

```

[
  "my-instance-targets",
  "my-ip-targets",
  "my-lambda-target"
]

```

Para obtener más información, consulte [Grupos objetivo](#) en la Guía de balanceadores de carga de aplicaciones.

- Para API obtener más información, consulte la Referencia [DescribeTargetGroups](#) de AWS CLI comandos.

## describe-target-health

En el siguiente ejemplo de código, se muestra cómo usar `describe-target-health`.

### AWS CLI

Ejemplo 1: Descripción del estado de los destinos para un grupo de destino

En el siguiente ejemplo de `describe-target-health` se muestran los detalles del estado de los destinos del grupo de destino especificado. Estos destinos se encuentran en buen estado.

```
aws elbv2 describe-target-health \  
  --target-group-arn arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Salida:

```
{  
  "TargetHealthDescriptions": [  
    {  
      "HealthCheckPort": "80",  
      "Target": {  
        "Id": "i-ceddcd4d",  
        "Port": 80  
      },  
      "TargetHealth": {  
        "State": "healthy"  
      }  
    },  
    {  
      "HealthCheckPort": "80",  
      "Target": {  
        "Id": "i-0f76fade",  
        "Port": 80  
      },  
      "TargetHealth": {  
        "State": "healthy"  
      }  
    }  
  ]  
}
```

Ejemplo 2: Descripción del estado de un destino

En el siguiente ejemplo de `describe-target-health` se muestran los detalles del estado del destino especificado. Este destino se encuentra en buen estado.

```
aws elbv2 describe-target-health \
  --targets Id=i-0f76fade,Port=80 \
  --target-group-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Salida:

```
{
  "TargetHealthDescriptions": [
    {
      "HealthCheckPort": "80",
      "Target": {
        "Id": "i-0f76fade",
        "Port": 80
      },
      "TargetHealth": {
        "State": "healthy"
      }
    }
  ]
}
```

La siguiente salida de ejemplo es para un destino cuyo grupo de destino no está especificado en una acción para un oyente. Este destino no puede recibir el tráfico del equilibrador de carga.

```
{
  "TargetHealthDescriptions": [
    {
      "HealthCheckPort": "80",
      "Target": {
        "Id": "i-0f76fade",
        "Port": 80
      },
      "TargetHealth": {
        "State": "unused",
        "Reason": "Target.NotInUse",
        "Description": "Target group is not configured to receive traffic from the load balancer"
      }
    }
  ]
}
```



```
    }  
  ]  
}
```

La siguiente salida de ejemplo es para un destino cuyo grupo de destino no se especificó en una acción para un oyente. El destino sigue registrado.

```
{  
  "TargetHealthDescriptions": [  
    {  
      "HealthCheckPort": "80",  
      "Target": {  
        "Id": "i-0f76fade",  
        "Port": 80  
      },  
      "TargetHealth": {  
        "State": "initial",  
        "Reason": "Elb.RegistrationInProgress",  
        "Description": "Target registration is in progress"  
      }  
    }  
  ]  
}
```

El siguiente ejemplo de salida es un destino en mal estado.

```
{  
  "TargetHealthDescriptions": [  
    {  
      "HealthCheckPort": "80",  
      "Target": {  
        "Id": "i-0f76fade",  
        "Port": 80  
      },  
      "TargetHealth": {  
        "State": "unhealthy",  
        "Reason": "Target.Timeout",  
        "Description": "Connection to target timed out"  
      }  
    }  
  ]  
}
```

La siguiente salida de ejemplo es para un destino que es una función de Lambda y las comprobaciones de estado están deshabilitadas.

```
{
  "TargetHealthDescriptions": [
    {
      "Target": {
        "Id": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
        "AvailabilityZone": "all",
      },
      "TargetHealth": {
        "State": "unavailable",
        "Reason": "Target.HealthCheckDisabled",
        "Description": "Health checks are not enabled for this target"
      }
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeTargetHealth](#) la Referencia de AWS CLI comandos.

## modify-listener

En el siguiente ejemplo de código, se muestra cómo usar `modify-listener`.

### AWS CLI

Ejemplo 1: Para cambiar la acción por defecto por una acción de reenvío

El siguiente `modify-listener` ejemplo cambia la acción predeterminada (por una acción de reenvío) para el oyente especificado.

```
aws elbv2 modify-listener \
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
  --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-new-targets/2453ed029918f21f
```

Salida:

```
{
```

```

    "Listeners": [
      {
        "Protocol": "HTTP",
        "DefaultActions": [
          {
            "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-new-targets/2453ed029918f21f",
            "Type": "forward"
          }
        ],
        "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
        "Port": 80,
        "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2"
      }
    ]
  }

```

Ejemplo 2: Para cambiar la acción predeterminada por una acción de redireccionamiento

El siguiente `modify-listener` ejemplo cambia la acción predeterminada por una acción de redireccionamiento para el oyente especificado.

```

aws elbv2 modify-listener \
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
  --default-actions Type=redirect,TargetGroupArn=arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-new-targets/2453ed029918f21f

```

Salida:

```

{
  "Listeners": [
    {
      "Protocol": "HTTP",
      "DefaultActions": [
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-new-targets/2453ed029918f21f",
          "Type": "redirect"
        }
      ]
    }
  ]
}

```

```

    ],
    "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
    "Port": 80,
    "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2"
  }
]
}

```

### Ejemplo 3: Para cambiar el certificado del servidor

En este ejemplo, se cambia el certificado de servidor del agente de HTTPS escucha especificado.

```

aws elbv2 modify-listener \
  --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/0467ef3c8400ae65 \
  --certificates CertificateArn=arn:aws:iam::123456789012:server-certificate/my-
new-server-cert

```

Salida:

```

{
  "Listeners": [
    {
      "Protocol": "HTTPS",
      "DefaultActions": [
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
          "Type": "forward"
        }
      ],
      "SslPolicy": "ELBSecurityPolicy-2015-05",
      "Certificates": [
        {
          "CertificateArn": "arn:aws:iam::123456789012:server-certificate/
my-new-server-cert"
        }
      ],
      "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
      "Port": 443,

```

```
        "ListenerArn": "arn:aws:elasticloadbalancing:us-  
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/0467ef3c8400ae65"  
      }  
    ]  
  }  
}
```

- Para API obtener más información, consulte [ModifyListener](#) la Referencia de AWS CLI comandos.

## modify-load-balancer-attributes

En el siguiente ejemplo de código, se muestra cómo usar `modify-load-balancer-attributes`.

### AWS CLI

Para habilitar la protección contra la eliminación

Este ejemplo habilita la protección contra la eliminación para el balanceador de cargas especificado.

Comando:

```
aws elbv2 modify-load-balancer-attributes --load-balancer-  
arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-  
balancer/50dc6c495c0c9188 --attributes Key=deletion_protection.enabled,Value=true
```

Salida:

```
{  
  "Attributes": [  
    {  
      "Value": "true",  
      "Key": "deletion_protection.enabled"  
    },  
    {  
      "Value": "false",  
      "Key": "access_logs.s3.enabled"  
    },  
    {  
      "Value": "60",  
      "Key": "idle_timeout.timeout_seconds"  
    }  
  ]  
}
```

```
    },
    {
      "Value": "",
      "Key": "access_logs.s3.prefix"
    },
    {
      "Value": "",
      "Key": "access_logs.s3.bucket"
    }
  ]
}
```

Para cambiar el tiempo de espera de inactividad

En este ejemplo, se cambia el valor del tiempo de espera de inactividad del balanceador de cargas especificado.

Comando:

```
aws elbv2 modify-load-balancer-attributes --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 --attributes Key=idle_timeout.timeout_seconds,Value=30
```

Salida:

```
{
  "Attributes": [
    {
      "Value": "30",
      "Key": "idle_timeout.timeout_seconds"
    },
    {
      "Value": "false",
      "Key": "access_logs.s3.enabled"
    },
    {
      "Value": "",
      "Key": "access_logs.s3.prefix"
    },
    {
      "Value": "true",
      "Key": "deletion_protection.enabled"
    }
  ]
}
```

```

    },
    {
      "Value": "",
      "Key": "access_logs.s3.bucket"
    }
  ]
}

```

Para habilitar los registros de acceso

En este ejemplo, se habilitan los registros de acceso para el balanceador de cargas especificado. Tenga en cuenta que el bucket de S3 debe estar en la misma región que el balanceador de cargas y debe tener una política adjunta que permita el acceso al servicio Elastic Load Balancing.

Comando:

```

aws elbv2 modify-load-balancer-attributes --load-balancer-
arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-
balancer/50dc6c495c0c9188 --
attributes Key=access_logs.s3.enabled,Value=true Key=access_logs.s3.bucket,Value=my-
loadbalancer-logs Key=access_logs.s3.prefix,Value=myapp

```

Salida:

```

{
  "Attributes": [
    {
      "Value": "true",
      "Key": "access_logs.s3.enabled"
    },
    {
      "Value": "my-load-balancer-logs",
      "Key": "access_logs.s3.bucket"
    },
    {
      "Value": "myapp",
      "Key": "access_logs.s3.prefix"
    },
    {
      "Value": "60",
      "Key": "idle_timeout.timeout_seconds"
    }
  ],
}

```

```
{
  "Value": "false",
  "Key": "deletion_protection.enabled"
}
]
```

- Para API obtener más información, consulte [ModifyLoadBalancerAttributes](#) la Referencia de AWS CLI comandos.

## modify-rule

En el siguiente ejemplo de código, se muestra cómo usar `modify-rule`.

### AWS CLI

Para modificar una regla

En el siguiente `modify-rule` ejemplo, se actualizan las acciones y condiciones de la regla especificada.

```
aws elbv2 modify-rule \  
  --actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-  
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067 \  
  --conditions Field=path-pattern,Values='/images/*' \  
  --rule-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-rule/app/  
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/9683b2d02a6cabee
```

Salida:

```
{
  "Rules": [
    {
      "Priority": "10",
      "Conditions": [
        {
          "Field": "path-pattern",
          "Values": [
            "/images/*"
          ]
        }
      ]
    }
  ]
}
```



```

    ],
    "RuleArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener-rule/app/my-load-balancer/50dc6c495c0c9188/
f2f7dc8efc522ab2/9683b2d02a6cabee",
    "IsDefault": false,
    "Actions": [
      {
        "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
        "Type": "forward"
      }
    ]
  }
]
}

```

- Para API obtener más información, consulte [ModifyRule](#) la Referencia de AWS CLI comandos.

## modify-target-group-attributes

En el siguiente ejemplo de código, se muestra cómo usar `modify-target-group-attributes`.

### AWS CLI

Para modificar el tiempo de espera de la cancelación del registro

En este ejemplo, se establece el tiempo de espera de la cancelación del registro en el valor especificado para el grupo objetivo especificado.

Comando:

```

aws elbv2 modify-target-group-attributes --target-group-
arn arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067 --
attributes Key=deregistration_delay.timeout_seconds,Value=600

```

Salida:

```

{
  "Attributes": [
    {
      "Value": "false",

```

```

    "Key": "stickiness.enabled"
  },
  {
    "Value": "600",
    "Key": "deregistration_delay.timeout_seconds"
  },
  {
    "Value": "lb_cookie",
    "Key": "stickiness.type"
  },
  {
    "Value": "86400",
    "Key": "stickiness.lb_cookie.duration_seconds"
  }
]
}

```

- Para API obtener más información, consulte la Referencia de [ModifyTargetGroupAttributes](#) comandos AWS CLI .

## modify-target-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-target-group`.

### AWS CLI

Para modificar la configuración de la comprobación de estado de un grupo objetivo

El siguiente `modify-target-group` ejemplo cambia la configuración de las comprobaciones de estado utilizadas para evaluar el estado de los objetivos del grupo objetivo especificado. Tenga en cuenta que, debido a la forma en que CLI analiza las comas, debe rodear el rango de la `--matcher` opción con comillas simples en lugar de comillas dobles.

```

aws elbv2 modify-target-group \
  --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-https-targets/2453ed029918f21f \
  --health-check-protocol HTTPS \
  --health-check-port 443 \
  --matcher HttpCode='200,299'

```

Salida:

```
{
  "TargetGroups": [
    {
      "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-https-targets/2453ed029918f21f",
      "TargetGroupName": "my-https-targets",
      "Protocol": "HTTPS",
      "Port": 443,
      "VpcId": "vpc-3ac0fb5f",
      "HealthCheckProtocol": "HTTPS",
      "HealthCheckPort": "443",
      "HealthCheckEnabled": true,
      "HealthCheckIntervalSeconds": 30,
      "HealthCheckTimeoutSeconds": 5,
      "HealthyThresholdCount": 5,
      "UnhealthyThresholdCount": 2,
      "Matcher": {
        "HttpCode": "200,299"
      },
      "LoadBalancerArns": [
        "arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/
app/my-load-balancer/50dc6c495c0c9188"
      ],
      "TargetType": "instance",
      "ProtocolVersion": "HTTP1",
      "IpAddressType": "ipv4"
    }
  ]
}
```

Para obtener más información, consulte los [grupos objetivo](#) en la Guía de balanceadores de carga de aplicaciones.

- Para API obtener más información, consulte la Referencia [ModifyTargetGroup](#) de AWS CLI comandos.

## register-targets

En el siguiente ejemplo de código, se muestra cómo usar `register-targets`.

### AWS CLI

Ejemplo 1: Para registrar objetivos con un grupo de objetivos por ID de instancia

En el siguiente `register-targets` ejemplo, se registran las instancias especificadas con un grupo de destino. El grupo objetivo debe tener un tipo de destino de `instance`.

```
aws elbv2 register-targets \  
  --target-group-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067 \  
  --targets Id=i-1234567890abcdef0 Id=i-0abcdef1234567890
```

Ejemplo 2: Para registrar los destinos en un grupo objetivo mediante anulaciones de puertos

En el siguiente `register-targets` ejemplo, se registra la instancia especificada en un grupo de destino mediante varios puertos. Esto le permite registrar los contenedores en la misma instancia que los destinos del grupo de destino.

```
aws elbv2 register-targets \  
  --target-group-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-internal-targets/3bb63f11dfb0faf9 \  
  --targets Id=i-0598c7d356eba48d7,Port=80 Id=i-0598c7d356eba48d7,Port=766
```

Ejemplo 3: Para registrar los objetivos con un grupo de objetivos por dirección IP

El siguiente `register-targets` ejemplo registra las direcciones IP especificadas en un grupo de destino. El grupo objetivo debe tener un tipo de destino de `ip`.

```
aws elbv2 register-targets \  
  --target-group-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-tcp-ip-targets/8518e899d173178f \  
  --targets Id=10.0.1.15 Id=10.0.1.23
```

Ejemplo 4: Para registrar una función Lambda como destino

El siguiente `register-targets` ejemplo registra las direcciones IP especificadas en un grupo objetivo. El grupo objetivo debe tener un tipo de destino de `lambda`. Debe conceder el permiso a Elastic Load Balancing para invocar la función Lambda.

```
aws elbv2 register-targets \  
  --target-group-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-tcp-ip-targets/8518e899d173178f \  
  --targets Id=arn:aws:lambda:us-west-2:123456789012:function:my-function
```

- Para API obtener más información, consulte la Referencia [RegisterTargets](#) de AWS CLI comandos.

## **remove-listener-certificates**

En el siguiente ejemplo de código, se muestra cómo usar `remove-listener-certificates`.

### AWS CLI

Para eliminar un certificado de un agente de escucha seguro

En este ejemplo, se quita el certificado especificado del agente de escucha seguro especificado.

Comando:

```
aws elbv2 remove-listener-certificates --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 --certificates CertificateArn=arn:aws:acm:us-west-2:123456789012:certificate/5cc54884-f4a3-4072-80be-05b9ba72f705
```

- Para API obtener más información, consulte [RemoveListenerCertificates](#) la Referencia de AWS CLI comandos.

## **remove-tags**

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags`.

### AWS CLI

Para eliminar etiquetas de un balanceador de cargas

En el siguiente `remove-tags` ejemplo, se eliminan las `department` etiquetas `project` y del equilibrador de cargas especificado.

```
aws elbv2 remove-tags \
  --resource-arns arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \
  --tag-keys project department
```

- Para API obtener más información, consulte [RemoveTags](#) la Referencia de AWS CLI comandos.

## set-ip-address-type

En el siguiente ejemplo de código, se muestra cómo usar `set-ip-address-type`.

### AWS CLI

Para configurar el tipo de dirección de un balanceador de cargas

En este ejemplo, se establece el tipo de dirección del balanceador de cargas especificado en `dualstack`. Las subredes del balanceador de carga deben tener bloques asociados. IPv6 CIDR

Comando:

```
aws elbv2 set-ip-address-type --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 --ip-address-type dualstack
```

Salida:

```
{
  "IpAddressType": "dualstack"
}
```

- Para API obtener más información, consulte la Referencia [SetIpAddressType](#) de AWS CLI comandos.

## set-rule-priorities

En el siguiente ejemplo de código, se muestra cómo usar `set-rule-priorities`.

### AWS CLI

Para establecer la prioridad de la regla

En este ejemplo se establece la prioridad de la regla especificada.

Comando:

```
aws elbv2 set-rule-priorities --rule-priorities RuleArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-rule/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/1291d13826f405c3,Priority=5
```

**Salida:**

```
{
  "Rules": [
    {
      "Priority": "5",
      "Conditions": [
        {
          "Field": "path-pattern",
          "Values": [
            "/img/*"
          ]
        }
      ],
      "RuleArn": "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-rule/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/1291d13826f405c3",
      "IsDefault": false,
      "Actions": [
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
          "Type": "forward"
        }
      ]
    }
  ]
}
```

- Para API obtener más información, consulte [SetRulePriorities](#) la Referencia de AWS CLI comandos.

**set-security-groups**

En el siguiente ejemplo de código, se muestra cómo usar `set-security-groups`.

**AWS CLI**

Para asociar un grupo de seguridad a un balanceador de cargas

En este ejemplo, se asocia el grupo de seguridad especificado al equilibrador de cargas especificado.

Comando:

```
aws elbv2 set-security-groups --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 --security-groups sg-5943793c
```

Salida:

```
{
  "SecurityGroupIds": [
    "sg-5943793c"
  ]
}
```

- Para API obtener más información, consulte [SetSecurityGroups](#) la Referencia de AWS CLI comandos.

## set-subnets

En el siguiente ejemplo de código, se muestra cómo usar set-subnets.

### AWS CLI

Para habilitar las zonas de disponibilidad para un balanceador de cargas

En este ejemplo, se habilita la zona de disponibilidad de la subred especificada para el equilibrador de carga especificado.

Comando:

```
aws elbv2 set-subnets --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 --subnets subnet-8360a9e7 subnet-b7d581c0
```

Salida:

```
{
  "AvailabilityZones": [
    {
      "SubnetId": "subnet-8360a9e7",
      "ZoneName": "us-west-2a"
    },
  ],
}
```



```
{
  "SubnetId": "subnet-b7d581c0",
  "ZoneName": "us-west-2b"
}
]
```

- Para API obtener más información, consulte la Referencia [SetSubnets](#) de AWS CLI comandos.

## Ejemplos de Elastic Transcoder usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Elastic Transcoder.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

#### **cancel-job**

En el siguiente ejemplo de código, se muestra cómo usar `cancel-job`.

#### AWS CLI

Para cancelar un trabajo para ElasticTranscoder

Esto cancela el trabajo especificado para ElasticTranscoder.

Comando:

```
aws elastictranscoder cancel-job --id 333333333333-abcde3
```

- Para API obtener más información, consulte [CancelJob](#) la Referencia de AWS CLI comandos.

## create-job

En el siguiente ejemplo de código, se muestra cómo usar `create-job`.

### AWS CLI

Para crear un trabajo para ElasticTranscoder

En el siguiente `create-job` ejemplo se crea un trabajo para ElasticTranscoder.

```
aws elastictranscoder create-job \  
  --pipeline-id 111111111111-abcde1 \  
  --inputs file://inputs.json \  
  --outputs file://outputs.json \  
  --output-key-prefix "recipes/" \  
  --user-metadata file://user-metadata.json
```

Contenidos de `inputs.json`:

```
[{  
  "Key": "ETS_example_file.mp4",  
  "FrameRate": "auto",  
  "Resolution": "auto",  
  "AspectRatio": "auto",  
  "Interlaced": "auto",  
  "Container": "mp4"  
}]
```

Contenido de `outputs.json`:

```
[  
  {  
    "Key": "webm/ETS_example_file-kindlefirehd.webm",  
    "Rotate": "0",  
    "PresetId": "1351620000001-100250"  
  }  
]
```

Contenidos de `user-metadata.json`:

```
{
  "Food type":"Italian",
  "Cook book":"recipe notebook"
}
```

**Salida:**

```
{
  "Job": {
    "Status": "Submitted",
    "Inputs": [
      {
        "Container": "mp4",
        "FrameRate": "auto",
        "Key": "ETS_example_file.mp4",
        "AspectRatio": "auto",
        "Resolution": "auto",
        "Interlaced": "auto"
      }
    ],
    "Playlists": [],
    "Outputs": [
      {
        "Status": "Submitted",
        "Rotate": "0",
        "PresetId": "1351620000001-100250",
        "Watermarks": [],
        "Key": "webm/ETS_example_file-kindlefirehd.webm",
        "Id": "1"
      }
    ],
    "PipelineId": "3333333333333333-abcde3",
    "OutputKeyPrefix": "recipes/",
    "UserMetadata": {
      "Cook book": "recipe notebook",
      "Food type": "Italian"
    },
    "Output": {
      "Status": "Submitted",
      "Rotate": "0",
      "PresetId": "1351620000001-100250",
      "Watermarks": [],
      "Key": "webm/ETS_example_file-kindlefirehd.webm",
```

```

        "Id": "1"
    },
    "Timing": {
        "SubmitTimeMillis": 1533838012298
    },
    "Input": {
        "Container": "mp4",
        "FrameRate": "auto",
        "Key": "ETS_example_file.mp4",
        "AspectRatio": "auto",
        "Resolution": "auto",
        "Interlaced": "auto"
    },
    "Id": "1533838012294-example",
    "Arn": "arn:aws:elastictranscoder:us-west-2:123456789012:job/1533838012294-
example"
}
}

```

- Para obtener API más información, consulte la Referencia de comandos [CreateJob](#).AWS CLI

## create-pipeline

En el siguiente ejemplo de código, se muestra cómo usar create-pipeline.

### AWS CLI

Para crear una canalización para ElasticTranscoder

En el siguiente create-pipeline ejemplo se crea una canalización para ElasticTranscoder.

```

aws elastictranscoder create-pipeline \
  --name Default \
  --input-bucket salesoffice.example.com-source \
  --role arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role \
  --notifications Progressing="",Completed="",Warning="",Error=arn:aws:sns:us-
east-1:111222333444:ETS_Errors \
  --content-config file://content-config.json \
  --thumbnail-config file://thumbnail-config.json

```

Contenidos de content-config.json:

```
{
```

```
"Bucket":"salesoffice.example.com-public-promos",
"Permissions":[
  {
    "GranteeType":"Email",
    "Grantee":"marketing-promos@example.com",
    "Access":[
      "FullControl"
    ]
  }
],
"StorageClass":"Standard"
}
```

Contenidos de thumbnail-config.json:

```
{
  "Bucket":"salesoffice.example.com-public-promos-thumbnails",
  "Permissions":[
    {
      "GranteeType":"Email",
      "Grantee":"marketing-promos@example.com",
      "Access":[
        "FullControl"
      ]
    }
  ],
  "StorageClass":"ReducedRedundancy"
}
```

Salida:

```
{
  "Pipeline": {
    "Status": "Active",
    "ContentConfig": {
      "Bucket": "salesoffice.example.com-public-promos",
      "StorageClass": "Standard",
      "Permissions": [
        {
          "Access": [
            "FullControl"
          ],
          "Grantee": "marketing-promos@example.com",

```

```
        "GranteeType": "Email"
      }
    ]
  },
  "Name": "Default",
  "ThumbnailConfig": {
    "Bucket": "salesoffice.example.com-public-promos-thumbnails",
    "StorageClass": "ReducedRedundancy",
    "Permissions": [
      {
        "Access": [
          "FullControl"
        ],
        "Grantee": "marketing-promos@example.com",
        "GranteeType": "Email"
      }
    ]
  },
  "Notifications": {
    "Completed": "",
    "Warning": "",
    "Progressing": "",
    "Error": "arn:aws:sns:us-east-1:123456789012:ETS_Errors"
  },
  "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
  "InputBucket": "salesoffice.example.com-source",
  "Id": "1533765810590-example",
  "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/1533765810590-example"
},
"Warnings": [
  {
    "Message": "The SNS notification topic for Error events and the pipeline
are in different regions, which increases processing time for jobs in the pipeline
and can incur additional charges. To decrease processing time and prevent cross-
regional charges, use the same region for the SNS notification topic and the
pipeline.",
    "Code": "6006"
  }
]
}
```

- Para API obtener más información, consulte [CreatePipeline](#) la Referencia de AWS CLI comandos.

## create-preset

En el siguiente ejemplo de código, se muestra cómo usar create-preset.

### AWS CLI

Para crear un ajuste preestablecido para ElasticTranscoder

En el create-preset ejemplo siguiente se crea un ajuste preestablecido para ElasticTranscoder.

```
aws elastictranscoder create-preset \  
  --name DefaultPreset \  
  --description "Use for published videos" \  
  --container mp4 \  
  --video file://video.json \  
  --audio file://audio.json \  
  --thumbnails file://thumbnails.json
```

Contenidos de video.json:

```
{  
  "Codec": "H.264",  
  "CodecOptions": {  
    "Profile": "main",  
    "Level": "2.2",  
    "MaxReferenceFrames": "3",  
    "MaxBitRate": "",  
    "BufferSize": "",  
    "InterlacedMode": "Progressive",  
    "ColorSpaceConversionMode": "None"  
  },  
  "KeyframesMaxDist": "240",  
  "FixedGOP": "false",  
  "BitRate": "1600",  
  "FrameRate": "auto",  
  "MaxFrameRate": "30",  
  "MaxWidth": "auto",  
  "MaxHeight": "auto",
```

```
"SizingPolicy":"Fit",
"PaddingPolicy":"Pad",
"DisplayAspectRatio":"auto",
"Watermarks":[
  {
    "Id":"company logo",
    "MaxWidth":"20%",
    "MaxHeight":"20%",
    "SizingPolicy":"ShrinkToFit",
    "HorizontalAlign":"Right",
    "HorizontalOffset":"10px",
    "VerticalAlign":"Bottom",
    "VerticalOffset":"10px",
    "Opacity":"55.5",
    "Target":"Content"
  }
]
```

Contenidos de audio.json:

```
{
  "Codec":"AAC",
  "CodecOptions":{"
    "Profile":"AAC-LC"
  }},
  "SampleRate":"44100",
  "BitRate":"96",
  "Channels":"2"
}
```

Contenidos de thumbnails.json:

```
{
  "Format":"png",
  "Interval":"120",
  "MaxWidth":"auto",
  "MaxHeight":"auto",
  "SizingPolicy":"Fit",
  "PaddingPolicy":"Pad"
}
```

Salida:



```
{
  "Preset": {
    "Thumbnails": {
      "SizingPolicy": "Fit",
      "MaxWidth": "auto",
      "Format": "png",
      "PaddingPolicy": "Pad",
      "Interval": "120",
      "MaxHeight": "auto"
    },
    "Container": "mp4",
    "Description": "Use for published videos",
    "Video": {
      "SizingPolicy": "Fit",
      "MaxWidth": "auto",
      "PaddingPolicy": "Pad",
      "MaxFrameRate": "30",
      "FrameRate": "auto",
      "MaxHeight": "auto",
      "KeyframesMaxDist": "240",
      "FixedGOP": "false",
      "Codec": "H.264",
      "Watermarks": [
        {
          "SizingPolicy": "ShrinkToFit",
          "VerticalOffset": "10px",
          "VerticalAlign": "Bottom",
          "Target": "Content",
          "MaxWidth": "20%",
          "MaxHeight": "20%",
          "HorizontalAlign": "Right",
          "HorizontalOffset": "10px",
          "Opacity": "55.5",
          "Id": "company logo"
        }
      ],
      "CodecOptions": {
        "Profile": "main",
        "MaxBitRate": "32",
        "InterlacedMode": "Progressive",
        "Level": "2.2",
        "ColorSpaceConversionMode": "None",
        "MaxReferenceFrames": "3",
```

```

        "BufferSize": "5"
      },
      "BitRate": "1600",
      "DisplayAspectRatio": "auto"
    },
    "Audio": {
      "Channels": "2",
      "CodecOptions": {
        "Profile": "AAC-LC"
      },
      "SampleRate": "44100",
      "Codec": "AAC",
      "BitRate": "96"
    },
    "Type": "Custom",
    "Id": "1533765290724-example"
    "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:preset/1533765290724-example",
    "Name": "DefaultPreset"
  },
  "Warning": ""
}

```

- Para API obtener más información, consulte [CreatePreset](#) la Referencia de AWS CLI comandos.

## delete-pipeline

En el siguiente ejemplo de código, se muestra cómo usar delete-pipeline.

### AWS CLI

Para eliminar la ElasticTranscoder canalización especificada

Esto elimina la ElasticTranscoder canalización especificada.

Comando:

```
aws elastictranscoder delete-pipeline --id 111111111111-abcde1
```

Salida:

```
{
```

```
"Success": "true"  
}
```

- Para API obtener más información, consulte [DeletePipeline](#) la Referencia de AWS CLI comandos.

## **delete-preset**

En el siguiente ejemplo de código, se muestra cómo usar `delete-preset`.

### AWS CLI

Para eliminar el ElasticTranscoder ajuste preestablecido especificado

Esto elimina el ElasticTranscoder ajuste preestablecido especificado.

Comando:

```
aws elastictranscoder delete-preset --id 555555555555-abcde5
```

- Para API obtener más información, consulte [DeletePreset](#) la Referencia de AWS CLI comandos.

## **list-jobs-by-pipeline**

En el siguiente ejemplo de código, se muestra cómo usar `list-jobs-by-pipeline`.

### AWS CLI

Para recuperar una lista de ElasticTranscoder trabajos en la canalización especificada

En este ejemplo, se recupera una lista de ElasticTranscoder trabajos de la canalización especificada.

Comando:

```
aws elastictranscoder list-jobs-by-pipeline --pipeline-id 111111111111-abcde1
```

Salida:

```
{
```

```
"Jobs": []  
}
```

- Para API obtener más información, consulte [ListJobsByPipeline](#) la Referencia de AWS CLI comandos.

## **list-jobs-by-status**

En el siguiente ejemplo de código, se muestra cómo usar `list-jobs-by-status`.

### AWS CLI

Para recuperar una lista de ElasticTranscoder trabajos con el estado Completo

En este ejemplo, se recupera una lista de ElasticTranscoder trabajos con el estado Completo.

Comando:

```
aws elastictranscoder list-jobs-by-status --status Complete
```

Salida:

```
{  
  "Jobs": []  
}
```

- Para API obtener más información, consulte [ListJobsByStatus](#) la Referencia de AWS CLI comandos.

## **list-pipelines**

En el siguiente ejemplo de código, se muestra cómo usar `list-pipelines`.

### AWS CLI

Para recuperar una lista de ElasticTranscoder canalizaciones

En este ejemplo, se recupera una lista de ElasticTranscoder canalizaciones.

Comando:

**aws elastictranscoder list-pipelines**

Salida:

```
{
  "Pipelines": [
    {
      "Status": "Active",
      "ContentConfig": {
        "Bucket": "ets-example",
        "Permissions": []
      },
      "Name": "example-pipeline",
      "ThumbnailConfig": {
        "Bucket": "ets-example",
        "Permissions": []
      },
      "Notifications": {
        "Completed": "arn:aws:sns:us-west-2:123456789012:ets_example",
        "Warning": "",
        "Progressing": "",
        "Error": ""
      },
      "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
      "InputBucket": "ets-example",
      "OutputBucket": "ets-example",
      "Id": "333333333333-abcde3",
      "Arn": "arn:aws:elastictranscoder:us-west-2:123456789012:pipeline/333333333333-abcde3"
    },
    {
      "Status": "Paused",
      "ContentConfig": {
        "Bucket": "ets-example",
        "Permissions": []
      },
      "Name": "example-php-test",
      "ThumbnailConfig": {
        "Bucket": "ets-example",
        "Permissions": []
      },
      "Notifications": {
        "Completed": "",
```

```

        "Warning": "",
        "Progressing": "",
        "Error": ""
    },
    "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
    "InputBucket": "ets-example",
    "OutputBucket": "ets-example",
    "Id": "333333333333-abcde2",
    "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/333333333333-abcde2"
},
{
    "Status": "Active",
    "ContentConfig": {
        "Bucket": "ets-west-output",
        "Permissions": []
    },
},
    "Name": "pipeline-west",
    "ThumbnailConfig": {
        "Bucket": "ets-west-output",
        "Permissions": []
    },
},
    "Notifications": {
        "Completed": "arn:aws:sns:us-west-2:123456789012:ets-notifications",
        "Warning": "",
        "Progressing": "",
        "Error": ""
    },
    "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
    "InputBucket": "ets-west-input",
    "OutputBucket": "ets-west-output",
    "Id": "333333333333-abcde1",
    "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/333333333333-abcde1"
}
]
}

```

- Para API obtener más información, consulte la Referencia [ListPipelines](#) de AWS CLI comandos.

## list-presets

En el siguiente ejemplo de código, se muestra cómo usar `list-presets`.

## AWS CLI

Para recuperar una lista de ajustes ElasticTranscoder preestablecidos

En este ejemplo se recupera una lista de ElasticTranscoder ajustes preestablecidos.

Comando:

```
aws elastictranscoder list-presets --max-items 2
```

Salida:

```
{
  "Presets": [
    {
      "Container": "mp4",
      "Name": "KindleFireHD-preset",
      "Video": {
        "Resolution": "1280x720",
        "FrameRate": "30",
        "KeyframesMaxDist": "90",
        "FixedGOP": "false",
        "Codec": "H.264",
        "Watermarks": [],
        "CodecOptions": {
          "Profile": "main",
          "MaxReferenceFrames": "3",
          "ColorSpaceConversionMode": "None",
          "InterlacedMode": "Progressive",
          "Level": "4"
        },
        "AspectRatio": "16:9",
        "BitRate": "2200"
      },
      "Audio": {
        "Channels": "2",
        "CodecOptions": {
          "Profile": "AAC-LC"
        },
        "SampleRate": "48000",
        "Codec": "AAC",
        "BitRate": "160"
      },
      "Type": "Custom",
    }
  ]
}
```

```
    "Id": "333333333333-abcde2",
    "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:preset/333333333333-abcde2",
    "Thumbnails": {
      "AspectRatio": "16:9",
      "Interval": "60",
      "Resolution": "192x108",
      "Format": "png"
    }
  },
  {
    "Thumbnails": {
      "AspectRatio": "16:9",
      "Interval": "60",
      "Resolution": "192x108",
      "Format": "png"
    },
    "Container": "mp4",
    "Description": "Custom preset for transcoding jobs",
    "Video": {
      "Resolution": "1280x720",
      "FrameRate": "30",
      "KeyframesMaxDist": "90",
      "FixedGOP": "false",
      "Codec": "H.264",
      "Watermarks": [],
      "CodecOptions": {
        "Profile": "main",
        "MaxReferenceFrames": "3",
        "ColorSpaceConversionMode": "None",
        "InterlacedMode": "Progressive",
        "Level": "3.1"
      },
      "AspectRatio": "16:9",
      "BitRate": "2200"
    },
    "Audio": {
      "Channels": "2",
      "CodecOptions": {
        "Profile": "AAC-LC"
      },
      "SampleRate": "44100",
      "Codec": "AAC",
      "BitRate": "160"
    }
  }
}
```



```

    },
    "Type": "Custom",
    "Id": "333333333333-abcde3",
    "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:preset/333333333333-abcde3",
    "Name": "Roman's Preset"
  }
],
"NextToken": "eyJQYWdlVG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}

```

- Para API obtener más información, consulte la Referencia [ListPresets](#) de AWS CLI comandos.

## read-job

En el siguiente ejemplo de código, se muestra cómo usar `read-job`.

### AWS CLI

Para recuperar un ElasticTranscoder trabajo

En este ejemplo se recupera el ElasticTranscoder trabajo especificado.

Comando:

```
aws elastictranscoder read-job --id 1533838012294-example
```

Salida:

```

{
  "Job": {
    "Status": "Progressing",
    "Inputs": [
      {
        "Container": "mp4",
        "FrameRate": "auto",
        "Key": "ETS_example_file.mp4",
        "AspectRatio": "auto",
        "Resolution": "auto",
        "Interlaced": "auto"
      }
    ]
  },

```

```
"Playlists": [],
"Outputs": [
  {
    "Status": "Progressing",
    "Rotate": "0",
    "PresetId": "1351620000001-100250",
    "Watermarks": [],
    "Key": "webm/ETS_example_file-kindlefirehd.webm",
    "Id": "1"
  }
],
"PipelineId": "3333333333333-abcde3",
"OutputKeyPrefix": "recipes/",
"UserMetadata": {
  "Cook book": "recipe notebook",
  "Food type": "Italian"
},
"Output": {
  "Status": "Progressing",
  "Rotate": "0",
  "PresetId": "1351620000001-100250",
  "Watermarks": [],
  "Key": "webm/ETS_example_file-kindlefirehd.webm",
  "Id": "1"
},
"Timing": {
  "SubmitTimeMillis": 1533838012298,
  "StartTimeMillis": 1533838013786
},
"Input": {
  "Container": "mp4",
  "FrameRate": "auto",
  "Key": "ETS_example_file.mp4",
  "AspectRatio": "auto",
  "Resolution": "auto",
  "Interlaced": "auto"
},
"Id": "1533838012294-example",
"Arn": "arn:aws:elastictranscoder:us-west-2:123456789012:job/1533838012294-
example"
}
}
```

- Para API obtener más información, consulte [ReadJob](#) la Referencia de AWS CLI comandos.

## read-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `read-pipeline`.

### AWS CLI

Para recuperar una ElasticTranscoder canalización

En este ejemplo, se recupera la ElasticTranscoder canalización especificada.

Comando:

```
aws elastictranscoder read-pipeline --id 333333333333-abcde3
```

Salida:

```
{
  "Pipeline": {
    "Status": "Active",
    "ContentConfig": {
      "Bucket": "ets-example",
      "StorageClass": "Standard",
      "Permissions": [
        {
          "Access": [
            "FullControl"
          ],
          "Grantee": "marketing-promos@example.com",
          "GranteeType": "Email"
        }
      ]
    },
    "Name": "Default",
    "ThumbnailConfig": {
      "Bucket": "ets-example",
      "StorageClass": "ReducedRedundancy",
      "Permissions": [
        {
          "Access": [
            "FullControl"
          ],
          "Grantee": "marketing-promos@example.com",
          "GranteeType": "Email"
        }
      ]
    }
  }
}
```

```

    ]
  },
  "Notifications": {
    "Completed": "",
    "Warning": "",
    "Progressing": "",
    "Error": "arn:aws:sns:us-east-1:123456789012:ETS_Errors"
  },
  "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
  "InputBucket": "ets-example",
  "Id": "3333333333333-abcde3",
  "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/3333333333333-abcde3"
  },
  "Warnings": [
    {
      "Message": "The SNS notification topic for Error events and the pipeline
are in different regions, which increases processing time for jobs in the pipeline
and can incur additional charges. To decrease processing time and prevent cross-
regional charges, use the same region for the SNS notification topic and the
pipeline.",
      "Code": "6006"
    }
  ]
}

```

- Para API obtener más información, consulte [ReadPipeline](#) la Referencia de AWS CLI comandos.

## read-preset

En el siguiente ejemplo de código, se muestra cómo usar `read-preset`.

### AWS CLI

Para recuperar un ElasticTranscoder ajuste preestablecido

En este ejemplo se recupera el ElasticTranscoder ajuste preestablecido especificado.

Comando:

```
aws elastictranscoder read-preset --id 1351620000001-500020
```

**Salida:**

```
{
  "Preset": {
    "Thumbnails": {
      "SizingPolicy": "ShrinkToFit",
      "MaxWidth": "192",
      "Format": "png",
      "PaddingPolicy": "NoPad",
      "Interval": "300",
      "MaxHeight": "108"
    },
    "Container": "fmp4",
    "Description": "System preset: MPEG-Dash Video - 4.8M",
    "Video": {
      "SizingPolicy": "ShrinkToFit",
      "MaxWidth": "1280",
      "PaddingPolicy": "NoPad",
      "FrameRate": "30",
      "MaxHeight": "720",
      "KeyframesMaxDist": "60",
      "FixedGOP": "true",
      "Codec": "H.264",
      "Watermarks": [
        {
          "SizingPolicy": "ShrinkToFit",
          "VerticalOffset": "10%",
          "VerticalAlign": "Top",
          "Target": "Content",
          "MaxWidth": "10%",
          "MaxHeight": "10%",
          "HorizontalAlign": "Left",
          "HorizontalOffset": "10%",
          "Opacity": "100",
          "Id": "TopLeft"
        },
        {
          "SizingPolicy": "ShrinkToFit",
          "VerticalOffset": "10%",
          "VerticalAlign": "Top",
          "Target": "Content",
          "MaxWidth": "10%",
          "MaxHeight": "10%",
          "HorizontalAlign": "Right",
```

```
        "HorizontalOffset": "10%",
        "Opacity": "100",
        "Id": "TopRight"
    },
    {
        "SizingPolicy": "ShrinkToFit",
        "VerticalOffset": "10%",
        "VerticalAlign": "Bottom",
        "Target": "Content",
        "MaxWidth": "10%",
        "MaxHeight": "10%",
        "HorizontalAlign": "Left",
        "HorizontalOffset": "10%",
        "Opacity": "100",
        "Id": "BottomLeft"
    },
    {
        "SizingPolicy": "ShrinkToFit",
        "VerticalOffset": "10%",
        "VerticalAlign": "Bottom",
        "Target": "Content",
        "MaxWidth": "10%",
        "MaxHeight": "10%",
        "HorizontalAlign": "Right",
        "HorizontalOffset": "10%",
        "Opacity": "100",
        "Id": "BottomRight"
    }
],
"CodecOptions": {
    "Profile": "main",
    "MaxBitRate": "4800",
    "InterlacedMode": "Progressive",
    "Level": "3.1",
    "ColorSpaceConversionMode": "None",
    "MaxReferenceFrames": "3",
    "BufferSize": "9600"
},
"BitRate": "4800",
"DisplayAspectRatio": "auto"
},
"Type": "System",
"Id": "1351620000001-500020",
```

```

    "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:preset/1351620000001-500020",
    "Name": "System preset: MPEG-Dash Video - 4.8M"
  }
}

```

- Para API obtener más información, consulte [ReadPreset](#) la Referencia de AWS CLI comandos.

## update-pipeline-notifications

En el siguiente ejemplo de código, se muestra cómo usar `update-pipeline-notifications`.

### AWS CLI

Para actualizar las notificaciones de una ElasticTranscoder canalización

En este ejemplo, se actualizan las notificaciones de la ElasticTranscoder canalización especificada.

Comando:

```

aws elastictranscoder update-pipeline-notifications --id 111111111111-
abcde1 --notifications Progressing=arn:aws:sns:us-west-2:0123456789012:my-
topic,Completed=arn:aws:sns:us-west-2:0123456789012:my-topic,Warning=arn:aws:sns:us-
west-2:0123456789012:my-topic,Error=arn:aws:sns:us-east-1:111222333444:ETS_Errors

```

Salida:

```

{
  "Pipeline": {
    "Status": "Active",
    "ContentConfig": {
      "Bucket": "ets-example",
      "StorageClass": "Standard",
      "Permissions": [
        {
          "Access": [
            "FullControl"
          ],
          "Grantee": "marketing-promos@example.com",
          "GranteeType": "Email"
        }
      ]
    }
  }
}

```

```

    }
  ]
},
"Name": "Default",
"ThumbnailConfig": {
  "Bucket": "ets-example",
  "StorageClass": "ReducedRedundancy",
  "Permissions": [
    {
      "Access": [
        "FullControl"
      ],
      "Grantee": "marketing-promos@example.com",
      "GranteeType": "Email"
    }
  ]
},
"Notifications": {
  "Completed": "arn:aws:sns:us-west-2:0123456789012:my-topic",
  "Warning": "arn:aws:sns:us-west-2:0123456789012:my-topic",
  "Progressing": "arn:aws:sns:us-west-2:0123456789012:my-topic",
  "Error": "arn:aws:sns:us-east-1:111222333444:ETS_Errors"
},
"Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
"InputBucket": "ets-example",
"Id": "111111111111-abcde1",
"Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/111111111111-abcde1"
}
}

```

- Para API obtener más información, consulte [UpdatePipelineNotifications](#) la Referencia de AWS CLI comandos.

## update-pipeline-status

En el siguiente ejemplo de código, se muestra cómo usar `update-pipeline-status`.

### AWS CLI

Para actualizar el estado de una ElasticTranscoder canalización

En este ejemplo, se actualiza el estado de la ElasticTranscoder canalización especificada.



## Comando:

```
aws elastictranscoder update-pipeline-status --id 111111111111-abcde1 --  
status Paused
```

## Salida:

```
{  
  "Pipeline": {  
    "Status": "Paused",  
    "ContentConfig": {  
      "Bucket": "ets-example",  
      "StorageClass": "Standard",  
      "Permissions": [  
        {  
          "Access": [  
            "FullControl"  
          ],  
          "Grantee": "marketing-promos@example.com",  
          "GranteeType": "Email"  
        }  
      ]  
    },  
    "Name": "Default",  
    "ThumbnailConfig": {  
      "Bucket": "ets-example",  
      "StorageClass": "ReducedRedundancy",  
      "Permissions": [  
        {  
          "Access": [  
            "FullControl"  
          ],  
          "Grantee": "marketing-promos@example.com",  
          "GranteeType": "Email"  
        }  
      ]  
    },  
    "Notifications": {  
      "Completed": "",  
      "Warning": "",  
      "Progressing": "",  
      "Error": "arn:aws:sns:us-east-1:803981987763:ETS_Errors"  
    }  
  },  
}
```

```

    "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
    "InputBucket": "ets-example",
    "Id": "111111111111-abcde1",
    "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/111111111111-abcde1"
  }
}

```

- Para API obtener más información, consulte [UpdatePipelineStatus](#) la Referencia de AWS CLI comandos.

## update-pipeline

En el siguiente ejemplo de código, se muestra cómo usar update-pipeline.

### AWS CLI

Para actualizar una ElasticTranscoder canalización

En el siguiente update-pipeline ejemplo, se actualiza la ElasticTranscoder canalización especificada.

```

aws elastictranscoder update-pipeline \
  --id 111111111111-abcde1
  --name DefaultExample \
  --input-bucket salesoffice.example.com-source \
  --role arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role \
  --notifications Progressing="",Completed="",Warning="",Error=arn:aws:sns:us-
east-1:111222333444:ETS_Errors \
  --content-config file://content-config.json \
  --thumbnail-config file://thumbnail-config.json

```

Contenidos de content-config.json:

```

{
  "Bucket": "salesoffice.example.com-public-promos",
  "Permissions": [
    {
      "GranteeType": "Email",
      "Grantee": "marketing-promos@example.com",
      "Access": [
        "FullControl"
      ]
    }
  ]
}

```

```
    ]
  }
],
"StorageClass": "Standard"
}
```

### Contenidos de thumbnail-config.json:

```
{
  "Bucket": "salesoffice.example.com-public-promos-thumbnails",
  "Permissions": [
    {
      "GranteeType": "Email",
      "Grantee": "marketing-promos@example.com",
      "Access": [
        "FullControl"
      ]
    }
  ],
  "StorageClass": "ReducedRedundancy"
}
```

### Salida:

```
{
  "Pipeline": {
    "Status": "Active",
    "ContentConfig": {
      "Bucket": "ets-example",
      "StorageClass": "Standard",
      "Permissions": [
        {
          "Access": [
            "FullControl"
          ],
          "Grantee": "marketing-promos@example.com",
          "GranteeType": "Email"
        }
      ]
    }
  },
  "Name": "DefaultExample",
  "ThumbnailConfig": {
    "Bucket": "ets-example",
```

```

    "StorageClass": "ReducedRedundancy",
    "Permissions": [
      {
        "Access": [
          "FullControl"
        ],
        "Grantee": "marketing-promos@example.com",
        "GranteeType": "Email"
      }
    ],
    "Notifications": {
      "Completed": "",
      "Warning": "",
      "Progressing": "",
      "Error": "arn:aws:sns:us-east-1:1112223333444:ETS_Errors"
    },
    "Role": "arn:aws:iam::123456789012:role/Elastic_Transcoder_Default_Role",
    "InputBucket": "ets-example",
    "Id": "333333333333-abcde3",
    "Arn": "arn:aws:elastictranscoder:us-
west-2:123456789012:pipeline/333333333333-abcde3"
  },
  "Warnings": [
    {
      "Message": "The SNS notification topic for Error events and the pipeline
are in different regions, which increases processing time for jobs in the pipeline
and can incur additional charges. To decrease processing time and prevent cross-
regional charges, use the same region for the SNS notification topic and the
pipeline.",
      "Code": "6006"
    }
  ]
}

```

- Para API obtener más información, consulte [UpdatePipeline](#) la Referencia de AWS CLI comandos.

## ElastiCache ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with ElastiCache.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **add-tags-to-resource**

En el siguiente ejemplo de código, se muestra cómo usar `add-tags-to-resource`.

AWS CLI

Para añadir etiquetas a un recurso

En el siguiente `add-tags-to-resource` ejemplo, se agregan hasta 10 etiquetas (pares clave-valor) a un recurso de clúster o instantánea.

```
aws elasticache add-tags-to-resource \  
  --resource-name "arn:aws:elasticache:us-east-1:1234567890:cluster:my-mem-  
cluster" \  
  --tags '{"20150202":15, "ElastiCache":"Service"}'
```

Salida:

```
{  
  "TagList": [  
    {  
      "Value": "20150202",  
      "Key": "APIVersion"  
    },  
    {  
      "Value": "ElastiCache",  
      "Key": "Service"  
    }  
  ]  
}
```

```
]
}
```

Para obtener más información, consulte [Monitorear los costos con etiquetas de asignación](#) de costos en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia [AddTagsToResource](#) de AWS CLI comandos.

## **authorize-cache-security-group-ingress**

En el siguiente ejemplo de código, se muestra cómo usar `authorize-cache-security-group-ingress`.

### AWS CLI

Para autorizar la entrada del grupo de seguridad de caché

El siguiente `authorize-cache-security-group-ingress` ejemplo permite el ingreso de la red a un grupo de seguridad de caché.

```
aws elasticache authorize-cache-security-group-ingress \
  --cache-security-group-name "my-sec-grp" \
  --ec2-security-group-name "my-ec2-sec-grp" \
  --ec2-security-group-owner-id "1234567890"
```

El comando no produce ningún resultado.

Para obtener más información, consulte [Actualizaciones de autoservicio en Amazon ElastiCache en](#) la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia de [AuthorizeCacheSecurityGroupIngress](#) comandos AWS CLI .

## **batch-apply-update-action**

En el siguiente ejemplo de código, se muestra cómo usar `batch-apply-update-action`.

### AWS CLI

Para aplicar una actualización de servicio

El siguiente `batch-apply-update-action` ejemplo aplica una actualización de servicio a un clúster de Redis.

```
aws elasticache batch-apply-update-action \  
  --service-update-name elc-xxxxx406-xxx \  
  --replication-group-ids test-cluster
```

Salida:

```
{  
  "ProcessedUpdateActions": [  
    {  
      "ReplicationGroupId": "pat-cluster",  
      "ServiceUpdateName": "elc-xxxxx406-xxx",  
      "UpdateActionStatus": "waiting-to-start"  
    }  
  ],  
  "UnprocessedUpdateActions": []  
}
```

Para obtener más información, consulte [Actualizaciones de autoservicio en Amazon ElastiCache en](#) la Guía del usuario de Elasticache.

- Para API obtener más información, consulte la Referencia de [BatchApplyUpdateAction](#) comandos AWS CLI .

## **batch-stop-update-action**

En el siguiente ejemplo de código, se muestra cómo usar `batch-stop-update-action`.

AWS CLI

Para detener una actualización de servicio

El siguiente `batch-stop-update-action` ejemplo aplica una actualización de servicio a un clúster de Redis.

```
aws elasticache batch-stop-update-action \  
  --service-update-name elc-xxxxx406-xxx \  
  --replication-group-ids test-cluster
```

Salida:

```
{
  "ProcessedUpdateActions": [
    {
      "ReplicationGroupId": "pat-cluster",
      "ServiceUpdateName": "elc-xxxxxx406-xxx",
      "UpdateActionStatus": "stopping"
    }
  ],
  "UnprocessedUpdateActions": []
}
```

Para obtener más información, consulte [Actualizaciones de autoservicio en Amazon ElastiCache](#) en la Guía del usuario de Elasticache.

- Para API obtener más información, consulte la Referencia de [BatchStopUpdateAction](#) comandos AWS CLI .

## copy-snapshot

En el siguiente ejemplo de código, se muestra cómo usar copy-snapshot.

AWS CLI

Para copiar una instantánea

En el siguiente copy-snapshot ejemplo, se hace una copia de una instantánea existente.

```
aws elasticache copy-snapshot \
  --source-snapshot-name "my-snapshot" \
  --target-snapshot-name "my-snapshot-copy"
```

Salida:

```
{
  "Snapshot": {
    "Engine": "redis",
    "CacheParameterGroupName": "default.redis3.2",
    "VpcId": "vpc-3820329f3",
    "CacheClusterId": "my-redis4",
    "SnapshotRetentionLimit": 7,
  }
}
```



```

    "NumCacheNodes": 1,
    "SnapshotName": "my-snapshot-copy",
    "CacheClusterCreateTime": "2016-12-21T22:24:04.955Z",
    "AutoMinorVersionUpgrade": true,
    "PreferredAvailabilityZone": "us-east-1c",
    "SnapshotStatus": "creating",
    "SnapshotSource": "manual",
    "SnapshotWindow": "07:00-08:00",
    "EngineVersion": "3.2.4",
    "NodeSnapshots": [
      {
        "CacheSize": "3 MB",
        "SnapshotCreateTime": "2016-12-28T07:00:52Z",
        "CacheNodeId": "0001",
        "CacheNodeCreateTime": "2016-12-21T22:24:04.955Z"
      }
    ],
    "CacheSubnetGroupName": "default",
    "Port": 6379,
    "PreferredMaintenanceWindow": "tue:09:30-tue:10:30",
    "CacheNodeType": "cache.m3.large"
  }
}

```

Para obtener más información, consulte [Exportación de un Backup](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia [CopySnapshot](#) de AWS CLI comandos.

## create-cache-cluster

En el siguiente ejemplo de código, se muestra cómo usar `create-cache-cluster`.

### AWS CLI

Para crear un clúster de caché

En el siguiente `create-cache-cluster` ejemplo, se crea un clúster de caché mediante el motor Redis.

```

aws elasticache create-cache-cluster \
  --cache-cluster-id "cluster-test" \

```

```
--engine redis \  
--cache-node-type cache.m5.large \  
--num-cache-nodes 1
```

Salida:

```
{  
  "CacheCluster": {  
    "CacheClusterId": "cluster-test",  
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/  
home#client-download:",  
    "CacheNodeType": "cache.m5.large",  
    "Engine": "redis",  
    "EngineVersion": "5.0.5",  
    "CacheClusterStatus": "creating",  
    "NumCacheNodes": 1,  
    "PreferredMaintenanceWindow": "sat:13:00-sat:14:00",  
    "PendingModifiedValues": {},  
    "CacheSecurityGroups": [],  
    "CacheParameterGroup": {  
      "CacheParameterGroupName": "default.redis5.0",  
      "ParameterApplyStatus": "in-sync",  
      "CacheNodeIdsToReboot": []  
    },  
    "CacheSubnetGroupName": "default",  
    "AutoMinorVersionUpgrade": true,  
    "SnapshotRetentionLimit": 0,  
    "SnapshotWindow": "06:30-07:30",  
    "TransitEncryptionEnabled": false,  
    "AtRestEncryptionEnabled": false  
  }  
}
```

Para obtener más información, consulte [Creación de un clúster](#) en la Guía del usuario de Elasticache.

- Para API obtener más información, consulte la Referencia [CreateCacheCluster](#) de AWS CLI comandos.

## **create-cache-parameter-group**

En el siguiente ejemplo de código, se muestra cómo usar `create-cache-parameter-group`.

## AWS CLI

Para crear un grupo de parámetros de caché

En el siguiente `create-cache-parameter-group` ejemplo, se crea un nuevo grupo de parámetros de ElastiCache caché de Amazon.

```
aws elasticache create-cache-parameter-group \  
  --cache-parameter-group-family "redis5.0" \  
  --cache-parameter-group-name "mygroup" \  
  --description "mygroup"
```

Salida:

```
{  
  "CacheParameterGroup": {  
    "CacheParameterGroupName": "mygroup",  
    "CacheParameterGroupFamily": "redis5.0",  
    "Description": "my group"  
  }  
}
```

Para obtener más información, consulte [Creación de un grupo de parámetros](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia [CreateCacheParameterGroup](#) de AWS CLI comandos.

## **create-cache-subnet-group**

En el siguiente ejemplo de código, se muestra cómo usar `create-cache-subnet-group`.

## AWS CLI

Para crear un grupo de subredes de caché

En el siguiente `create-cache-subnet-group` ejemplo, se crea un nuevo grupo de subredes de caché.

```
aws elasticache create-cache-subnet-group \  
  --cache-subnet-group-name "mygroup" \  
  --cache-subnet-group-description "my subnet group" \  
  --vpc-id vpc-12345678
```

```
--subnet-ids "subnet-xxxxec4f"
```

Salida:

```
{
  "CacheSubnetGroup": {
    "CacheSubnetGroupName": "mygroup",
    "CacheSubnetGroupDescription": "my subnet group",
    "VpcId": "vpc-a3e97cdb",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-xxxxec4f",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2d"
        }
      }
    ]
  }
}
```

Para obtener más información, consulte [Crear un grupo de subredes de caché](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia de [CreateCacheSubnetGroup](#) comandos AWS CLI .

## create-global-replication-group

En el siguiente ejemplo de código, se muestra cómo usar `create-global-replication-group`.

AWS CLI

Para crear un grupo de replicación global

El siguiente `create-global-replication-group` ejemplo crea un nuevo grupo de replicación global.

```
aws elasticache create-global-replication-group \
  --global-replication-group-id-suffix my-global-replication-group \
  --primary-replication-group-id my-primary-cluster
```

Salida:

```
{
  "GlobalReplicationGroup": {
    "GlobalReplicationGroupId": "sgaui-my-global-replication-group",
    "GlobalReplicationGroupDescription": " ",
    "Status": "creating",
    "CacheNodeType": "cache.r5.large",
    "Engine": "redis",
    "EngineVersion": "5.0.6",
    "Members": [
      {
        "ReplicationGroupId": "my-primary-cluster",
        "ReplicationGroupRegion": "us-west-2",
        "Role": "PRIMARY",
        "AutomaticFailover": "enabled",
        "Status": "associating"
      }
    ],
    "ClusterEnabled": true,
    "GlobalNodeGroups": [
      {
        "GlobalNodeGroupId": "sgaui-my-global-replication-group-0001",
        "Slots": "0-16383"
      }
    ],
    "AuthTokenEnabled": false,
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false
  }
}
```

Para obtener más información, consulte [Replicación entre AWS regiones mediante un almacén de datos global en la Guía del usuario](#) de Elasticache.

- Para obtener API más información, consulte la Referencia de comandos [CreateGlobalReplicationGroup](#).AWS CLI

## **create-replication-group**

En el siguiente ejemplo de código, se muestra cómo usar `create-replication-group`.

### AWS CLI

Para crear un grupo de replicación

En el siguiente `create-replication-group` ejemplo, se crea un grupo de replicación de Redis (modo de clúster desactivado) o de Redis (modo de clúster activado). Esta operación solo es válida para Redis.

```
aws elasticache create-replication-group \  
  --replication-group-id "mygroup" \  
  --replication-group-description "my group" \  
  --engine "redis" \  
  --cache-node-type "cache.m5.large"
```

Salida:

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "mygroup",  
    "Description": "my group",  
    "Status": "creating",  
    "PendingModifiedValues": {},  
    "MemberClusters": [  
      "mygroup-001"  
    ],  
    "AutomaticFailover": "disabled",  
    "SnapshotRetentionLimit": 0,  
    "SnapshotWindow": "06:00-07:00",  
    "ClusterEnabled": false,  
    "CacheNodeType": "cache.m5.large",  
    "TransitEncryptionEnabled": false,  
    "AtRestEncryptionEnabled": false  
  }  
}
```

Para obtener más información, consulte [Creación de un grupo de replicación de Redis](#) en la Guía del usuario de Elasticache.

- Para API obtener más información, consulte la Referencia de [CreateReplicationGroup](#) comandos AWS CLI .

## create-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `create-snapshot`.

## AWS CLI

Para crear una instantánea

En el siguiente `create-snapshot` ejemplo, se crea una instantánea mediante el motor Redis.

```
aws elasticache create-snapshot \  
  --snapshot-name mynsnapshot \  
  --cache-cluster-id cluster-test
```

Salida:

```
{  
  "Snapshot": {  
    "SnapshotName": "mynsnapshot",  
    "CacheClusterId": "cluster-test",  
    "SnapshotStatus": "creating",  
    "SnapshotSource": "manual",  
    "CacheNodeType": "cache.m5.large",  
    "Engine": "redis",  
    "EngineVersion": "5.0.5",  
    "NumCacheNodes": 1,  
    "PreferredAvailabilityZone": "us-west-2b",  
    "CacheClusterCreateTime": "2020-03-19T03:12:01.483Z",  
    "PreferredMaintenanceWindow": "sat:13:00-sat:14:00",  
    "Port": 6379,  
    "CacheParameterGroupName": "default.redis5.0",  
    "CacheSubnetGroupName": "default",  
    "VpcId": "vpc-a3e97cdb",  
    "AutoMinorVersionUpgrade": true,  
    "SnapshotRetentionLimit": 0,  
    "SnapshotWindow": "06:30-07:30",  
    "NodeSnapshots": [  
      {  
        "CacheNodeId": "0001",  
        "CacheSize": "",  
        "CacheNodeCreateTime": "2020-03-19T03:12:01.483Z"  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [Backup and Restore ElastiCache for Redis](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia de [CreateSnapshot](#) comandos AWS CLI .

## create-user-group

En el siguiente ejemplo de código, se muestra cómo usar `create-user-group`.

### AWS CLI

Para crear un grupo de usuarios

En el siguiente `create-user-group` ejemplo, se crea un grupo de usuarios nuevo.

```
aws elasticache create-user-group \  
  --user-group-id myusergroup \  
  --engine redis \  
  --user-ids default
```

Salida:

```
{  
  "UserGroupId": "myusergroup",  
  "Status": "creating",  
  "Engine": "redis",  
  "UserIds": [  
    "default"  
  ],  
  "ReplicationGroups": [],  
  "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:usergroup:myusergroup"  
}
```

Para obtener más información, consulte [Autenticación de usuarios con el control de acceso basado en roles \(RBAC\)](#) en la Guía del usuario de ElastiCache.

- Para obtener API más información, consulte la Referencia de comandos. [CreateUserGroup](#) AWS CLI



## create-user

En el siguiente ejemplo de código, se muestra cómo usar `create-user`.

### AWS CLI

Para crear un usuario

En el `create-user` ejemplo siguiente se crea un usuario nuevo.

```
aws elasticache create-user \  
  --user-id user1 \  
  --user-name myUser \  
  --passwords mYnuUzrpAxXw2rdzx \  
  --engine redis \  
  --access-string "on ~app:* -@all +@read"
```

Salida:

```
{  
  "UserId": "user2",  
  "UserName": "myUser",  
  "Status": "active",  
  "Engine": "redis",  
  "AccessString": "on ~app:* -@all +@read +@hash +@bitmap +@geo -setbit -bitfield  
-hset -hsetnx -hmset -hincrby -hincrbyfloat -hdel -bitop -geoadd -georadius -  
georadiusbymember",  
  "UserGroupIds": [],  
  "Authentication": {  
    "Type": "password",  
    "PasswordCount": 1  
  },  
  "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxxx52:user:user2"  
}
```

Para obtener más información, consulte [Autenticación de usuarios con el control de acceso basado en roles \(RBAC\)](#) en la Guía del usuario de Elasticache.

- Para obtener API más información, consulte la Referencia de comandos. [CreateUser](#)AWS CLI

## decrease-node-groups-in-global-replication-group

En el siguiente ejemplo de código, se muestra cómo usar `decrease-node-groups-in-global-replication-group`.

### AWS CLI

Para reducir el número de grupos de nodos en un grupo de replicación global

Lo siguiente `decrease-node-groups-in-global-replication-group` reduce el recuento de grupos de nodos mediante el motor Redis.

```
aws elasticache decrease-node-groups-in-global-replication-group \  
  --global-replication-group-id sgaui-test \  
  --node-group-count 1 \  
  --apply-immediately \  
  --global-node-groups-to-retain sgaui-test-0003
```

Salida:

```
{  
  "GlobalReplicationGroup":  
  {  
    "GlobalReplicationGroupId": "sgaui-test",  
    "GlobalReplicationGroupDescription": "test",  
    "Status": "modifying",  
    "CacheNodeType": "cache.r5.large",  
    "Engine": "redis",  
    "EngineVersion": "5.0.6",  
    "Members": [  
      {  
        "ReplicationGroupId": "test-2",  
        "ReplicationGroupRegion": "us-east-1",  
        "Role": "SECONDARY",  
        "AutomaticFailover": "enabled",  
        "Status": "associated"  
      },  
      {  
        "ReplicationGroupId": "test-1",  
        "ReplicationGroupRegion": "us-west-2",  
        "Role": "PRIMARY",  
        "AutomaticFailover": "enabled",  
        "Status": "associated"  
      }  
    ]  
  }  
}
```

```
    }
  ],
  "ClusterEnabled": true,
  "GlobalNodeGroups": [
    {
      "GlobalNodeGroupId": "sgaui-test-0001",
      "Slots": "0-449,1816-5461"
    },
    {
      "GlobalNodeGroupId": "sgaui-test-0002",
      "Slots": "6827-10922"
    },
    {
      "GlobalNodeGroupId": "sgaui-test-0003",
      "Slots": "10923-14052,15418-16383"
    },
    {
      "GlobalNodeGroupId": "sgaui-test-0004",
      "Slots": "450-1815,5462-6826,14053-15417"
    }
  ],
  "AuthTokenEnabled": false,
  "TransitEncryptionEnabled": false,
  "AtRestEncryptionEnabled": false
}
}
```

Para obtener más información, consulte [Replicación entre AWS regiones mediante un almacén de datos global en la Guía del usuario](#) de ElastiCache.

- Para obtener API más información, consulte la Referencia de comandos [DecreaseNodeGroupsInGlobalReplicationGroup](#).AWS CLI

## decrease-replica-count

En el siguiente ejemplo de código, se muestra cómo usar `decrease-replica-count`.

### AWS CLI

Para reducir el número de réplicas

El siguiente `decrease-replica-count` ejemplo reduce dinámicamente el número de réplicas en un grupo de replicación de Redis (modo de clúster desactivado) o el número de nodos de

réplica en uno o más grupos de nodos (fragmentos) de un grupo de replicación de Redis (modo de clúster activado). Esta operación se realiza sin tiempo de inactividad del clúster.

```
aws elasticache decrease-replica-count \  
  --replication-group-id my-cluster \  
  --apply-immediately \  
  --new-replica-count 2
```

Salida:

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "my-cluster",  
    "Description": " ",  
    "Status": "modifying",  
    "PendingModifiedValues": {},  
    "MemberClusters": [  
      "myrepliac",  
      "my-cluster-001",  
      "my-cluster-002",  
      "my-cluster-003"  
    ],  
    "NodeGroups": [  
      {  
        "NodeGroupId": "0001",  
        "Status": "modifying",  
        "PrimaryEndpoint": {  
          "Address": "my-cluster.xxxxx.ng.0001.usw2.cache.amazonaws.com",  
          "Port": 6379  
        },  
        "ReaderEndpoint": {  
          "Address": "my-cluster-  
ro.xxxxx.ng.0001.usw2.cache.amazonaws.com",  
          "Port": 6379  
        },  
        "NodeGroupMembers": [  
          {  
            "CacheClusterId": "myrepliac",  
            "CacheNodeId": "0001",  
            "ReadEndpoint": {  
              "Address":  
"myrepliac.xxxxx.0001.usw2.cache.amazonaws.com",  
              "Port": 6379  
            }  
          }  
        ]  
      }  
    ]  
  }  
}
```

```

    },
    "PreferredAvailabilityZone": "us-west-2a",
    "CurrentRole": "replica"
  },
  {
    "CacheClusterId": "my-cluster-001",
    "CacheNodeId": "0001",
    "ReadEndpoint": {
      "Address": "my-
cluster-001.xxxxx.0001.usw2.cache.amazonaws.com",
      "Port": 6379
    },
    "PreferredAvailabilityZone": "us-west-2a",
    "CurrentRole": "primary"
  },
  {
    "CacheClusterId": "my-cluster-002",
    "CacheNodeId": "0001",
    "ReadEndpoint": {
      "Address": "my-
cluster-002.xxxxx.0001.usw2.cache.amazonaws.com",
      "Port": 6379
    },
    "PreferredAvailabilityZone": "us-west-2a",
    "CurrentRole": "replica"
  },
  {
    "CacheClusterId": "my-cluster-003",
    "CacheNodeId": "0001",
    "ReadEndpoint": {
      "Address": "my-
cluster-003.xxxxx.0001.usw2.cache.amazonaws.com",
      "Port": 6379
    },
    "PreferredAvailabilityZone": "us-west-2a",
    "CurrentRole": "replica"
  }
]
}
],
"AutomaticFailover": "disabled",
"SnapshotRetentionLimit": 0,
"SnapshotWindow": "07:30-08:30",
"ClusterEnabled": false,

```

```
    "CacheNodeType": "cache.r5.xlarge",
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false
  }
}
```

Para obtener más información, consulte [Cambiar el número de réplicas](#) en la Guía del usuario de Elasticache.

- Para API obtener más información, consulte la Referencia de [DecreaseReplicaCount](#) comandos AWS CLI .

## **delete-cache-cluster**

En el siguiente ejemplo de código, se muestra cómo usar `delete-cache-cluster`.

### AWS CLI

Para eliminar un clúster de caché

En el siguiente `delete-cache-cluster` ejemplo, se elimina el clúster previamente provisionado especificado. El comando elimina todos los nodos de caché asociados, los puntos finales de los nodos y el propio clúster. Cuando recibas una respuesta satisfactoria de esta operación, Amazon empezará ElastiCache inmediatamente a eliminar el clúster; no podrás cancelar ni revertir esta operación.

Esta operación no es válida para lo siguiente:

Clústeres de Redis (modo de clúster habilitado) Un clúster que es la última réplica de lectura de un grupo de replicación Un grupo de nodos (fragmento) que tiene habilitado el modo Multi-AZ Un clúster de un grupo de replicación de Redis (modo de clúster habilitado) Un clúster que no está en el estado disponible

```
aws elasticache delete-cache-cluster \
  --cache-cluster-id "my-cluster-002"
```

Salida:

```
{
  "CacheCluster": {
```

```
"CacheClusterId": "my-cluster-002",
"ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
"CacheNodeType": "cache.r5.xlarge",
"Engine": "redis",
"EngineVersion": "5.0.5",
"CacheClusterStatus": "deleting",
"NumCacheNodes": 1,
"PreferredAvailabilityZone": "us-west-2a",
"CacheClusterCreateTime": "2019-11-26T03:35:04.546Z",
"PreferredMaintenanceWindow": "mon:04:05-mon:05:05",
"PendingModifiedValues": {},
"NotificationConfiguration": {
  "TopicArn": "arn:aws:sns:us-west-x:xxxxxxx4152:My_Topic",
  "TopicStatus": "active"
},
"CacheSecurityGroups": [],
"CacheParameterGroup": {
  "CacheParameterGroupName": "mygroup",
  "ParameterApplyStatus": "in-sync",
  "CacheNodeIdsToReboot": []
},
"CacheSubnetGroupName": "kxkxk",
"AutoMinorVersionUpgrade": true,
"SecurityGroups": [
  {
    "SecurityGroupId": "sg-xxxxxxxxxx9836",
    "Status": "active"
  },
  {
    "SecurityGroupId": "sg-xxxxxxxxxx7b",
    "Status": "active"
  }
],
"ReplicationGroupId": "my-cluster",
"SnapshotRetentionLimit": 0,
"SnapshotWindow": "07:30-08:30",
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
}
```

[Para obtener más información, consulte Eliminar un clúster en la Guía del usuario de Elasticache.](#)

- Para API obtener más información, consulte la Referencia [DeleteCacheCluster](#) de AWS CLI comandos.

## **delete-cache-parameter-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-cache-parameter-group`.

### AWS CLI

Para eliminar un grupo de parámetros de caché

En el siguiente `delete-cache-parameter-group` ejemplo, se elimina el grupo de parámetros de caché especificado. No puede eliminar un grupo de parámetros de caché si está asociado a algún clúster de caché.

```
aws elasticache delete-cache-parameter-group \  
  --cache-parameter-group-name myparamgroup
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un grupo de parámetros](#) en la Guía del usuario de Elasticache.

- Para API obtener más información, consulte la Referencia [DeleteCacheParameterGroup](#) de AWS CLI comandos.

## **delete-cache-subnet-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-cache-subnet-group`.

### AWS CLI

Para eliminar un grupo de subredes de caché

En el siguiente `delete-cache-subnet-group` ejemplo, se elimina el grupo de subredes de caché especificado. No puedes eliminar un grupo de subredes de caché si está asociado a algún clúster.

```
aws elasticache delete-cache-subnet-group \  
  --cache-subnet-group-name "mygroup"
```



Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un grupo de subredes](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia de [DeleteCacheSubnetGroup](#) comandos AWS CLI .

## delete-global-replication-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-global-replication-group`.

### AWS CLI

Para eliminar un grupo de replicación global

En el siguiente `delete-global-replication-group` ejemplo, se elimina un nuevo grupo de replicación global.

```
aws elasticache delete-global-replication-group \  
  --global-replication-group-id my-global-replication-group \  
  --retain-primary-replication-group
```

Salida:

```
{  
  "GlobalReplicationGroup": {  
    "GlobalReplicationGroupId": "sgaui-my-grg",  
    "GlobalReplicationGroupDescription": "my-grg",  
    "Status": "deleting",  
    "CacheNodeType": "cache.r5.large",  
    "Engine": "redis",  
    "EngineVersion": "5.0.6",  
    "Members": [  
      {  
        "ReplicationGroupId": "my-cluster-grg",  
        "ReplicationGroupRegion": "us-west-2",  
        "Role": "PRIMARY",  
        "AutomaticFailover": "enabled",  
        "Status": "associated"  
      }  
    ],  
    "ClusterEnabled": false,  
  }  
}
```

```
    "AuthTokenEnabled": false,  
    "TransitEncryptionEnabled": false,  
    "AtRestEncryptionEnabled": false  
  }  
}
```

Para obtener más información, consulte [Replicación entre AWS regiones mediante un almacén de datos global en la Guía del usuario](#) de Elasticache.

- Para obtener API más información, consulte la Referencia de comandos [DeleteGlobalReplicationGroup](#).AWS CLI

## delete-replication-group

En el siguiente ejemplo de código, se muestra cómo usar delete-replication-group.

### AWS CLI

Para eliminar un grupo de replicación

En el siguiente delete-replication-group ejemplo, se elimina un grupo de replicación existente. De forma predeterminada, esta operación elimina todo el grupo de replicación, incluidas las réplicas principales/principales y todas las réplicas de lectura. Si el grupo de replicación tiene solo una principal, si lo desea, puede eliminar solo las réplicas de lectura y conservar la principal configurando =true. RetainPrimaryCluster

Cuando recibas una respuesta satisfactoria de esta operación, Amazon ElastiCache empezará inmediatamente a eliminar los recursos seleccionados; no podrás cancelar ni revertir esta operación. Válido solo para Redis.

```
aws elasticache delete-replication-group \  
  --replication-group-id "mygroup"
```

Salida:

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "mygroup",  
    "Description": "my group",  
    "Status": "deleting",  
    "PendingModifiedValues": {},  
    "AutomaticFailover": "disabled",
```

```

    "SnapshotRetentionLimit": 0,
    "SnapshotWindow": "06:00-07:00",
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false
  }
}

```

- Para API obtener más información, consulte [DeleteReplicationGroup](#) la Referencia de AWS CLI comandos.

## delete-snapshot

En el siguiente ejemplo de código, se muestra cómo usar delete-snapshot.

### AWS CLI

Eliminar una instantánea

En el siguiente delete-snapshot ejemplo, se eliminó una instantánea con el motor Redis.

```

aws elasticache delete-snapshot \
  --snapshot-name mynsnapshot

```

Salida:

```

{
  "Snapshot": {
    "SnapshotName": "my-cluster-snapshot",
    "ReplicationGroupId": "mycluster",
    "ReplicationGroupDescription": "mycluster",
    "SnapshotStatus": "deleting",
    "SnapshotSource": "manual",
    "CacheNodeType": "cache.r5.xlarge",
    "Engine": "redis",
    "EngineVersion": "5.0.5",
    "PreferredMaintenanceWindow": "thu:12:00-thu:13:00",
    "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxxxxx152:My_Topic",
    "Port": 6379,
    "CacheParameterGroupName": "default.redis5.0.cluster.on",
    "CacheSubnetGroupName": "default",
    "VpcId": "vpc-a3e97cdb",
    "AutoMinorVersionUpgrade": true,
  }
}

```

```

"SnapshotRetentionLimit": 1,
"SnapshotWindow": "13:00-14:00",
"NumNodeGroups": 4,
"AutomaticFailover": "enabled",
"NodeSnapshots": [
  {
    "CacheClusterId": "mycluster-0002-003",
    "NodeGroupId": "0002",
    "CacheNodeId": "0001",
    "CacheSize": "6 MB",
    "CacheNodeCreateTime": "2020-06-18T00:05:44.719000+00:00",
    "SnapshotCreateTime": "2020-06-25T20:34:30+00:00"
  },
  {
    "CacheClusterId": "mycluster-0003-003",
    "NodeGroupId": "0003",
    "CacheNodeId": "0001",
    "CacheSize": "6 MB",
    "CacheNodeCreateTime": "2019-12-05T19:13:15.912000+00:00",
    "SnapshotCreateTime": "2020-06-25T20:34:30+00:00"
  },
  {
    "CacheClusterId": "mycluster-0004-002",
    "NodeGroupId": "0004",
    "CacheNodeId": "0001",
    "CacheSize": "6 MB",
    "CacheNodeCreateTime": "2019-12-09T19:44:34.324000+00:00",
    "SnapshotCreateTime": "2020-06-25T20:34:30+00:00"
  },
  {
    "CacheClusterId": "mycluster-0005-003",
    "NodeGroupId": "0005",
    "CacheNodeId": "0001",
    "CacheSize": "6 MB",
    "CacheNodeCreateTime": "2020-06-18T00:05:44.775000+00:00",
    "SnapshotCreateTime": "2020-06-25T20:34:30+00:00"
  }
]
}

```

Para obtener más información, consulte [Backup and Restore ElastiCache for Redis](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia de [DeleteSnapshot](#) comandos AWS CLI .

## delete-user-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-user-group`.

### AWS CLI

Para eliminar un grupo de usuarios

En el siguiente `delete-user-group` ejemplo, se elimina un grupo de usuarios.

```
aws elasticache delete-user-group \  
  --user-group-id myusergroup
```

Salida:

```
{  
  "UserGroupId": "myusergroup",  
  "Status": "deleting",  
  "Engine": "redis",  
  "UserIds": [  
    "default"  
  ],  
  "ReplicationGroups": [],  
  "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:usergroup:myusergroup"  
}
```

Para obtener más información, consulte [Autenticación de usuarios con el control de acceso basado en roles \(RBAC\)](#) en la Guía del usuario de Elasticache.

- Para obtener API más información, consulte la Referencia de comandos. [DeleteUserGroup](#) AWS CLI

## delete-user

En el siguiente ejemplo de código, se muestra cómo usar `delete-user`.

### AWS CLI

Para eliminar un usuario

En el siguiente `delete-user` ejemplo, se elimina un usuario.

```
aws elasticache delete-user \  
  --user-id user2
```

Salida:

```
{  
  "UserId": "user1",  
  "UserName": "myUser",  
  "Status": "deleting",  
  "Engine": "redis",  
  "AccessString": "on ~* +@all",  
  "UserGroupIds": [  
    "myusergroup"  
  ],  
  "Authentication": {  
    "Type": "password",  
    "PasswordCount": 1  
  },  
  "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxxx52:user:user1"  
}
```

Para obtener más información, consulte [Autenticación de usuarios con el control de acceso basado en roles \(RBAC\)](#) en la Guía del usuario de Elasticache.

- Para obtener API más información, consulte la Referencia de comandos. [DeleteUser](#) AWS CLI

## describe-cache-clusters

En el siguiente ejemplo de código, se muestra cómo usar `describe-cache-clusters`.

AWS CLI

Para describir un clúster de caché

El siguiente `describe-cache-clusters` ejemplo describe un clúster de caché.

```
aws elasticache describe-cache-clusters
```

Salida:

```

{
  "CacheClusters": [
    {
      "CacheClusterId": "my-cluster-003",
      "ClientDownloadLandingPage": "https://console.aws.amazon.com/
elasticache/home#client-download:",
      "CacheNodeType": "cache.r5.large",
      "Engine": "redis",
      "EngineVersion": "5.0.5",
      "CacheClusterStatus": "available",
      "NumCacheNodes": 1,
      "PreferredAvailabilityZone": "us-west-2a",
      "CacheClusterCreateTime": "2019-11-26T01:22:52.396Z",
      "PreferredMaintenanceWindow": "mon:17:30-mon:18:30",
      "PendingModifiedValues": {},
      "NotificationConfiguration": {
        "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxxx152:My_Topic",
        "TopicStatus": "active"
      },
      "CacheSecurityGroups": [],
      "CacheParameterGroup": {
        "CacheParameterGroupName": "default.redis5.0",
        "ParameterApplyStatus": "in-sync",
        "CacheNodeIdsToReboot": []
      },
      "CacheSubnetGroupName": "kxkxk",
      "AutoMinorVersionUpgrade": true,
      "SecurityGroups": [
        {
          "SecurityGroupId": "sg-xxxxxd7b",
          "Status": "active"
        }
      ],
      "ReplicationGroupId": "my-cluster",
      "SnapshotRetentionLimit": 0,
      "SnapshotWindow": "06:30-07:30",
      "AuthTokenEnabled": false,
      "TransitEncryptionEnabled": false,
      "AtRestEncryptionEnabled": false,
      "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxxx152:cluster:my-cache-
cluster",
      "ReplicationGroupLogDeliveryEnabled": false,
      "LogDeliveryConfigurations": [

```

```

        {
            "LogType": "slow-log",
            "DestinationType": "cloudwatch-logs",
            "DestinationDetails": {
                "CloudWatchLogsDetails": {
                    "LogGroup": "test-log"
                }
            },
            "LogFormat": "text",
            "Status": "active"
        }
    ]
}

```

Para obtener más información, consulte [Gestión de clústeres](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia [DescribeCacheClusters](#) de AWS CLI comandos.

## describe-cache-engine-versions

En el siguiente ejemplo de código, se muestra cómo usar `describe-cache-engine-versions`.

### AWS CLI

Para describir una versión del motor de caché

El siguiente `describe-cache-engine-versions` ejemplo devuelve una lista de los motores de caché disponibles y sus versiones.

```
aws elasticache describe-cache-engine-versions \
  --engine Redis
```

Salida:

```

{
  "CacheEngineVersions": [
    {
      "Engine": "redis",
      "EngineVersion": "2.6.13",

```



```
    "CacheParameterGroupFamily": "redis2.6",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 2.6.13"
  },
  {
    "Engine": "redis",
    "EngineVersion": "2.8.19",
    "CacheParameterGroupFamily": "redis2.8",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 2.8.19"
  },
  {
    "Engine": "redis",
    "EngineVersion": "2.8.21",
    "CacheParameterGroupFamily": "redis2.8",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 2.8.21"
  },
  {
    "Engine": "redis",
    "EngineVersion": "2.8.22",
    "CacheParameterGroupFamily": "redis2.8",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 2.8.22"
  },
  {
    "Engine": "redis",
    "EngineVersion": "2.8.23",
    "CacheParameterGroupFamily": "redis2.8",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 2.8.23"
  },
  {
    "Engine": "redis",
    "EngineVersion": "2.8.24",
    "CacheParameterGroupFamily": "redis2.8",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 2.8.24"
  },
  {
    "Engine": "redis",
    "EngineVersion": "2.8.6",
    "CacheParameterGroupFamily": "redis2.8",
    "CacheEngineDescription": "Redis",
```

```
    "CacheEngineVersionDescription": "redis version 2.8.6"
  },
  {
    "Engine": "redis",
    "EngineVersion": "3.2.10",
    "CacheParameterGroupFamily": "redis3.2",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 3.2.10"
  },
  {
    "Engine": "redis",
    "EngineVersion": "3.2.4",
    "CacheParameterGroupFamily": "redis3.2",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 3.2.4"
  },
  {
    "Engine": "redis",
    "EngineVersion": "3.2.6",
    "CacheParameterGroupFamily": "redis3.2",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 3.2.6"
  },
  {
    "Engine": "redis",
    "EngineVersion": "4.0.10",
    "CacheParameterGroupFamily": "redis4.0",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 4.0.10"
  },
  {
    "Engine": "redis",
    "EngineVersion": "5.0.0",
    "CacheParameterGroupFamily": "redis5.0",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 5.0.0"
  },
  {
    "Engine": "redis",
    "EngineVersion": "5.0.3",
    "CacheParameterGroupFamily": "redis5.0",
    "CacheEngineDescription": "Redis",
    "CacheEngineVersionDescription": "redis version 5.0.3"
  },
},
```

```

    {
      "Engine": "redis",
      "EngineVersion": "5.0.4",
      "CacheParameterGroupFamily": "redis5.0",
      "CacheEngineDescription": "Redis",
      "CacheEngineVersionDescription": "redis version 5.0.4"
    },
    {
      "Engine": "redis",
      "EngineVersion": "5.0.5",
      "CacheParameterGroupFamily": "redis5.0",
      "CacheEngineDescription": "Redis",
      "CacheEngineVersionDescription": "redis version 5.0.5"
    }
  ]
}

```

- Para API obtener más información, consulte [DescribeCacheEngineVersions](#) la Referencia de AWS CLI comandos.

## describe-cache-parameter-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-cache-parameter-groups`.

### AWS CLI

Para describir un grupo de parámetros de caché

El siguiente `describe-cache-parameter-groups` ejemplo devuelve una lista de descripciones de grupos de parámetros de caché.

```

aws elasticache describe-cache-parameter-groups \
  --cache-parameter-group-name "mygroup"

```

Salida:

```

{
  "CacheParameterGroups": [
    {
      "CacheParameterGroupName": "mygroup",
      "CacheParameterGroupFamily": "redis5.0",
      "Description": " "
    }
  ]
}

```

```

    }
  ]
}

```

Para obtener más información, consulte [Configuración de los parámetros del motor mediante grupos de parámetros](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia [DescribeCacheParameterGroups](#) de AWS CLI comandos.

## describe-cache-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-cache-parameters`.

### AWS CLI

Para describir los parámetros de la caché

El siguiente ejemplo `describe-cache-parameters` «» devuelve la lista de parámetros detallada del grupo de parámetros de caché especificado.

```

aws elasticache describe-cache-parameters \
  --cache-parameter-group-name "myparamgroup"

```

Salida:

```

{
  "Parameters": [
    {
      "ParameterName": "activedefrag",
      "ParameterValue": "yes",
      "Description": "Enabled active memory defragmentation",
      "Source": "user",
      "DataType": "string",
      "AllowedValues": "yes,no",
      "IsModifiable": true,
      "MinimumEngineVersion": "5.0.0",
      "ChangeType": "immediate"
    },
    {
      "ParameterName": "active-defrag-cycle-max",
      "ParameterValue": "75",
      "Description": "Maximal effort for defrag in CPU percentage",

```

```
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-75",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-cycle-min",
    "ParameterValue": "5",
    "Description": "Minimal effort for defrag in CPU percentage",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-75",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-ignore-bytes",
    "ParameterValue": "104857600",
    "Description": "Minimum amount of fragmentation waste to start active
defrag",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1048576-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-max-scan-fields",
    "ParameterValue": "1000",
    "Description": "Maximum number of set/hash/zset/list fields that will be
processed from the main dictionary scan",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-1000000",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-threshold-lower",
```

```

    "ParameterValue": "10",
    "Description": "Minimum percentage of fragmentation to start active
defrag",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-100",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-threshold-upper",
    "ParameterValue": "100",
    "Description": "Maximum percentage of fragmentation at which we use
maximum effort",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-100",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "activeresharding",
    "ParameterValue": "yes",
    "Description": "Apply rehashing or not.",
    "Source": "user",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "requires-reboot"
  },
  {
    "ParameterName": "appendfsync",
    "ParameterValue": "everysec",
    "Description": "fsync policy for AOF persistence",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "always,everysec,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
},

```

```
{
  "ParameterName": "appendonly",
  "ParameterValue": "no",
  "Description": "Enable Redis persistence.",
  "Source": "system",
  "DataType": "string",
  "AllowedValues": "yes,no",
  "IsModifiable": false,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "client-output-buffer-limit-normal-hard-limit",
  "ParameterValue": "0",
  "Description": "Normal client output buffer hard limit in bytes.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "client-output-buffer-limit-normal-soft-limit",
  "ParameterValue": "0",
  "Description": "Normal client output buffer soft limit in bytes.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "client-output-buffer-limit-normal-soft-seconds",
  "ParameterValue": "0",
  "Description": "Normal client output buffer soft limit in seconds.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
}
```

```
{
  "ParameterName": "client-output-buffer-limit-pubsub-hard-limit",
  "ParameterValue": "33554432",
  "Description": "Pubsub client output buffer hard limit in bytes.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "client-output-buffer-limit-pubsub-soft-limit",
  "ParameterValue": "8388608",
  "Description": "Pubsub client output buffer soft limit in bytes.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "client-output-buffer-limit-pubsub-soft-seconds",
  "ParameterValue": "60",
  "Description": "Pubsub client output buffer soft limit in seconds.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "client-output-buffer-limit-replica-soft-seconds",
  "ParameterValue": "60",
  "Description": "Replica client output buffer soft limit in seconds.",
  "Source": "system",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": false,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
},
```



```
{
  "ParameterName": "client-query-buffer-limit",
  "ParameterValue": "1073741824",
  "Description": "Max size of a single client query buffer",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "1048576-1073741824",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "close-on-replica-write",
  "ParameterValue": "yes",
  "Description": "If enabled, clients who attempt to write to a read-only
replica will be disconnected. Applicable to 2.8.23 and higher.",
  "Source": "user",
  "DataType": "string",
  "AllowedValues": "yes,no",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "cluster-enabled",
  "ParameterValue": "no",
  "Description": "Enable cluster mode",
  "Source": "user",
  "DataType": "string",
  "AllowedValues": "yes,no",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "requires-reboot"
},
{
  "ParameterName": "cluster-require-full-coverage",
  "ParameterValue": "no",
  "Description": "Whether cluster becomes unavailable if one or more slots
are not covered",
  "Source": "user",
  "DataType": "string",
  "AllowedValues": "yes,no",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
```

```
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "databases",
    "ParameterValue": "16",
    "Description": "Set the number of databases.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-1200000",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "requires-reboot"
  },
  {
    "ParameterName": "hash-max-ziplist-entries",
    "ParameterValue": "512",
    "Description": "The maximum number of hash entries in order for the
dataset to be compressed.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "hash-max-ziplist-value",
    "ParameterValue": "64",
    "Description": "The threshold of biggest hash entries in order for the
dataset to be compressed.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "hll-sparse-max-bytes",
    "ParameterValue": "3000",
    "Description": "HyperLogLog sparse representation bytes limit",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-16000",
```

```
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lazyfree-lazy-eviction",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous delete on evictions",
    "Source": "user",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lazyfree-lazy-expire",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous delete on expired keys",
    "Source": "user",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lazyfree-lazy-server-del",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous delete on key updates",
    "Source": "user",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lfu-decay-time",
    "ParameterValue": "1",
    "Description": "The amount of time in minutes to decrement the key
counter for LFU eviction policy",
    "Source": "user",
    "DataType": "integer",
```

```
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lfu-log-factor",
    "ParameterValue": "10",
    "Description": "The log factor for incrementing key counter for LFU
eviction policy",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "list-compress-depth",
    "ParameterValue": "0",
    "Description": "Number of quicklist ziplist nodes from each side of
the list to exclude from compression. The head and tail of the list are always
uncompressed for fast push/pop operations",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "list-max-ziplist-size",
    "ParameterValue": "-2",
    "Description": "The number of entries allowed per internal list node can
be specified as a fixed maximum size or a maximum number of elements",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "-5,-4,-3,-2,-1,1-",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lua-replicate-commands",
```

```
    "ParameterValue": "yes",
    "Description": "Always enable Lua effect replication or not",
    "Source": "user",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lua-time-limit",
    "ParameterValue": "5000",
    "Description": "Max execution time of a Lua script in milliseconds. 0
for unlimited execution without warnings.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "5000",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "maxclients",
    "ParameterValue": "65000",
    "Description": "The maximum number of Redis clients.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "1-65000",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "requires-reboot"
  },
  {
    "ParameterName": "maxmemory-policy",
    "ParameterValue": "volatile-lru",
    "Description": "Max memory policy.",
    "Source": "user",
    "DataType": "string",
    "AllowedValues": "volatile-lru,allkeys-lru,volatile-lfu,allkeys-
lfu,volatile-random,allkeys-random,volatile-ttl,noeviction",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
},
```

```
{
  "ParameterName": "maxmemory-samples",
  "ParameterValue": "3",
  "Description": "Max memory samples.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "1-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "min-replicas-max-lag",
  "ParameterValue": "10",
  "Description": "The maximum amount of replica lag in seconds beyond
which the master would stop taking writes. A value of 0 means the master always
takes writes.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "min-replicas-to-write",
  "ParameterValue": "0",
  "Description": "The minimum number of replicas that must be present with
lag no greater than min-replicas-max-lag for master to take writes. Setting this to
0 means the master always takes writes.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "notify-keyspace-events",
  "Description": "The keyspace events for Redis to notify Pub/Sub clients
about. By default all notifications are disabled",
  "Source": "user",
  "DataType": "string",
  "IsModifiable": true,
```

```

    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "proto-max-bulk-len",
    "ParameterValue": "536870912",
    "Description": "Max size of a single element request",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "1048576-536870912",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "rename-commands",
    "ParameterValue": "",
    "Description": "Redis commands that can be dynamically renamed by the
customer",
    "Source": "user",
    "DataType": "string",
    "AllowedValues":
"APPEND,BITCOUNT,BITFIELD,BITOP,BITPOS,BLPOP,BRPOP,BRPOPLPUSH,BZPOPMIN,BZPOPMAX,CLIENT,COMM
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.3",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "repl-backlog-size",
    "ParameterValue": "1048576",
    "Description": "The replication backlog size in bytes for PSYNC. This is
the size of the buffer which accumulates slave data when slave is disconnected for
some time, so that when slave reconnects again, only transfer the portion of data
which the slave missed. Minimum value is 16K.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "16384-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "repl-backlog-ttl",
    "ParameterValue": "3600",

```

```
    "Description": "The amount of time in seconds after the master no longer
have any slaves connected for the master to free the replication backlog. A value
of 0 means to never release the backlog.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "replica-allow-chaining",
    "ParameterValue": "no",
    "Description": "Configures if chaining of replicas is allowed",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "replica-ignore-maxmemory",
    "ParameterValue": "yes",
    "Description": "Determines if replica ignores maxmemory setting by not
evicting items independent from the master",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "replica-lazy-flush",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous flushDB during replica sync",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
},
```



```
{
  "ParameterName": "reserved-memory-percent",
  "ParameterValue": "25",
  "Description": "The percent of memory reserved for non-cache memory
usage. You may want to increase this parameter for nodes with read replicas, AOF
enabled, etc, to reduce swap usage.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-100",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "set-max-intset-entries",
  "ParameterValue": "512",
  "Description": "The limit in the size of the set in order for the
dataset to be compressed.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "0-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "slowlog-log-slower-than",
  "ParameterValue": "10000",
  "Description": "The execution time, in microseconds, to exceed in order
for the command to get logged. Note that a negative number disables the slow log,
while a value of zero forces the logging of every command.",
  "Source": "user",
  "DataType": "integer",
  "AllowedValues": "-",
  "IsModifiable": true,
  "MinimumEngineVersion": "5.0.0",
  "ChangeType": "immediate"
},
{
  "ParameterName": "slowlog-max-len",
  "ParameterValue": "128",
  "Description": "The length of the slow log. There is no limit to this
length. Just be aware that it will consume memory. You can reclaim memory used by
the slow log with SLOWLOG RESET.",
```

```
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "stream-node-max-bytes",
    "ParameterValue": "4096",
    "Description": "The maximum size of a single node in a stream in bytes",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "stream-node-max-entries",
    "ParameterValue": "100",
    "Description": "The maximum number of items a single node in a stream
can contain",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "tcp-keepalive",
    "ParameterValue": "300",
    "Description": "If non-zero, send ACKs every given number of seconds.",
    "Source": "user",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "timeout",
    "ParameterValue": "0",
```

```

        "Description": "Close connection if client is idle for a given number of
seconds, or never if 0.",
        "Source": "user",
        "DataType": "integer",
        "AllowedValues": "0,20-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "zset-max-ziplist-entries",
        "ParameterValue": "128",
        "Description": "The maximum number of sorted set entries in order for
the dataset to be compressed.",
        "Source": "user",
        "DataType": "integer",
        "AllowedValues": "0-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "zset-max-ziplist-value",
        "ParameterValue": "64",
        "Description": "The threshold of biggest sorted set entries in order for
the dataset to be compressed.",
        "Source": "user",
        "DataType": "integer",
        "AllowedValues": "0-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    }
}
]
}

```

Para obtener más información, consulte [Administración de parámetros](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia [DescribeCacheParameters](#) de AWS CLI comandos.

## describe-cache-subnet-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-cache-subnet-groups`.

### AWS CLI

Para describir los grupos de subredes de caché

El siguiente `describe-cache-subnet-groups` ejemplo devuelve una lista de grupos de subredes.

```
aws elasticache describe-cache-subnet-groups
```

Salida:

```
{
  "CacheSubnetGroups": [
    {
      "CacheSubnetGroupName": "default",
      "CacheSubnetGroupDescription": "Default CacheSubnetGroup",
      "VpcId": "vpc-a3e97cdb",
      "Subnets": [
        {
          "SubnetIdentifier": "subnet-8d4bacf5",
          "SubnetAvailabilityZone": {
            "Name": "us-west-2b"
          }
        },
        {
          "SubnetIdentifier": "subnet-dde21380",
          "SubnetAvailabilityZone": {
            "Name": "us-west-2c"
          }
        },
        {
          "SubnetIdentifier": "subnet-6485ec4f",
          "SubnetAvailabilityZone": {
            "Name": "us-west-2d"
          }
        },
        {
          "SubnetIdentifier": "subnet-b4ebebff",
          "SubnetAvailabilityZone": {
```

```

        "Name": "us-west-2a"
      }
    ]
  },
  {
    "CacheSubnetGroupName": "kxkxk",
    "CacheSubnetGroupDescription": "mygroup",
    "VpcId": "vpc-a3e97cdb",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-b4ebebff",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2a"
        }
      }
    ]
  },
  {
    "CacheSubnetGroupName": "test",
    "CacheSubnetGroupDescription": "test",
    "VpcId": "vpc-a3e97cdb",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-b4ebebff",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2a"
        }
      }
    ]
  }
]
}

```

Para obtener más información, consulte [Subredes y grupos de subredes](#) en la Guía del usuario de Elasticache o Subredes y grupos de subredes en la Guía del usuario [de Memcached](#).  
ElastiCache

- Para obtener más información, consulte la Referencia de comandos. API [DescribeCacheSubnetGroups](#) AWS CLI

## describe-engine-default-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-engine-default-parameters`.

### AWS CLI

Para describir los parámetros por defecto del motor

El siguiente `describe-engine-default-parameters` ejemplo devuelve la información de los parámetros predeterminados del motor y del sistema para el motor de caché especificado.

```
aws elasticache describe-engine-default-parameters \  
  --cache-parameter-group-family "redis5.0"
```

Salida:

```
{  
  "EngineDefaults": {  
    "Parameters": [  
      {  
        "ParameterName": "activedefrag",  
        "ParameterValue": "no",  
        "Description": "Enabled active memory defragmentation",  
        "Source": "system",  
        "DataType": "string",  
        "AllowedValues": "yes,no",  
        "IsModifiable": true,  
        "MinimumEngineVersion": "5.0.0",  
        "ChangeType": "immediate"  
      },  
      {  
        "ParameterName": "active-defrag-cycle-max",  
        "ParameterValue": "75",  
        "Description": "Maximal effort for defrag in CPU percentage",  
        "Source": "system",  
        "DataType": "integer",  
        "AllowedValues": "1-75",  
        "IsModifiable": true,  
        "MinimumEngineVersion": "5.0.0",  
        "ChangeType": "immediate"  
      },  
      {
```

```
    "ParameterName": "active-defrag-cycle-min",
    "ParameterValue": "5",
    "Description": "Minimal effort for defrag in CPU percentage",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "1-75",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-ignore-bytes",
    "ParameterValue": "104857600",
    "Description": "Minimum amount of fragmentation waste to start
active defrag",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "1048576-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-max-scan-fields",
    "ParameterValue": "1000",
    "Description": "Maximum number of set/hash/zset/list fields that
will be processed from the main dictionary scan",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "1-1000000",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "active-defrag-threshold-lower",
    "ParameterValue": "10",
    "Description": "Minimum percentage of fragmentation to start active
defrag",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "1-100",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
```

```
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "active-defrag-threshold-upper",
        "ParameterValue": "100",
        "Description": "Maximum percentage of fragmentation at which we use
maximum effort",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "1-100",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "activeresharding",
        "ParameterValue": "yes",
        "Description": "Apply rehashing or not.",
        "Source": "system",
        "DataType": "string",
        "AllowedValues": "yes,no",
        "IsModifiable": false,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "requires-reboot"
    },
    {
        "ParameterName": "appendfsync",
        "ParameterValue": "everysec",
        "Description": "fsync policy for AOF persistence",
        "Source": "system",
        "DataType": "string",
        "AllowedValues": "always,everysec,no",
        "IsModifiable": false,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "appendonly",
        "ParameterValue": "no",
        "Description": "Enable Redis persistence.",
        "Source": "system",
        "DataType": "string",
        "AllowedValues": "yes,no",
        "IsModifiable": false,
```



```
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-normal-hard-limit",
    "ParameterValue": "0",
    "Description": "Normal client output buffer hard limit in bytes.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-normal-soft-limit",
    "ParameterValue": "0",
    "Description": "Normal client output buffer soft limit in bytes.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-normal-soft-seconds",
    "ParameterValue": "0",
    "Description": "Normal client output buffer soft limit in seconds.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-pubsub-hard-limit",
    "ParameterValue": "33554432",
    "Description": "Pubsub client output buffer hard limit in bytes.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
```

```
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-pubsub-soft-limit",
    "ParameterValue": "8388608",
    "Description": "Pubsub client output buffer soft limit in bytes.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-pubsub-soft-seconds",
    "ParameterValue": "60",
    "Description": "Pubsub client output buffer soft limit in seconds.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-output-buffer-limit-replica-soft-seconds",
    "ParameterValue": "60",
    "Description": "Replica client output buffer soft limit in
seconds.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "client-query-buffer-limit",
    "ParameterValue": "1073741824",
    "Description": "Max size of a single client query buffer",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "1048576-1073741824",
```

```
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "close-on-replica-write",
    "ParameterValue": "yes",
    "Description": "If enabled, clients who attempt to write to a read-
only replica will be disconnected. Applicable to 2.8.23 and higher.",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "cluster-enabled",
    "ParameterValue": "no",
    "Description": "Enable cluster mode",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "requires-reboot"
  },
  {
    "ParameterName": "cluster-require-full-coverage",
    "ParameterValue": "no",
    "Description": "Whether cluster becomes unavailable if one or more
slots are not covered",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "databases",
    "ParameterValue": "16",
    "Description": "Set the number of databases.",
    "Source": "system",
```

```
        "DataType": "integer",
        "AllowedValues": "1-1200000",
        "IsModifiable": false,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "requires-reboot"
    },
    {
        "ParameterName": "hash-max-ziplist-entries",
        "ParameterValue": "512",
        "Description": "The maximum number of hash entries in order for the
dataset to be compressed.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "hash-max-ziplist-value",
        "ParameterValue": "64",
        "Description": "The threshold of biggest hash entries in order for
the dataset to be compressed.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "hll-sparse-max-bytes",
        "ParameterValue": "3000",
        "Description": "HyperLogLog sparse representation bytes limit",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "1-16000",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "lazyfree-lazy-eviction",
        "ParameterValue": "no",
```

```
    "Description": "Perform an asynchronous delete on evictions",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lazyfree-lazy-expire",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous delete on expired keys",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lazyfree-lazy-server-del",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous delete on key updates",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lfu-decay-time",
    "ParameterValue": "1",
    "Description": "The amount of time in minutes to decrement the key
counter for LFU eviction policy",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lfu-log-factor",
```

```
    "ParameterValue": "10",
    "Description": "The log factor for incrementing key counter for LFU
eviction policy",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "1-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "list-compress-depth",
    "ParameterValue": "0",
    "Description": "Number of quicklist ziplist nodes from each side
of the list to exclude from compression. The head and tail of the list are always
uncompressed for fast push/pop operations",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "list-max-ziplist-size",
    "ParameterValue": "-2",
    "Description": "The number of entries allowed per internal list node
can be specified as a fixed maximum size or a maximum number of elements",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "-5,-4,-3,-2,-1,1-",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "lua-replicate-commands",
    "ParameterValue": "yes",
    "Description": "Always enable Lua effect replication or not",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
```

```

        "ChangeType": "immediate"
    },
    {
        "ParameterName": "lua-time-limit",
        "ParameterValue": "5000",
        "Description": "Max execution time of a Lua script in milliseconds.
0 for unlimited execution without warnings.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "5000",
        "IsModifiable": false,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "maxclients",
        "ParameterValue": "65000",
        "Description": "The maximum number of Redis clients.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "1-65000",
        "IsModifiable": false,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "requires-reboot"
    },
    {
        "ParameterName": "maxmemory-policy",
        "ParameterValue": "volatile-lru",
        "Description": "Max memory policy.",
        "Source": "system",
        "DataType": "string",
        "AllowedValues": "volatile-lru,allkeys-lru,volatile-lfu,allkeys-
lfu,volatile-random,allkeys-random,volatile-ttl,noeviction",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "maxmemory-samples",
        "ParameterValue": "3",
        "Description": "Max memory samples.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "1-",

```

```
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "min-replicas-max-lag",
    "ParameterValue": "10",
    "Description": "The maximum amount of replica lag in seconds beyond
which the master would stop taking writes. A value of 0 means the master always
takes writes.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "min-replicas-to-write",
    "ParameterValue": "0",
    "Description": "The minimum number of replicas that must be present
with lag no greater than min-replicas-max-lag for master to take writes. Setting
this to 0 means the master always takes writes.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "notify-keyspace-events",
    "Description": "The keyspace events for Redis to notify Pub/Sub
clients about. By default all notifications are disabled",
    "Source": "system",
    "DataType": "string",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "proto-max-bulk-len",
    "ParameterValue": "536870912",
    "Description": "Max size of a single element request",
```



```

        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "1048576-536870912",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "rename-commands",
        "ParameterValue": "",
        "Description": "Redis commands that can be dynamically renamed by
the customer",
        "Source": "system",
        "DataType": "string",
        "AllowedValues":
"APPEND,BITCOUNT,BITFIELD,BITOP,BITPOS,BLPOP,BRPOP,BRPOPLUSH,BZPOPMIN,BZPOPMAX,CLIENT,COMM
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.3",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "repl-backlog-size",
        "ParameterValue": "1048576",
        "Description": "The replication backlog size in bytes for PSYNC.
This is the size of the buffer which accumulates slave data when slave is
disconnected for some time, so that when slave reconnects again, only transfer the
portion of data which the slave missed. Minimum value is 16K.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "16384-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "repl-backlog-ttl",
        "ParameterValue": "3600",
        "Description": "The amount of time in seconds after the master no
longer have any slaves connected for the master to free the replication backlog. A
value of 0 means to never release the backlog.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0-",
        "IsModifiable": true,

```

```
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "replica-allow-chaining",
    "ParameterValue": "no",
    "Description": "Configures if chaining of replicas is allowed",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "replica-ignore-maxmemory",
    "ParameterValue": "yes",
    "Description": "Determines if replica ignores maxmemory setting by
not evicting items independent from the master",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "replica-lazy-flush",
    "ParameterValue": "no",
    "Description": "Perform an asynchronous flushDB during replica
sync",
    "Source": "system",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "IsModifiable": false,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "reserved-memory-percent",
    "ParameterValue": "25",
    "Description": "The percent of memory reserved for non-cache memory
usage. You may want to increase this parameter for nodes with read replicas, AOF
enabled, etc, to reduce swap usage.",
```

```
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-100",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "set-max-intset-entries",
    "ParameterValue": "512",
    "Description": "The limit in the size of the set in order for the
dataset to be compressed.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "slowlog-log-slower-than",
    "ParameterValue": "10000",
    "Description": "The execution time, in microseconds, to exceed in
order for the command to get logged. Note that a negative number disables the slow
log, while a value of zero forces the logging of every command.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  },
  {
    "ParameterName": "slowlog-max-len",
    "ParameterValue": "128",
    "Description": "The length of the slow log. There is no limit to
this length. Just be aware that it will consume memory. You can reclaim memory used
by the slow log with SLOWLOG RESET.",
    "Source": "system",
    "DataType": "integer",
    "AllowedValues": "0-",
    "IsModifiable": true,
    "MinimumEngineVersion": "5.0.0",
    "ChangeType": "immediate"
  }
}
```

```
    },
    {
      "ParameterName": "stream-node-max-bytes",
      "ParameterValue": "4096",
      "Description": "The maximum size of a single node in a stream in
bytes",
      "Source": "system",
      "DataType": "integer",
      "AllowedValues": "0-",
      "IsModifiable": true,
      "MinimumEngineVersion": "5.0.0",
      "ChangeType": "immediate"
    },
    {
      "ParameterName": "stream-node-max-entries",
      "ParameterValue": "100",
      "Description": "The maximum number of items a single node in a
stream can contain",
      "Source": "system",
      "DataType": "integer",
      "AllowedValues": "0-",
      "IsModifiable": true,
      "MinimumEngineVersion": "5.0.0",
      "ChangeType": "immediate"
    },
    {
      "ParameterName": "tcp-keepalive",
      "ParameterValue": "300",
      "Description": "If non-zero, send ACKs every given number of
seconds.",
      "Source": "system",
      "DataType": "integer",
      "AllowedValues": "0-",
      "IsModifiable": true,
      "MinimumEngineVersion": "5.0.0",
      "ChangeType": "immediate"
    },
    {
      "ParameterName": "timeout",
      "ParameterValue": "0",
      "Description": "Close connection if client is idle for a given
number of seconds, or never if 0.",
      "Source": "system",
      "DataType": "integer",
```

```

        "AllowedValues": "0,20-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "zset-max-ziplist-entries",
        "ParameterValue": "128",
        "Description": "The maximum number of sorted set entries in order
for the dataset to be compressed.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    },
    {
        "ParameterName": "zset-max-ziplist-value",
        "ParameterValue": "64",
        "Description": "The threshold of biggest sorted set entries in order
for the dataset to be compressed.",
        "Source": "system",
        "DataType": "integer",
        "AllowedValues": "0-",
        "IsModifiable": true,
        "MinimumEngineVersion": "5.0.0",
        "ChangeType": "immediate"
    }
}
]
}
}

```

- Para API obtener más información, consulte [DescribeEngineDefaultParameters](#) la Referencia de AWS CLI comandos.

## describe-events

En el siguiente ejemplo de código, se muestra cómo usar `describe-events`.

### AWS CLI

Para describir los eventos de un grupo de replicación

El siguiente `describe-events` ejemplo devuelve una lista de eventos de un grupo de replicación.

```
aws elasticache describe-events \  
  --source-identifier test-cluster \  
  --source-type replication-group
```

Salida:

```
{  
  "Events": [  
    {  
      "SourceIdentifier": "test-cluster",  
      "SourceType": "replication-group",  
      "Message": "Automatic failover has been turned on for replication group  
test-cluster",  
      "Date": "2020-03-18T23:51:34.457Z"  
    },  
    {  
      "SourceIdentifier": "test-cluster",  
      "SourceType": "replication-group",  
      "Message": "Replication group test-cluster created",  
      "Date": "2020-03-18T23:50:31.378Z"  
    }  
  ]  
}
```

Para obtener más información, consulte [Monitorización de eventos](#) en la Guía del usuario de Elasticache.

- Para API obtener más información, consulte la Referencia [DescribeEvents](#) de AWS CLI comandos.

## **describe-global-replication-groups**

En el siguiente ejemplo de código, se muestra cómo usar `describe-global-replication-groups`.

AWS CLI

Para describir los grupos de replicación globales

El siguiente `describe-global-replication-groups` ejemplo devuelve los detalles de un almacén de datos global.

```
aws elasticache describe-global-replication-groups \  
--global-replication-group-id my-grg
```

Salida:

```
{  
  "GlobalReplicationGroups": [  
    {  
      "GlobalReplicationGroupId": "my-grg",  
      "GlobalReplicationGroupDescription": "my-grg",  
      "Status": "creating",  
      "CacheNodeType": "cache.r5.large",  
      "Engine": "redis",  
      "EngineVersion": "5.0.6",  
      "ClusterEnabled": false,  
      "AuthTokenEnabled": false,  
      "TransitEncryptionEnabled": false,  
      "AtRestEncryptionEnabled": false  
    }  
  ]  
}
```

Para obtener más información, consulte [Replicación entre AWS regiones mediante un almacén de datos global en la Guía del usuario de Elasticache](#).

- Para obtener API más información, consulte la Referencia de comandos [DescribeGlobalReplicationGroups](#).AWS CLI

## **describe-replication-groups**

En el siguiente ejemplo de código, se muestra cómo usar `describe-replication-groups`.

AWS CLI

Para devolver una lista de los detalles del grupo de replicación

El siguiente `describe-replication-groups` ejemplo devuelve los grupos de replicación.

```
aws elasticache describe-replication-groups
```

**Salida:**

```
{
  "ReplicationGroups": [
    {
      "ReplicationGroupId": "my-cluster",
      "Description": "mycluster",
      "Status": "available",
      "PendingModifiedValues": {},
      "MemberClusters": [
        "pat-cluster-001",
        "pat-cluster-002",
        "pat-cluster-003",
        "pat-cluster-004"
      ],
      "NodeGroups": [
        {
          "NodeGroupId": "0001",
          "Status": "available",
          "PrimaryEndpoint": {
            "Address": "my-
cluster.xxxxih.ng.0001.usw2.cache.amazonaws.com",
            "Port": 6379
          },
          "ReaderEndpoint": {
            "Address": "my-cluster-
ro.xxxxih.ng.0001.usw2.cache.amazonaws.com",
            "Port": 6379
          },
          "NodeGroupMembers": [
            {
              "CacheClusterId": "my-cluster-001",
              "CacheNodeId": "0001",
              "ReadEndpoint": {
                "Address": "pat-
cluster-001.xxxxih.0001.usw2.cache.amazonaws.com",
                "Port": 6379
              },
              "PreferredAvailabilityZone": "us-west-2a",
              "CurrentRole": "primary"
            },
            {
              "CacheClusterId": "my-cluster-002",
              "CacheNodeId": "0001",
```



```

        "ReadEndpoint": {
            "Address": "pat-
cluster-002.xxxxih.0001.usw2.cache.amazonaws.com",
            "Port": 6379
        },
        "PreferredAvailabilityZone": "us-west-2a",
        "CurrentRole": "replica"
    },
    {
        "CacheClusterId": "my-cluster-003",
        "CacheNodeId": "0001",
        "ReadEndpoint": {
            "Address": "pat-
cluster-003.xxxxih.0001.usw2.cache.amazonaws.com",
            "Port": 6379
        },
        "PreferredAvailabilityZone": "us-west-2a",
        "CurrentRole": "replica"
    },
    {
        "CacheClusterId": "my-cluster-004",
        "CacheNodeId": "0001",
        "ReadEndpoint": {
            "Address": "pat-
cluster-004.xxxxih.0001.usw2.cache.amazonaws.com",
            "Port": 6379
        },
        "PreferredAvailabilityZone": "us-west-2a",
        "CurrentRole": "replica"
    }
]
}
],
"AutomaticFailover": "disabled",
"SnapshotRetentionLimit": 0,
"SnapshotWindow": "07:30-08:30",
"ClusterEnabled": false,
"CacheNodeType": "cache.r5.xlarge",
"AuthTokenEnabled": false,
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false,
"ARN": "arn:aws:elasticache:us-
west-2:xxxxxxxxxxxx152:replicationgroup:my-cluster",
"LogDeliveryConfigurations": [

```

```
    {
      "LogType": "slow-log",
      "DestinationType": "cloudwatch-logs",
      "DestinationDetails": {
        "CloudWatchLogsDetails": {
          "LogGroup": "test-log"
        }
      },
      "LogFormat": "json",
      "Status": "active"
    }
  ]
}
```

Para obtener más información, consulte [Gestión de clústeres](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia [DescribeReplicationGroups](#) de AWS CLI comandos.

## **describe-reserved-cache-nodes-offerings**

En el siguiente ejemplo de código, se muestra cómo usar `describe-reserved-cache-nodes-offerings`.

AWS CLI

Para describir `reserved-cache-nodes-offerings`

El siguiente `describe-reserved-cache-nodes-offerings` ejemplo devuelve los detalles de una `reserved-cache-node` opción.

```
aws elasticache describe-reserved-cache-nodes-offerings
```

Salida:

```
{
  "ReservedCacheNodesOfferings": [
    {
```

```

    "ReservedCacheNodesOfferingId": "01ce0a19-a476-41cb-8aee-48eacbcd8e5",
    "CacheNodeType": "cache.t3.small",
    "Duration": 31536000,
    "FixedPrice": 97.0,
    "UsagePrice": 0.0,
    "ProductDescription": "memcached",
    "OfferingType": "Partial Upfront",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": 0.011,
        "RecurringChargeFrequency": "Hourly"
      }
    ]
  },
  {
    "ReservedCacheNodesOfferingId": "0443a27b-4da5-4b90-b92d-929fbd7abed2",
    "CacheNodeType": "cache.m3.2xlarge",
    "Duration": 31536000,
    "FixedPrice": 1772.0,
    "UsagePrice": 0.0,
    "ProductDescription": "redis",
    "OfferingType": "Heavy Utilization",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": 0.25,
        "RecurringChargeFrequency": "Hourly"
      }
    ]
  },
  ...
]
}

```

Para obtener más información, consulte [Obtener información sobre las ofertas de nodos reservados](#) en la Guía del usuario de Elasticache Redis o [Obtener información sobre las ofertas de nodos reservados](#) en la Guía del usuario de Elasticache Memcached.

- Para API obtener más información, consulte la Referencia de comandos. [DescribeReservedCacheNodesOfferings](#) AWS CLI

## describe-reserved-cache-nodes

En el siguiente ejemplo de código, se muestra cómo usar `describe-reserved-cache-nodes`.

### AWS CLI

Para describir los nodos de caché reservados

El siguiente `describe-reserved-cache-nodes` ejemplo devuelve información sobre los nodos de caché reservados para esta cuenta o sobre el nodo de caché reservado especificado.

```
aws elasticache describe-reserved-cache-nodes
```

Salida:

```
{
  "ReservedCacheNodes": [
    {
      "ReservedCacheNodeId": "mynode",
      "ReservedCacheNodesOfferingId": "xxxxxxxxxx-xxxxxx-xxxxxx-xxxx-xxxxxxxxxx71",
      "CacheNodeType": "cache.t3.small",
      "StartTime": "2019-12-06T02:50:44.003Z",
      "Duration": 31536000,
      "FixedPrice": 0.0,
      "UsagePrice": 0.0,
      "CacheNodeCount": 1,
      "ProductDescription": "redis",
      "OfferingType": "No Upfront",
      "State": "payment-pending",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": 0.023,
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "ReservationARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxxxxx52:reserved-instance:mynode"
    }
  ]
}
```

Para obtener más información, consulte [Administración de costos con nodos reservados](#) en la Guía del usuario de Elasticache.

- Para API obtener más información, consulte la Referencia [DescribeReservedCacheNodes](#) de AWS CLI comandos.

## describe-service-updates

En el siguiente ejemplo de código, se muestra cómo usar describe-service-updates.

### AWS CLI

Para describir las actualizaciones del servicio

En el siguiente describe-service-updates ejemplo, se muestran detalles sobre las actualizaciones del servicio.

```
aws elasticache describe-service-updates
```

Salida:

```
{
  "ServiceUpdates": [
    {
      "ServiceUpdateName": "elc-xxxxxxx7-001",
      "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
      "ServiceUpdateEndDate": "2020-02-09T15:59:59Z",
      "ServiceUpdateSeverity": "important",
      "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
      "ServiceUpdateStatus": "available",
      "ServiceUpdateDescription": "Upgrades to improve the security,
reliability, and operational performance of your ElastiCache nodes",
      "ServiceUpdateType": "security-update",
      "Engine": "redis, memcached",
      "EngineVersion": "redis 2.6.13 and onwards, memcached 1.4.5 and
onwards",
      "AutoUpdateAfterRecommendedApplyByDate": false,
      "EstimatedUpdateTime": "30 minutes per node"
    },
    {
      "ServiceUpdateName": "elc-xxxxxxx4-001",
      "ServiceUpdateReleaseDate": "2019-06-11T15:00:00Z",
      "ServiceUpdateEndDate": "2019-10-01T09:24:00Z",
      "ServiceUpdateSeverity": "important",
      "ServiceUpdateRecommendedApplyByDate": "2019-07-11T14:59:59Z",
```

```

        "ServiceUpdateStatus": "expired",
        "ServiceUpdateDescription": "Upgrades to improve the security,
reliability, and operational performance of your ElastiCache nodes",
        "ServiceUpdateType": "security-update",
        "Engine": "redis",
        "EngineVersion": "redis 3.2.6, redis 4.0 and onwards",
        "AutoUpdateAfterRecommendedApplyByDate": false,
        "EstimatedUpdateTime": "30 minutes per node"
    }
]
}

```

- Para API obtener más información, consulte [DescribeServiceUpdates](#) la Referencia de AWS CLI comandos.

## describe-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `describe-snapshots`.

### AWS CLI

Para describir las instantáneas

El siguiente ejemplo de «`describe-snapshots`» devuelve información sobre las instantáneas de su clúster o grupo de replicación.

```
aws elasticache describe-snapshots
```

Salida:

```

{
  "Snapshots": [
    {
      "SnapshotName": "automatic.my-cluster2-002-2019-12-05-06-38",
      "CacheClusterId": "my-cluster2-002",
      "SnapshotStatus": "available",
      "SnapshotSource": "automated",
      "CacheNodeType": "cache.r5.large",
      "Engine": "redis",
      "EngineVersion": "5.0.5",
      "NumCacheNodes": 1,
      "PreferredAvailabilityZone": "us-west-2a",
    }
  ]
}

```

```

    "CacheClusterCreateTime": "2019-11-26T01:22:52.396Z",
    "PreferredMaintenanceWindow": "mon:17:30-mon:18:30",
    "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxx52:My_Topic",
    "Port": 6379,
    "CacheParameterGroupName": "default.redis5.0",
    "CacheSubnetGroupName": "kxkxk",
    "VpcId": "vpc-a3e97cdb",
    "AutoMinorVersionUpgrade": true,
    "SnapshotRetentionLimit": 1,
    "SnapshotWindow": "06:30-07:30",
    "NodeSnapshots": [
      {
        "CacheNodeId": "0001",
        "CacheSize": "5 MB",
        "CacheNodeCreateTime": "2019-11-26T01:22:52.396Z",
        "SnapshotCreateTime": "2019-12-05T06:38:23Z"
      }
    ]
  },
  {
    "SnapshotName": "myreplica-backup",
    "CacheClusterId": "myreplica",
    "SnapshotStatus": "available",
    "SnapshotSource": "manual",
    "CacheNodeType": "cache.r5.large",
    "Engine": "redis",
    "EngineVersion": "5.0.5",
    "NumCacheNodes": 1,
    "PreferredAvailabilityZone": "us-west-2a",
    "CacheClusterCreateTime": "2019-11-26T00:14:52.439Z",
    "PreferredMaintenanceWindow": "sat:10:00-sat:11:00",
    "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxxx152:My_Topic",
    "Port": 6379,
    "CacheParameterGroupName": "default.redis5.0",
    "CacheSubnetGroupName": "kxkxk",
    "VpcId": "vpc-a3e97cdb",
    "AutoMinorVersionUpgrade": true,
    "SnapshotRetentionLimit": 0,
    "SnapshotWindow": "09:00-10:00",
    "NodeSnapshots": [
      {
        "CacheNodeId": "0001",
        "CacheSize": "5 MB",
        "CacheNodeCreateTime": "2019-11-26T00:14:52.439Z",

```

```

        "SnapshotCreateTime": "2019-11-26T00:25:01Z"
    }
]
},
{
    "SnapshotName": "my-cluster",
    "CacheClusterId": "my-cluster-003",
    "SnapshotStatus": "available",
    "SnapshotSource": "manual",
    "CacheNodeType": "cache.r5.large",
    "Engine": "redis",
    "EngineVersion": "5.0.5",
    "NumCacheNodes": 1,
    "PreferredAvailabilityZone": "us-west-2a",
    "CacheClusterCreateTime": "2019-11-25T23:56:17.186Z",
    "PreferredMaintenanceWindow": "sat:10:00-sat:11:00",
    "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxx152:My_Topic",
    "Port": 6379,
    "CacheParameterGroupName": "default.redis5.0",
    "CacheSubnetGroupName": "kxkxk",
    "VpcId": "vpc-a3e97cdb",
    "AutoMinorVersionUpgrade": true,
    "SnapshotRetentionLimit": 0,
    "SnapshotWindow": "09:00-10:00",
    "NodeSnapshots": [
        {
            "CacheNodeId": "0001",
            "CacheSize": "5 MB",
            "CacheNodeCreateTime": "2019-11-25T23:56:17.186Z",
            "SnapshotCreateTime": "2019-11-26T03:08:33Z"
        }
    ]
}
]
}

```

Para obtener más información, consulte [Backup and Restore ElastiCache for Redis](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia de [DescribeSnapshots](#) comandos AWS CLI .



## describe-update-actions

En el siguiente ejemplo de código, se muestra cómo usar `describe-update-actions`.

### AWS CLI

Para describir las acciones de actualización

El siguiente `describe-update-actions` ejemplo devuelve detalles de las acciones de actualización.

```
aws elasticache describe-update-actions
```

Salida:

```
{
  "UpdateActions": [
    {
      "ReplicationGroupId": "mycluster",
      "ServiceUpdateName": "elc-20191007-001",
      "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
      "ServiceUpdateSeverity": "important",
      "ServiceUpdateStatus": "available",
      "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
      "ServiceUpdateType": "security-update",
      "UpdateActionAvailableDate": "2019-12-05T19:15:19.995Z",
      "UpdateActionStatus": "complete",
      "NodesUpdated": "9/9",
      "UpdateActionStatusModifiedDate": "2019-12-05T19:15:20.461Z",
      "SlaMet": "n/a",
      "Engine": "redis"
    },
    {
      "CacheClusterId": "my-memcached-cluster",
      "ServiceUpdateName": "elc-20191007-001",
      "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
      "ServiceUpdateSeverity": "important",
      "ServiceUpdateStatus": "available",
      "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
      "ServiceUpdateType": "security-update",
      "UpdateActionAvailableDate": "2019-12-04T18:26:05.349Z",
      "UpdateActionStatus": "complete",
      "NodesUpdated": "1/1",
```

```

    "UpdateActionStatusModifiedDate": "2019-12-04T18:26:05.352Z",
    "SlaMet": "n/a",
    "Engine": "redis"
  },
  {
    "ReplicationGroupId": "my-cluster",
    "ServiceUpdateName": "elc-20191007-001",
    "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
    "ServiceUpdateSeverity": "important",
    "ServiceUpdateStatus": "available",
    "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
    "ServiceUpdateType": "security-update",
    "UpdateActionAvailableDate": "2019-11-26T03:36:26.320Z",
    "UpdateActionStatus": "complete",
    "NodesUpdated": "4/4",
    "UpdateActionStatusModifiedDate": "2019-12-04T22:11:12.664Z",
    "SlaMet": "n/a",
    "Engine": "redis"
  },
  {
    "ReplicationGroupId": "my-cluster2",
    "ServiceUpdateName": "elc-20191007-001",
    "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
    "ServiceUpdateSeverity": "important",
    "ServiceUpdateStatus": "available",
    "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
    "ServiceUpdateType": "security-update",
    "UpdateActionAvailableDate": "2019-11-26T01:26:01.617Z",
    "UpdateActionStatus": "complete",
    "NodesUpdated": "3/3",
    "UpdateActionStatusModifiedDate": "2019-11-26T01:26:01.753Z",
    "SlaMet": "n/a",
    "Engine": "redis"
  }
]
}

```

Para obtener más información, consulte [Actualizaciones de autoservicio en Amazon ElastiCache](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia de [DescribeUpdateActions](#) comandos AWS CLI .

## describe-user-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-user-groups`.

### AWS CLI

Para describir los grupos de usuarios

El siguiente `describe-user-groups` ejemplo devuelve una lista de grupos de usuarios.

```
aws elasticache describe-user-groups
```

Salida:

```
{
  "UserGroups": [
    {
      "UserGroupId": "myusergroup",
      "Status": "active",
      "Engine": "redis",
      "UserIds": [
        "default"
      ],
      "ReplicationGroups": [],
      "ARN": "arn:aws:elasticache:us-
west-2:xxxxxxxxxx52:usergroup:myusergroup"
    }
  ]
}
```

Para obtener más información, consulte [Autenticación de usuarios con el control de acceso basado en roles \(RBAC\)](#) en la Guía del usuario de Elasticache.

- Para obtener API más información, consulte la Referencia de comandos. [DescribeUserGroups](#) AWS CLI

## describe-users

En el siguiente ejemplo de código, se muestra cómo usar `describe-users`.

### AWS CLI

Para describir a los usuarios

El siguiente describe-users ejemplo devuelve una lista de usuarios.

```
aws elasticache describe-users
```

Salida:

```
{
  "Users": [
    {
      "UserId": "default",
      "UserName": "default",
      "Status": "active",
      "Engine": "redis",
      "AccessString": "on ~* +@all",
      "UserGroupIds": [
        "myusergroup"
      ],
      "Authentication": {
        "Type": "no-password"
      },
      "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:default"
    },
    {
      "UserId": "user1",
      "UserName": "myUser",
      "Status": "active",
      "Engine": "redis",
      "AccessString": "on ~* +@all",
      "UserGroupIds": [],
      "Authentication": {
        "Type": "password",
        "PasswordCount": 1
      },
      "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:user1"
    },
    {
      "UserId": "user2",
      "UserName": "myUser",
      "Status": "active",
      "Engine": "redis",
      "AccessString": "on ~app:* -@all +@read +@hash +@bitmap +@geo -setbit -
bitfield -hset -hsetnx -hmset -hincrby -hincrbyfloat -hdel -bitop -geoadd -georadius
-georadiusbymember",
```

```

        "UserGroupIds": [],
        "Authentication": {
            "Type": "password",
            "PasswordCount": 1
        },
        "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:user2"
    }
]
}

```

Para obtener más información, consulte [Autenticación de usuarios con el control de acceso basado en roles \(RBAC\)](#) en la Guía del usuario de Elasticache.

- Para obtener API más información, consulte la Referencia de comandos. [DescribeUsers](#)AWS CLI

## disassociate-global-replication-group

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-global-replication-group`.

AWS CLI

Para disociar un clúster secundario de un grupo de replicación global

El siguiente `disassociate-global-replication-group` ejemplo elimina un clúster secundario de un almacén de datos global

```

aws elasticache disassociate-global-replication-group \
  --global-replication-group-id my-grg \
  --replication-group-id my-cluster-grg-secondary \
  --replication-group-region us-east-1

```

Salida:

```

{
  "GlobalReplicationGroup": {
    "GlobalReplicationGroupId": "my-grg",
    "GlobalReplicationGroupDescription": "my-grg",
    "Status": "modifying",
    "CacheNodeType": "cache.r5.large",

```

```
"Engine": "redis",
"EngineVersion": "5.0.6",
"Members": [
  {
    "ReplicationGroupId": "my-cluster-grg-secondary",
    "ReplicationGroupRegion": "us-east-1",
    "Role": "SECONDARY",
    "AutomaticFailover": "enabled",
    "Status": "associated"
  },
  {
    "ReplicationGroupId": "my-cluster-grg",
    "ReplicationGroupRegion": "us-west-2",
    "Role": "PRIMARY",
    "AutomaticFailover": "enabled",
    "Status": "associated"
  }
],
"ClusterEnabled": false,
"AuthTokenEnabled": false,
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
```

Para obtener más información, consulte [Replicación entre AWS regiones mediante un almacén de datos global en la Guía del usuario de ElastiCache](#).

- Para obtener API más información, consulte la Referencia de comandos [DisassociateGlobalReplicationGroup](#).AWS CLI

## increase-node-groups-in-global-replication-group

En el siguiente ejemplo de código, se muestra cómo usar `increase-node-groups-in-global-replication-group`.

### AWS CLI

Para aumentar el número de grupos de nodos en un grupo de replicación global

Lo siguiente `increase-node-groups-in-global-replication-group` aumenta el número de grupos de nodos mediante el motor Redis.

```
aws elasticache increase-node-groups-in-global-replication-group \  
  --global-replication-group-id sgai-pat-test-4 \  
  --node-group-count 6 \  
  --apply-immediately
```

Salida:

```
{  
  "GlobalReplicationGroup": {  
    "GlobalReplicationGroupId": "sgai-test-4",  
    "GlobalReplicationGroupDescription": "test-4",  
    "Status": "modifying",  
    "CacheNodeType": "cache.r5.large",  
    "Engine": "redis",  
    "EngineVersion": "5.0.6",  
    "Members": [  
      {  
        "ReplicationGroupId": "my-cluster-b",  
        "ReplicationGroupRegion": "us-east-1",  
        "Role": "SECONDARY",  
        "AutomaticFailover": "enabled",  
        "Status": "associated"  
      },  
      {  
        "ReplicationGroupId": "my-cluster-a",  
        "ReplicationGroupRegion": "us-west-2",  
        "Role": "PRIMARY",  
        "AutomaticFailover": "enabled",  
        "Status": "associated"  
      }  
    ],  
    "ClusterEnabled": true,  
    "GlobalNodeGroups": [  
      {  
        "GlobalNodeGroupId": "sgai-test-4-0001",  
        "Slots": "0-234,2420-5461"  
      },  
      {  
        "GlobalNodeGroupId": "sgai-test-4-0002",  
        "Slots": "5462-5904,6997-9830"  
      },  
      {  
        "GlobalNodeGroupId": "sgai-test-4-0003",
```

```

        "Slots": "10923-11190,13375-16383"
    },
    {
        "GlobalNodeId": "sgai-test-4-0004",
        "Slots": "235-2419,5905-6996"
    },
    {
        "GlobalNodeId": "sgai-test-4-0005",
        "Slots": "9831-10922,11191-13374"
    }
],
"AuthTokenEnabled": false,
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
}

```

Para obtener más información, consulte [Replicación entre AWS regiones mediante un almacén de datos global en la Guía del usuario](#) de Elasticache.

- Para obtener API más información, consulte la Referencia de comandos [IncreaseNodeGroupsInGlobalReplicationGroup](#).AWS CLI

## increase-replica-count

En el siguiente ejemplo de código, se muestra cómo usar `increase-replica-count`.

### AWS CLI

Para aumentar el número de réplicas

El siguiente `increase-replica-count` ejemplo hace una de estas dos cosas. Puede aumentar dinámicamente el número de réplicas en un grupo de replications de Redis (modo de clúster desactivado). O puede aumentar dinámicamente el número de nodos de réplica en uno o más grupos de nodos (fragmentos) de un grupo de replicación de Redis (habilitado para el modo de clúster). Esta operación se realiza sin tiempo de inactividad del clúster.

```

aws elasticache increase-replica-count \
  --replication-group-id "my-cluster" \
  --apply-immediately \
  --new-replica-count 3

```



**Salida:**

```

{
  "ReplicationGroup": {
    "ReplicationGroupId": "my-cluster",
    "Description": " ",
    "Status": "modifying",
    "PendingModifiedValues": {},
    "MemberClusters": [
      "my-cluster-001",
      "my-cluster-002",
      "my-cluster-003",
      "my-cluster-004"
    ],
    "NodeGroups": [
      {
        "NodeGroupId": "0001",
        "Status": "modifying",
        "PrimaryEndpoint": {
          "Address": "my-
cluster.xxxxxih.ng.0001.usw2.cache.amazonaws.com",
          "Port": 6379
        },
        "ReaderEndpoint": {
          "Address": "my-cluster-
ro.xxxxxxih.ng.0001.usw2.cache.amazonaws.com",
          "Port": 6379
        },
        "NodeGroupMembers": [
          {
            "CacheClusterId": "my-cluster-001",
            "CacheNodeId": "0001",
            "ReadEndpoint": {
              "Address": "my-
cluster-001.xxxxxih.0001.usw2.cache.amazonaws.com",
              "Port": 6379
            },
            "PreferredAvailabilityZone": "us-west-2a",
            "CurrentRole": "primary"
          },
          {
            "CacheClusterId": "my-cluster-003",
            "CacheNodeId": "0001",
            "ReadEndpoint": {

```

```

        "Address": "my-
cluster-003.xxxxxih.0001.usw2.cache.amazonaws.com",
        "Port": 6379
    },
    "PreferredAvailabilityZone": "us-west-2a",
    "CurrentRole": "replica"
}
]
}
],
"AutomaticFailover": "disabled",
"SnapshotRetentionLimit": 0,
"SnapshotWindow": "07:30-08:30",
"ClusterEnabled": false,
"CacheNodeType": "cache.r5.xlarge",
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
}

```

Para obtener más información, consulte [Aumentar el número de réplicas en un fragmento en](#) la Guía del usuario de ElastiCache.

- Para obtener API más información, consulte la Referencia de comandos [IncreaseReplicaCount](#).AWS CLI

## list-allowed-node-type-modifications

En el siguiente ejemplo de código, se muestra cómo usar `list-allowed-node-type-modifications`.

### AWS CLI

Para ver una lista de las modificaciones de nodo permitidas

El siguiente `list-allowed-node-type-modifications` ejemplo muestra todos los tipos de nodos disponibles a los que puede escalar el tipo de nodo actual de su clúster o grupo de replicación de Redis.

```

aws elasticache list-allowed-node-type-modifications \
  --replication-group-id "my-replication-group"

```

**Salida:**

```
{
  "ScaleUpModifications": [
    "cache.m5.12xlarge",
    "cache.m5.24xlarge",
    "cache.m5.4xlarge",
    "cache.r5.12xlarge",
    "cache.r5.24xlarge",
    "cache.r5.2xlarge",
    "cache.r5.4xlarge"
  ],
  "ScaleDownModifications": [
    "cache.m3.large",
    "cache.m3.medium",
    "cache.m3.xlarge",
    "cache.m4.large",
    "cache.m4.xlarge",
    "cache.m5.2xlarge",
    "cache.m5.large",
    "cache.m5.xlarge",
    "cache.r3.large",
    "cache.r4.large",
    "cache.r4.xlarge",
    "cache.r5.large",
    "cache.t2.medium",
    "cache.t2.micro",
    "cache.t2.small",
    "cache.t3.medium",
    "cache.t3.micro",
    "cache.t3.small"
  ]
}
```

Para obtener más información, consulte [Scaling ElastiCache for Redis Clusters](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia de [ListAllowedNodeTypeModifications](#) comandos AWS CLI .

**list-tags-for-resource**

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

## AWS CLI

Para enumerar las etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se muestran las etiquetas de un recurso.

```
aws elasticache list-tags-for-resource \  
  --resource-name "arn:aws:elasticache:us-east-1:123456789012:cluster:my-cluster"
```

Salida:

```
{  
  "TagList": [  
    {  
      "Key": "Project",  
      "Value": "querySpeedUp"  
    },  
    {  
      "Key": "Environment",  
      "Value": "PROD"  
    }  
  ]  
}
```

Para obtener más información, consulte Cómo [enumerar las etiquetas AWS CLI en la](#) Guía del usuario de Elasticache.

- Para API obtener más información, consulte la Referencia [ListTagsForResource](#) de AWS CLI comandos.

## modify-cache-cluster

En el siguiente ejemplo de código, se muestra cómo usar `modify-cache-cluster`.

## AWS CLI

Para modificar los clústeres de caché

En el siguiente `modify-cache-cluster` ejemplo, se modifica la configuración del clúster especificado.

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id "my-cluster" \  
  --
```

```
--num-cache-nodes 1
```

Salida:

```
{
  "CacheCluster": {
    "CacheClusterId": "my-cluster",
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
    "CacheNodeType": "cache.m5.large",
    "Engine": "redis",
    "EngineVersion": "5.0.5",
    "CacheClusterStatus": "available",
    "NumCacheNodes": 1,
    "PreferredAvailabilityZone": "us-west-2c",
    "CacheClusterCreateTime": "2019-12-04T18:24:56.652Z",
    "PreferredMaintenanceWindow": "sat:10:00-sat:11:00",
    "PendingModifiedValues": {},
    "CacheSecurityGroups": [],
    "CacheParameterGroup": {
      "CacheParameterGroupName": "default.redis5.0",
      "ParameterApplyStatus": "in-sync",
      "CacheNodeIdsToReboot": []
    },
    "CacheSubnetGroupName": "default",
    "AutoMinorVersionUpgrade": true,
    "SnapshotRetentionLimit": 0,
    "SnapshotWindow": "07:00-08:00",
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false
  }
}
```

Para obtener más información, consulte [Modificación de un ElastiCache clúster](#) en la Guía del usuario de Elasticache.

- Para API obtener más información, consulte la Referencia [ModifyCacheCluster](#) de AWS CLI comandos.

## **modify-cache-parameter-group**

En el siguiente ejemplo de código, se muestra cómo usar `modify-cache-parameter-group`.

## AWS CLI

Para modificar un grupo de parámetros de caché

En el siguiente `modify-cache-parameter-group` ejemplo, se modifican los parámetros del grupo de parámetros de caché especificado.

```
aws elasticache modify-cache-parameter-group \  
  --cache-parameter-group-name "mygroup" \  
  --parameter-name-values "ParameterName=activedefrag, ParameterValue=no"
```

Salida:

```
{  
  "CacheParameterGroupName": "mygroup"  
}
```

Para obtener más información, consulte [Modificación de un grupo de parámetros](#) en la Guía del usuario de Elasticache.

- Para API obtener más información, consulte la Referencia [ModifyCacheParameterGroup](#) de AWS CLI comandos.

## `modify-cache-subnet-group`

En el siguiente ejemplo de código, se muestra cómo usar `modify-cache-subnet-group`.

## AWS CLI

Para modificar un grupo de subredes de caché

El siguiente `modify-cache-subnet-group` ejemplo modifica el grupo de subredes de caché especificado.

```
aws elasticache modify-cache-subnet-group \  
  --cache-subnet-group-name kxkxk \  
  --cache-subnet-group-description "mygroup"
```

Salida:

```
{  
  "CacheSubnetGroup": {
```

```

    "CacheSubnetGroupName": "kxkxk",
    "CacheSubnetGroupDescription": "mygroup",
    "VpcId": "vpc-xxxxcdb",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-xxxxbff",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2a"
        }
      }
    ]
  }
}

```

Para obtener más información, consulte [Modificación de un grupo de subredes](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia de [ModifyCacheSubnetGroup](#) comandos AWS CLI .

## modify-global-replication-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-global-replication-group`.

### AWS CLI

Para modificar un grupo de replicación global

A continuación, se `modify-global-replication-group` modifican las propiedades de un grupo de replicación global y, en este caso, se deshabilita la conmutación por error automática mediante el motor Redis.

```

aws elasticache modify-global-replication-group \
  --global-replication-group-id sgai-pat-group \
  --apply-immediately \
  --no-automatic-failover-enabled

```

### Salida

```

{
  "GlobalReplicationGroup": {
    "GlobalReplicationGroupId": "sgai-test-group",

```

```
    "GlobalReplicationGroupDescription": " ",
    "Status": "modifying",
    "CacheNodeType": "cache.r5.large",
    "Engine": "redis",
    "EngineVersion": "5.0.6",
    "ClusterEnabled": false,
    "AuthTokenEnabled": false,
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false
  }
}
```

Para obtener más información, consulte [Replicación entre AWS regiones mediante un almacén de datos global en la Guía del usuario de ElastiCache](#).

- Para obtener API más información, consulte la Referencia de comandos [ModifyGlobalReplicationGroup](#).AWS CLI

## modify-replication-group-shard-configuration

En el siguiente ejemplo de código, se muestra cómo usar `modify-replication-group-shard-configuration`.

### AWS CLI

Para modificar la configuración de un fragmento de un grupo de replicación

Lo siguiente `modify-replication-group-shard-configuration` reduce el recuento de grupos de nodos mediante el motor Redis.

```
aws elasticache modify-replication-group-shard-configuration \
  --replication-group-id mycluster \
  --node-group-count 3 \
  --apply-immediately \
  --node-groups-to-remove 0002
```

### Salida

```
{
  "ReplicationGroup": {
    "ReplicationGroupId": "mycluster",
    "Description": "mycluster",
```



```
"GlobalReplicationGroupInfo": {},
"Status": "modifying",
"PendingModifiedValues": {},
"MemberClusters": [
  "mycluster-0002-001",
  "mycluster-0002-002",
  "mycluster-0002-003",
  "mycluster-0003-001",
  "mycluster-0003-002",
  "mycluster-0003-003",
  "mycluster-0003-004",
  "mycluster-0004-001",
  "mycluster-0004-002",
  "mycluster-0004-003",
  "mycluster-0005-001",
  "mycluster-0005-002",
  "mycluster-0005-003"
],
"NodeGroups": [
  {
    "NodeGroupId": "0002",
    "Status": "modifying",
    "Slots": "894-1767,3134-4443,5149-5461,6827-7332,12570-13662",
    "NodeGroupMembers": [
      {
        "CacheClusterId": "mycluster-0002-001",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2c"
      },
      {
        "CacheClusterId": "mycluster-0002-002",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2a"
      },
      {
        "CacheClusterId": "mycluster-0002-003",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2b"
      }
    ]
  },
  {
    "NodeGroupId": "0003",
    "Status": "modifying",
```

```

    "Slots":
    "0-324,5462-5692,6784-6826,7698-8191,10923-11075,12441-12569,13663-16383",
    "NodeGroupMembers": [
      {
        "CacheClusterId": "mycluster-0003-001",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2c"
      },
      {
        "CacheClusterId": "mycluster-0003-002",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2b"
      },
      {
        "CacheClusterId": "mycluster-0003-003",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2a"
      },
      {
        "CacheClusterId": "mycluster-0003-004",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2c"
      }
    ]
  },
  {
    "NodeGroupId": "0004",
    "Status": "modifying",
    "Slots": "325-336,4706-5148,7333-7697,9012-10922,11076-12440",
    "NodeGroupMembers": [
      {
        "CacheClusterId": "mycluster-0004-001",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2b"
      },
      {
        "CacheClusterId": "mycluster-0004-002",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2a"
      },
      {
        "CacheClusterId": "mycluster-0004-003",
        "CacheNodeId": "0001",
        "PreferredAvailabilityZone": "us-west-2c"
      }
    ]
  }
]

```

```

    }
  ]
},
{
  "NodeGroupId": "0005",
  "Status": "modifying",
  "Slots": "337-893,1768-3133,4444-4705,5693-6783,8192-9011",
  "NodeGroupMembers": [
    {
      "CacheClusterId": "mycluster-0005-001",
      "CacheNodeId": "0001",
      "PreferredAvailabilityZone": "us-west-2a"
    },
    {
      "CacheClusterId": "mycluster-0005-002",
      "CacheNodeId": "0001",
      "PreferredAvailabilityZone": "us-west-2c"
    },
    {
      "CacheClusterId": "mycluster-0005-003",
      "CacheNodeId": "0001",
      "PreferredAvailabilityZone": "us-west-2b"
    }
  ]
}
],
"AutomaticFailover": "enabled",
"MultiAZ": "enabled",
"ConfigurationEndpoint": {
  "Address": "mycluster.g2xbih.clustercfg.usw2.cache.amazonaws.com",
  "Port": 6379
},
"SnapshotRetentionLimit": 1,
"SnapshotWindow": "13:00-14:00",
"ClusterEnabled": true,
"CacheNodeType": "cache.r5.xlarge",
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
}

```

Para obtener más información, consulte [Scaling ElastiCache for Redis Clusters](#) en la Guía del usuario de Elasticache.

- Para API obtener más información, consulte la Referencia de [ModifyReplicationGroupShardConfiguration](#) comandos AWS CLI .

## modify-replication-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-replication-group`.

### AWS CLI

Para modificar un grupo de replicación

Lo siguiente `modify-replication-group` desactiva la Multi-AZ mediante el motor Redis.

```
aws elasticache modify-replication-group \  
  --replication-group-id test-cluster \  
  --no-multi-az-enabled \  
  --apply-immediately
```

### Salida

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "test-cluster",  
    "Description": "test-cluster",  
    "GlobalReplicationGroupInfo": {  
      "GlobalReplicationGroupId": "sgaui-pat-group",  
      "GlobalReplicationGroupMemberRole": "PRIMARY"  
    },  
    "Status": "available",  
    "PendingModifiedValues": {},  
    "MemberClusters": [  
      "test-cluster-001",  
      "test-cluster-002",  
      "test-cluster-003"  
    ],  
    "NodeGroups": [  
      {  
        "NodeGroupId": "0001",  
        "Status": "available",  
        "PrimaryEndpoint": {  
          "Address": "test-  
cluster.g2xbih.ng.0001.usw2.cache.amazonaws.com",  
          "Port": 6379
```

```
    },
    "ReaderEndpoint": {
      "Address": "test-cluster-
ro.g2xbih.ng.0001.usw2.cache.amazonaws.com",
      "Port": 6379
    },
    },
    "NodeGroupMembers": [
      {
        "CacheClusterId": "test-cluster-001",
        "CacheNodeId": "0001",
        "ReadEndpoint": {
          "Address": "test-
cluster-001.g2xbih.0001.usw2.cache.amazonaws.com",
          "Port": 6379
        },
        "PreferredAvailabilityZone": "us-west-2c",
        "CurrentRole": "primary"
      },
      {
        "CacheClusterId": "test-cluster-002",
        "CacheNodeId": "0001",
        "ReadEndpoint": {
          "Address": "test-
cluster-002.g2xbih.0001.usw2.cache.amazonaws.com",
          "Port": 6379
        },
        "PreferredAvailabilityZone": "us-west-2b",
        "CurrentRole": "replica"
      },
      {
        "CacheClusterId": "test-cluster-003",
        "CacheNodeId": "0001",
        "ReadEndpoint": {
          "Address": "test-
cluster-003.g2xbih.0001.usw2.cache.amazonaws.com",
          "Port": 6379
        },
        "PreferredAvailabilityZone": "us-west-2a",
        "CurrentRole": "replica"
      }
    ]
  }
},
"SnapshottingClusterId": "test-cluster-002",
```

```

    "AutomaticFailover": "enabled",
    "MultiAZ": "disabled",
    "SnapshotRetentionLimit": 1,
    "SnapshotWindow": "08:00-09:00",
    "ClusterEnabled": false,
    "CacheNodeType": "cache.r5.large",
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false
  }
}

```

Para obtener más información, consulte [Modificación de un grupo de replicación](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia [ModifyReplicationGroup](#) de AWS CLI comandos.

## modify-user-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-user-group`.

### AWS CLI

Para modificar un grupo de usuarios

En el siguiente `modify-user-group` ejemplo, se agrega un usuario al grupo de usuarios.

```

aws elasticache modify-user-group \
  --user-group-id myusergroup \
  --user-ids-to-add user1

```

Salida:

```

{
  "UserGroupId": "myusergroup",
  "Status": "modifying",
  "Engine": "redis",
  "UserIds": [
    "default"
  ],
  "PendingChanges": {

```

```

    "UserIdsToAdd": [
      "user1"
    ],
    "ReplicationGroups": [],
    "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:usergroup:myusergroup"
  }

```

Para obtener más información, consulte [Autenticación de usuarios con el control de acceso basado en roles \(RBAC\)](#) en la Guía del usuario de Elasticache.

- Para obtener API más información, consulte la Referencia de comandos. [ModifyUserGroup](#) AWS CLI

## modify-user

En el siguiente ejemplo de código, se muestra cómo usar `modify-user`.

### AWS CLI

Para modificar un usuario

El siguiente `modify-user` ejemplo modifica la cadena de acceso de un usuario.

```

aws elasticache modify-user \
  --user-id user2 \
  --append-access-string "on ~* +@all"

```

Salida:

```

{
  "UserId": "user2",
  "UserName": "myUser",
  "Status": "modifying",
  "Engine": "redis",
  "AccessString": "on ~* +@all",
  "UserGroupIds": [],
  "Authentication": {
    "Type": "password",
    "PasswordCount": 1
  },
  "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:user2"
}

```

```
}
```

Para obtener más información, consulte [Autenticación de usuarios con el control de acceso basado en roles \(RBAC\)](#) en la Guía del usuario de ElastiCache.

- Para obtener API más información, consulte la Referencia de comandos. [ModifyUser](#) AWS CLI

## **purchase-reserved-cache-nodes-offering**

En el siguiente ejemplo de código, se muestra cómo usar `purchase-reserved-cache-nodes-offering`.

### AWS CLI

Para comprar un `reserved-cache-node-offering`

El siguiente `purchase-reserved-cache-nodes-offering` ejemplo le permite comprar una oferta de nodos de caché reservados.

```
aws elasticache purchase-reserved-cache-nodes-offering \  
  --reserved-cache-nodes-offering-id xxxxxxxx-4da5-4b90-b92d-929fbd7abed2
```

### Salida

```
{  
  "ReservedCacheNode": {  
    "ReservedCacheNodeId": "ri-2020-06-30-17-59-40-474",  
    "ReservedCacheNodesOfferingId": "xxxxxxx-4da5-4b90-b92d-929fbd7abed2",  
    "CacheNodeType": "cache.m3.2xlarge",  
    "StartTime": "2020-06-30T17:59:40.474000+00:00",  
    "Duration": 31536000,  
    "FixedPrice": 1772.0,  
    "UsagePrice": 0.0,  
    "CacheNodeCount": 1,  
    "ProductDescription": "redis",  
    "OfferingType": "Heavy Utilization",  
    "State": "payment-pending",  
    "RecurringCharges": [  
      {  
        "RecurringChargeAmount": 0.25,  
        "RecurringChargeFrequency": "Hourly"  
      }  
    ]  
  }  
}
```



```

    ]
  }
}

```

Para obtener más información, consulte [Obtener información sobre las ofertas de nodos reservados](#) en la Guía del usuario de Elasticache Redis o [Obtener información sobre las ofertas de nodos reservados](#) en la Guía del usuario de Elasticache Memcached.

- Para API obtener más información, consulte la Referencia de comandos. [PurchaseReservedCacheNodesOffering](#) AWS CLI

## reboot-cache-cluster

En el siguiente ejemplo de código, se muestra cómo usar `reboot-cache-cluster`.

### AWS CLI

Para reiniciar un clúster de caché

En el siguiente `reboot-cache-cluster` ejemplo, se reinician algunos o todos los nodos de caché de un clúster aprovisionado. Esta operación aplica cualquier grupo de parámetros de caché modificado al clúster. La operación de reinicio se lleva a cabo lo antes posible y provoca una interrupción momentánea del clúster. Durante el reinicio, el estado del clúster se establece en REBOOTING

```

aws elasticache reboot-cache-cluster \
  --cache-cluster-id "my-cluster-001" \
  --cache-node-ids-to-reboot "0001"

```

Salida:

```

{
  "CacheCluster": {
    "CacheClusterId": "my-cluster-001",
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/home#client-download:",
    "CacheNodeType": "cache.r5.xlarge",
    "Engine": "redis",
    "EngineVersion": "5.0.5",
    "CacheClusterStatus": "rebooting cache cluster nodes",
    "NumCacheNodes": 1,
  }
}

```

```

    "PreferredAvailabilityZone": "us-west-2a",
    "CacheClusterCreateTime": "2019-11-26T03:35:04.546Z",
    "PreferredMaintenanceWindow": "mon:04:05-mon:05:05",
    "PendingModifiedValues": {},
    "NotificationConfiguration": {
      "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxx152:My_Topic",
      "TopicStatus": "active"
    },
    "CacheSecurityGroups": [],
    "CacheParameterGroup": {
      "CacheParameterGroupName": "mygroup",
      "ParameterApplyStatus": "in-sync",
      "CacheNodeIdsToReboot": []
    },
    "CacheSubnetGroupName": "kxkxk",
    "AutoMinorVersionUpgrade": true,
    "SecurityGroups": [
      {
        "SecurityGroupId": "sg-xxxxxxxxxxxx836",
        "Status": "active"
      },
      {
        "SecurityGroupId": "sg-xxxxxxx7b",
        "Status": "active"
      }
    ],
    "ReplicationGroupId": "my-cluster",
    "SnapshotRetentionLimit": 0,
    "SnapshotWindow": "07:30-08:30",
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false
  }
}

```

Para obtener más información, consulte Reiniciar un clúster < <https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/clusters.rebooting.html> en la Guía del usuario de ElastiCache.

- Para obtener API más información, [RebootCacheCluster](#) consulte AWS CLI la Referencia de comandos.

## reset-cache-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `reset-cache-parameter-group`.

## AWS CLI

Para restablecer un grupo de parámetros de caché

En el siguiente `reset-cache-parameter-group` ejemplo, se modifican los parámetros de un grupo de parámetros de caché al valor predeterminado del motor o del sistema. Puede restablecer parámetros específicos enviando una lista de nombres de parámetros. Para restablecer todo el grupo de parámetros de caché, especifique los `--cache-parameter-group-name` parámetros `--reset-all-parameters` y.

```
aws elasticache reset-cache-parameter-group \  
  --cache-parameter-group-name "mygroup" \  
  --reset-all-parameters
```

Salida:

```
{  
  "CacheParameterGroupName": "mygroup"  
}
```

- Para API obtener más información, consulte [ResetCacheParameterGroup](#) la Referencia de AWS CLI comandos.

## start-migration

En el siguiente ejemplo de código, se muestra cómo usar `start-migration`.

## AWS CLI

Para iniciar una migración

A continuación, `start-migration` se migran los datos de Redis autohospedado en Amazon a EC2 Amazon ElastiCache mediante el motor de Redis.

```
aws elasticache start-migration \  
  --replication-group-id test \  
  --customer-node-endpoint-  
  list "Address='test.g2xbih.ng.0001.usw2.cache.amazonaws.com',Port=6379"
```

Salida

```
{
  "ReplicationGroup": {
    "ReplicationGroupId": "test",
    "Description": "test",
    "GlobalReplicationGroupInfo": {},
    "Status": "modifying",
    "PendingModifiedValues": {},
    "MemberClusters": [
      "test-001",
      "test-002",
      "test-003"
    ],
    "NodeGroups": [
      {
        "NodeGroupId": "0001",
        "Status": "available",
        "PrimaryEndpoint": {
          "Address": "test.g2xbih.ng.0001.usw2.cache.amazonaws.com",
          "Port": 6379
        },
        "ReaderEndpoint": {
          "Address": "test-ro.g2xbih.ng.0001.usw2.cache.amazonaws.com",
          "Port": 6379
        },
        "NodeGroupMembers": [
          {
            "CacheClusterId": "test-001",
            "CacheNodeId": "0001",
            "ReadEndpoint": {
              "Address":
"test-001.g2xbih.0001.usw2.cache.amazonaws.com",
              "Port": 6379
            },
            "PreferredAvailabilityZone": "us-west-2a",
            "CurrentRole": "primary"
          },
          {
            "CacheClusterId": "test-002",
            "CacheNodeId": "0001",
            "ReadEndpoint": {
              "Address":
"test-002.g2xbih.0001.usw2.cache.amazonaws.com",
              "Port": 6379
            }
          }
        ]
      }
    ]
  }
}
```

```

    },
    "PreferredAvailabilityZone": "us-west-2c",
    "CurrentRole": "replica"
  },
  {
    "CacheClusterId": "test-003",
    "CacheNodeId": "0001",
    "ReadEndpoint": {
      "Address":
"test-003.g2xbih.0001.usw2.cache.amazonaws.com",
      "Port": 6379
    },
    "PreferredAvailabilityZone": "us-west-2b",
    "CurrentRole": "replica"
  }
]
}
],
"SnapshottingClusterId": "test-002",
"AutomaticFailover": "enabled",
"MultiAZ": "enabled",
"SnapshotRetentionLimit": 1,
"SnapshotWindow": "07:30-08:30",
"ClusterEnabled": false,
"CacheNodeType": "cache.r5.large",
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
}

```

Para obtener más información, consulte [Migración en línea a ElastiCache](#) en la Guía del usuario de ElastiCache.

- Para API obtener más información, consulte la Referencia [StartMigration](#) de AWS CLI comandos.

## test-failover

En el siguiente ejemplo de código, se muestra cómo usar test-failover.

### AWS CLI

Para probar la conmutación por error de un grupo de nodos

El siguiente `test-failover` ejemplo prueba la conmutación por error automática en el grupo de nodos especificado (denominado fragmento en la consola) de un grupo de replicación (denominado clúster en la consola).

```
aws elasticache test-failover /  
  --replication-group-id "mycluster" /  
  --node-group-id "0001"
```

Salida:

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "mycluster",  
    "Description": "My Cluster",  
    "Status": "available",  
    "PendingModifiedValues": {},  
    "MemberClusters": [  
      "mycluster-0001-001",  
      "mycluster-0001-002",  
      "mycluster-0001-003",  
      "mycluster-0002-001",  
      "mycluster-0002-002",  
      "mycluster-0002-003",  
      "mycluster-0003-001",  
      "mycluster-0003-002",  
      "mycluster-0003-003"  
    ],  
    "NodeGroups": [  
      {  
        "NodeGroupId": "0001",  
        "Status": "available",  
        "Slots": "0-5461",  
        "NodeGroupMembers": [  
          {  
            "CacheClusterId": "mycluster-0001-001",  
            "CacheNodeId": "0001",  
            "PreferredAvailabilityZone": "us-west-2b"  
          },  
          {  
            "CacheClusterId": "mycluster-0001-002",  
            "CacheNodeId": "0001",  
            "PreferredAvailabilityZone": "us-west-2a"  
          }  
        ]  
      }  
    ]  
  }  
}
```

```
        {
            "CacheClusterId": "mycluster-0001-003",
            "CacheNodeId": "0001",
            "PreferredAvailabilityZone": "us-west-2c"
        }
    ]
},
{
    "NodeGroupId": "0002",
    "Status": "available",
    "Slots": "5462-10922",
    "NodeGroupMembers": [
        {
            "CacheClusterId": "mycluster-0002-001",
            "CacheNodeId": "0001",
            "PreferredAvailabilityZone": "us-west-2a"
        },
        {
            "CacheClusterId": "mycluster-0002-002",
            "CacheNodeId": "0001",
            "PreferredAvailabilityZone": "us-west-2b"
        },
        {
            "CacheClusterId": "mycluster-0002-003",
            "CacheNodeId": "0001",
            "PreferredAvailabilityZone": "us-west-2c"
        }
    ]
},
{
    "NodeGroupId": "0003",
    "Status": "available",
    "Slots": "10923-16383",
    "NodeGroupMembers": [
        {
            "CacheClusterId": "mycluster-0003-001",
            "CacheNodeId": "0001",
            "PreferredAvailabilityZone": "us-west-2c"
        },
        {
            "CacheClusterId": "mycluster-0003-002",
            "CacheNodeId": "0001",
            "PreferredAvailabilityZone": "us-west-2b"
        }
    ]
},
```

```
        {
            "CacheClusterId": "mycluster-0003-003",
            "CacheNodeId": "0001",
            "PreferredAvailabilityZone": "us-west-2a"
        }
    ]
}
],
"AutomaticFailover": "enabled",
"ConfigurationEndpoint": {
    "Address": "mycluster.xxxxih.clustercfg.usw2.cache.amazonaws.com",
    "Port": 6379
},
"SnapshotRetentionLimit": 1,
"SnapshotWindow": "13:00-14:00",
"ClusterEnabled": true,
"CacheNodeType": "cache.r5.large",
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false
}
}
```

- Para API obtener más información, consulte la Referencia [TestFailover](#) de AWS CLI comandos.

## MediaStore ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with MediaStore.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)



## Acciones

### **create-container**

En el siguiente ejemplo de código, se muestra cómo usar `create-container`.

#### AWS CLI

Para crear un contenedor

En el siguiente `create-container` ejemplo, se crea un contenedor nuevo y vacío.

```
aws mediastore create-container --container-name ExampleContainer
```

Salida:

```
{
  "Container": {
    "AccessLoggingEnabled": false,
    "CreationTime": 1563557265,
    "Name": "ExampleContainer",
    "Status": "CREATING",
    "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer"
  }
}
```

Para obtener más información, consulte [Creación de un contenedor](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [CreateContainer](#) la Referencia de AWS CLI comandos.

### **delete-container-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-container-policy`.

#### AWS CLI

Para eliminar una política de contenedores

En el siguiente `delete-container-policy` ejemplo, se elimina la política que está asignada al contenedor especificado. Cuando se elimina la política, AWS Elemental asigna MediaStore automáticamente la política predeterminada al contenedor.

```
aws mediastore delete-container-policy \  
  --container-name LiveEvents
```

Este comando no genera ninguna salida.

Para obtener más información, consulta la MediaStore API referencia [DeleteContainerPolicy](#) de AWS Elemental.

- Para API obtener más información, consulte [DeleteContainerPolicy](#) la Referencia de AWS CLI comandos.

## **delete-container**

En el siguiente ejemplo de código, se muestra cómo usar `delete-container`.

### AWS CLI

Para eliminar un contenedor

En el siguiente `delete-container` ejemplo, se elimina el contenedor especificado. Un contenedor únicamente se puede eliminar si no tiene objetos.

```
aws mediastore delete-container \  
  --container-name=ExampleLiveDemo
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un contenedor](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [DeleteContainer](#) la Referencia de AWS CLI comandos.

## **delete-cors-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-cors-policy`.

## AWS CLI

Para eliminar una CORS política

En el siguiente `delete-cors-policy` ejemplo, se elimina la política de intercambio de recursos entre orígenes (CORS) que está asignada al contenedor especificado.

```
aws mediastore delete-cors-policy \  
  --container-name ExampleContainer
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar una CORS política](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [DeleteCorsPolicy](#) la Referencia de AWS CLI comandos.

## `delete-lifecycle-policy`

En el siguiente ejemplo de código, se muestra cómo usar `delete-lifecycle-policy`.

## AWS CLI

Para eliminar una política de ciclo de vida de un objeto

En el siguiente `delete-lifecycle-policy` ejemplo, se elimina la política de ciclo de vida de los objetos adjunta al contenedor especificado. Este cambio puede tardar hasta 20 minutos en surtir efecto.

```
aws mediastore delete-lifecycle-policy \  
  --container-name LiveEvents
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar la política de ciclo de vida de un objeto](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [DeleteLifecyclePolicy](#) la Referencia de AWS CLI comandos.

## describe-container

En el siguiente ejemplo de código, se muestra cómo usar `describe-container`.

### AWS CLI

Para ver los detalles de un contenedor

En el siguiente `describe-container` ejemplo, se muestran los detalles del contenedor especificado.

```
aws mediastore describe-container \  
  --container-name ExampleContainer
```

Salida:

```
{  
  "Container": {  
    "CreationTime": 1563558086,  
    "AccessLoggingEnabled": false,  
    "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/  
ExampleContainer",  
    "Status": "ACTIVE",  
    "Name": "ExampleContainer",  
    "Endpoint": "https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com"  
  }  
}
```

Para obtener más información, consulte [Visualización de los detalles de un contenedor](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [DescribeContainer](#) la Referencia de AWS CLI comandos.

## describe-object

En el siguiente ejemplo de código, se muestra cómo usar `describe-object`.

### AWS CLI

Para ver una lista de objetos y carpetas en un contenedor específico

En el siguiente describe-object ejemplo, se muestran los elementos (objetos y carpetas) almacenados en un contenedor específico.

```
aws mediastore-data describe-object \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --path /folder_name/file1234.jpg
```

Salida:

```
{  
  "ContentType": "image/jpeg",  
  "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",  
  "ContentLength": "2307346",  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f5555555555555555da6d3"  
}
```

Para obtener más información, consulte [Visualización de los detalles de un objeto](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [DescribeObject](#) la Referencia de AWS CLI comandos.

## get-container-policy

En el siguiente ejemplo de código, se muestra cómo usar get-container-policy.

AWS CLI

Para ver una política de contenedores

En el siguiente get-container-policy ejemplo, se muestra la política basada en recursos del contenedor especificado.

```
aws mediastore get-container-policy \  
  --container-name ExampleLiveDemo
```

Salida:

```
{  
  "Policy": {
```

```

    "Version": "2012-10-17",
    "Statement": [
      {
        "Sid": "PublicReadOverHttps",
        "Effect": "Allow",
        "Principal": {
          "AWS": "arn:aws:iam::111122223333:root"
        },
        "Action": [
          "mediastore:GetObject",
          "mediastore:DescribeObject"
        ],
        "Resource": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleLiveDemo/",
        "Condition": {
          "Bool": {
            "aws:SecureTransport": "true"
          }
        }
      }
    ]
  }
}

```

Para obtener más información, consulte [Visualización de una política de contenedores](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [GetContainerPolicy](#) la Referencia de AWS CLI comandos.

## get-cors-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-cors-policy`.

### AWS CLI

Para ver una CORS política

En el siguiente `get-cors-policy` ejemplo, se muestra la política de uso compartido de recursos (CORS) entre orígenes que está asignada al contenedor especificado.

```

aws mediastore get-cors-policy \
  --container-name ExampleContainer \

```

```
--region us-west-2
```

Salida:

```
{
  "CorsPolicy": [
    {
      "AllowedMethods": [
        "GET",
        "HEAD"
      ],
      "MaxAgeSeconds": 3000,
      "AllowedOrigins": [
        ""
      ],
      "AllowedHeaders": [
        ""
      ]
    }
  ]
}
```

Para obtener más información, consulte [Visualización de una CORS política](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [GetCorsPolicy](#) la Referencia de AWS CLI comandos.

## get-lifecycle-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-lifecycle-policy`.

AWS CLI

Para ver la política de ciclo de vida de un objeto

En el siguiente `get-lifecycle-policy` ejemplo, se muestra la política de ciclo de vida de los objetos adjunta al contenedor especificado.

```
aws mediastore get-lifecycle-policy \  
  --container-name LiveEvents
```

**Salida:**

```
{
  "LifecyclePolicy": {
    "rules": [
      {
        "definition": {
          "path": [
            {
              "prefix": "Football/"
            },
            {
              "prefix": "Baseball/"
            }
          ],
          "days_since_create": [
            {
              "numeric": [
                ">",
                28
              ]
            }
          ],
          "action": "EXPIRE"
        }
      ]
    }
  }
}
```

Para obtener más información, consulte [Visualización de una política de ciclo de vida de objetos](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [GetLifecyclePolicy](#) la Referencia de AWS CLI comandos.

**get-object**

En el siguiente ejemplo de código, se muestra cómo usar `get-object`.

**AWS CLI**

Para descargar un objeto



En el siguiente `get-object` ejemplo, se descarga un objeto en el punto final especificado.

```
aws mediastore-data get-object \  
  --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --path=/folder_name/README.md README.md
```

Salida:

```
{  
  "ContentLength": "2307346",  
  "ContentType": "image/jpeg",  
  "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9e4dd89ff7f5555555555555555da6d3",  
  "StatusCode": 200  
}
```

Para descargar parte de un objeto

En el siguiente `get-object` ejemplo, se descarga una parte de un objeto en el punto final especificado.

```
aws mediastore-data get-object \  
  --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --path /folder_name/README.md \  
  --range="bytes=0-100" README2.md
```

Salida:

```
{  
  "StatusCode": 206,  
  "ContentRange": "bytes 0-100/2307346",  
  "ContentLength": "101",  
  "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",  
  "ContentType": "image/jpeg",  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9e4dd89ff7f5555555555555555da6d3"  
}
```

Para obtener más información, consulte [Descarga de un objeto](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [GetObject](#) la Referencia de AWS CLI comandos.

## list-containers

En el siguiente ejemplo de código, se muestra cómo usar `list-containers`.

### AWS CLI

Para ver una lista de contenedores

En el siguiente `list-containers` ejemplo, se muestra una lista de todos los contenedores que están asociados a su cuenta.

```
aws mediastore list-containers
```

Salida:

```
{
  "Containers": [
    {
      "CreationTime": 1505317931,
      "Endpoint": "https://aaabbbcccddee.data.mediastore.us-
west-2.amazonaws.com",
      "Status": "ACTIVE",
      "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleLiveDemo",
      "AccessLoggingEnabled": false,
      "Name": "ExampleLiveDemo"
    },
    {
      "CreationTime": 1506528818,
      "Endpoint": "https://fffggghhhiiijj.data.mediastore.us-
west-2.amazonaws.com",
      "Status": "ACTIVE",
      "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer",
      "AccessLoggingEnabled": false,
      "Name": "ExampleContainer"
    }
  ]
}
```

Para obtener más información, consulte [Visualización de una lista de contenedores](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [ListContainers](#) la Referencia de AWS CLI comandos.

## list-items

En el siguiente ejemplo de código, se muestra cómo usar `list-items`.

### AWS CLI

Ejemplo 1: Para ver una lista de objetos y carpetas en un contenedor específico

En el siguiente `list-items` ejemplo, se muestran los elementos (objetos y carpetas) almacenados en el contenedor especificado.

```
aws mediastore-data list-items \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com
```

Salida:

```
{  
  "Items": [  
    {  
      "ContentType": "image/jpeg",  
      "LastModified": 1563571859.379,  
      "Name": "filename.jpg",  
      "Type": "OBJECT",  
      "ETag":  
"543ab21abcd1a234ab123456a1a2b12345ab12abc12a1234abc1a2bc12345a12",  
      "ContentLength": 3784  
    },  
    {  
      "Type": "FOLDER",  
      "Name": "ExampleLiveDemo"  
    }  
  ]  
}
```

Ejemplo 2: Para ver una lista de objetos y carpetas de una carpeta específica

En el siguiente `list-items` ejemplo, se muestran los elementos (objetos y carpetas) almacenados en una carpeta específica.

```
aws mediastore-data list-items \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com
```

Salida:

```
{  
  "Items": [  
    {  
      "ContentType": "image/jpeg",  
      "LastModified": 1563571859.379,  
      "Name": "filename.jpg",  
      "Type": "OBJECT",  
      "ETag":  
"543ab21abcd1a234ab123456a1a2b12345ab12abc12a1234abc1a2bc12345a12",  
      "ContentLength": 3784  
    },  
    {  
      "Type": "FOLDER",  
      "Name": "ExampleLiveDemo"  
    }  
  ]  
}
```

Para obtener más información, consulte [Visualización de una lista de objetos](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [ListItems](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

AWS CLI

Para enumerar las etiquetas de un contenedor

En el siguiente `list-tags-for-resource` ejemplo, se muestran las claves y los valores de las etiquetas asignados al contenedor especificado.

```
aws mediastore list-tags-for-resource \  
  --resource arn:aws:mediastore:us-west-2:1213456789012:container/ExampleContainer
```

Salida:

```
{
  "Tags": [
    {
      "Value": "Test",
      "Key": "Environment"
    },
    {
      "Value": "West",
      "Key": "Region"
    }
  ]
}
```

Para obtener más información, consulte [ListTagsForResource](#) la MediaStore API Referencia AWS elemental.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## put-container-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-container-policy`.

### AWS CLI

Para editar una política de contenedores

En el siguiente `put-container-policy` ejemplo, se asigna una política diferente al contenedor especificado. En este ejemplo, la política actualizada se define en un archivo denominado `LiveEventsContainerPolicy.json`.

```
aws mediastore put-container-policy \
  --container-name LiveEvents \
  --policy file://LiveEventsContainerPolicy.json
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Edición de una política de contenedores](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [PutContainerPolicy](#) la Referencia de AWS CLI comandos.

## put-cors-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-cors-policy`.

### AWS CLI

Ejemplo 1: Para añadir una CORS política

En el siguiente `put-cors-policy` ejemplo, se agrega una política de uso compartido de recursos (CORS) entre orígenes al contenedor especificado. El contenido de la CORS política se encuentra en el archivo denominado `corsPolicy.json`.

```
aws mediastore put-cors-policy \  
  --container-name ExampleContainer \  
  --cors-policy file://corsPolicy.json
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir una CORS política a un contenedor](#) en la Guía del MediaStore usuario de AWS Elemental.

Ejemplo 2: Para editar una CORS política

En el siguiente `put-cors-policy` ejemplo, se actualiza la política de intercambio de recursos entre orígenes (CORS) que está asignada al contenedor especificado. El contenido de la CORS política actualizada se encuentra en el archivo denominado `corsPolicy2.json`.

Para obtener más información, consulte [Edición de una CORS política](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [PutCorsPolicy](#) la Referencia de AWS CLI comandos.

## put-lifecycle-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-lifecycle-policy`.

## AWS CLI

Para crear una política de ciclo de vida de los objetos

En el siguiente `put-lifecycle-policy` ejemplo, se adjunta una política de ciclo de vida de los objetos al contenedor especificado. Esto le permite especificar durante cuánto tiempo debe almacenar el servicio los objetos en su contenedor. MediaStore elimina los objetos del contenedor una vez que llegan a su fecha de caducidad, tal y como se indica en la política, que se encuentra en el nombre `LiveEventsLifecyclePolicy.json` del archivo.

```
aws mediastore put-lifecycle-policy \  
  --container-name ExampleContainer \  
  --lifecycle-policy file://ExampleLifecyclePolicy.json
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir una política de ciclo de vida de objetos a un contenedor](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [PutLifecyclePolicy](#) la Referencia de AWS CLI comandos.

## put-object

En el siguiente ejemplo de código, se muestra cómo usar `put-object`.

## AWS CLI

Para cargar un objeto

En el siguiente `put-object` ejemplo, se carga un objeto en el contenedor especificado. Puede especificar una ruta de carpeta en la que se guardará el objeto dentro del contenedor. Si la carpeta ya existe, AWS Elemental MediaStore guarda el objeto en la carpeta. Si la carpeta no existe, el servicio la crea y, a continuación, almacena el objeto en la carpeta.

```
aws mediastore-data put-object \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --body README.md \  
  --path /folder_name/README.md \  
  --cache-control "max-age=6, public" \  
  --
```

```
--content-type binary/octet-stream
```

Salida:

```
{
  "ContentSHA256":
  "74b5fdb517f423ed750ef214c44adfe2be36e37d861eafe9c842cbe1bf387a9d",
  "StorageClass": "TEMPORAL",
  "ETag": "af3e4731af032167a106015d1f2fe934e68b32ed1aa297a9e325f5c64979277b"
}
```

Para obtener más información, consulte [Carga de un objeto](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [PutObject](#) la Referencia de AWS CLI comandos.

## start-access-logging

En el siguiente ejemplo de código, se muestra cómo usar `start-access-logging`.

AWS CLI

Para habilitar el registro de acceso en un contenedor

En el siguiente `start-access-logging` ejemplo, se habilita el registro de acceso en el contenedor especificado.

```
aws mediastore start-access-logging \  
  --container-name LiveEvents
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Habilitar el registro de acceso para un contenedor](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [StartAccessLogging](#) la Referencia de AWS CLI comandos.

## stop-access-logging

En el siguiente ejemplo de código, se muestra cómo usar `stop-access-logging`.



## AWS CLI

Para deshabilitar el registro de acceso en un contenedor

El siguiente `stop-access-logging` ejemplo deshabilita el registro de acceso en el contenedor especificado.

```
aws mediastore stop-access-logging \  
  --container-name LiveEvents
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Deshabilitar el registro de acceso para un contenedor](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [StopAccessLogging](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

## AWS CLI

Para añadir etiquetas a un contenedor

En el siguiente `tag-resource` ejemplo, se agregan claves y valores de etiquetas al contenedor especificado.

```
aws mediastore tag-resource \  
  --resource arn:aws:mediastore:us-west-2:123456789012:container/ExampleContainer \  
  --tags '[{"Key": "Region", "Value": "West"}, {"Key": "Environment", "Value":  
  "Test"}']
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [TagResource](#) la MediaStore API Referencia AWS elemental.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar etiquetas de un contenedor

En el siguiente `untag-resource` ejemplo, se quita la clave de etiqueta especificada y su valor asociado de un contenedor.

```
aws mediastore untag-resource \  
  --resource arn:aws:mediastore:us-west-2:123456789012:container/ExampleContainer \  
 \  
  --tag-keys Region
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [UntagResource](#) la MediaStore API Referencia AWS elemental. .

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## EMREjemplos de Amazon que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante AmazonEMR.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### add-instance-fleet

En el siguiente ejemplo de código, se muestra cómo usar `add-instance-fleet`.

#### AWS CLI

Para añadir una flota de instancias de tareas a un clúster

En este ejemplo, se agrega una nueva flota de instancias de tareas al clúster especificado.

Comando:

```
aws emr add-instance-fleet --cluster-id 'j-12ABCDEFGHI34JK' --instance-fleet
InstanceFleetType=TASK,TargetSpotCapacity=1,LaunchSpecifications={SpotSpecification={Timeo
```

Salida:

```
{
  "ClusterId": "j-12ABCDEFGHI34JK",
  "InstanceFleetId": "if-23ABCDEFGHI45JJ"
}
```

- Para API obtener más información, consulte [AddInstanceFleet](#) la Referencia de AWS CLI comandos.

### add-steps

En el siguiente ejemplo de código, se muestra cómo usar `add-steps`.

#### AWS CLI

1. Para añadir JAR pasos personalizados a un clúster

Comando:

```
aws emr add-steps --cluster-id j-XXXXXXX --steps
Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s3://mybucket/
mytest.jar,Args=arg1,arg2,arg3
Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s3://mybucket/
mytest.jar,MainClass=mymainclass,Args=arg1,arg2,arg3
```

**Parámetros necesarios:**

```
Jar
```

**Parámetros opcionales:**

```
Type, Name, ActionOnFailure, Args
```

**Salida:**

```
{
  "StepIds": [
    "s-XXXXXXXX",
    "s-YYYYYYYY"
  ]
}
```

**2. Para añadir pasos de streaming a un clúster****Comando:**

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps Type=STREAMING,Name='Streaming
Program',ActionOnFailure=CONTINUE,Args=[-files,s3://elasticmapreduce/samples/
wordcount/wordSplitter.py,-mapper,wordSplitter.py,-reducer,aggregate,-input,s3://
elasticmapreduce/samples/wordcount/input,-output,s3://mybucket/wordcount/output]
```

**Parámetros necesarios:**

```
Type, Args
```

**Parámetros opcionales:**

```
Name, ActionOnFailure
```

**JSONequivalente (contenido de step.json):**

```
[
  {
    "Name": "JSON Streaming Step",
```

```

    "Args": ["-files", "s3://elasticmapreduce/samples/wordcount/wordSplitter.py", "-
mapper", "wordSplitter.py", "-reducer", "aggregate", "-input", "s3://elasticmapreduce/
samples/wordcount/input", "-output", "s3://mybucket/wordcount/output"],
    "ActionOnFailure": "CONTINUE",
    "Type": "STREAMING"
  }
]

```

NOTE: JSON los argumentos deben incluir opciones y valores como elementos propios de la lista.

Comando (usando step.json):

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps file:///./step.json
```

Salida:

```

{
  "StepIds": [
    "s-XXXXXXXX",
    "s-YYYYYYYY"
  ]
}

```

3. Para añadir un paso de streaming con varios archivos a un clúster (JSONúnicamente)

JSON(multiplefiles.json):

```

[
  {
    "Name": "JSON Streaming Step",
    "Type": "STREAMING",
    "ActionOnFailure": "CONTINUE",
    "Args": [
      "-files",
      "s3://mybucket/mapper.py,s3://mybucket/reducer.py",
      "-mapper",
      "mapper.py",
      "-reducer",
      "reducer.py",
      "-input",
      "s3://mybucket/input",

```

```

        "-output",
        "s3://mybucket/output"]
    }
]

```

Comando:

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps file://./multiplefiles.json
```

Parámetros necesarios:

Type, Args

Parámetros opcionales:

Name, ActionOnFailure

Salida:

```

{
  "StepIds":[
    "s-XXXXXXXX",
  ]
}

```

#### 4. Para añadir escalones de Hive a un clúster

Comando:

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps Type=HIVE,Name='Hive
program',ActionOnFailure=CONTINUE,Args=[-f,s3://mybucket/myhivescript.q,-
d,INPUT=s3://mybucket/myhiveinput,-d,OUTPUT=s3://mybucket/myhiveoutput,arg1,arg2]
Type=HIVE,Name='Hive steps',ActionOnFailure=TERMINATE_CLUSTER,Args=[-
f,s3://elasticmapreduce/samples/hive-ads/libs/model-build.q,-d,INPUT=s3://
elasticmapreduce/samples/hive-ads/tables,-d,OUTPUT=s3://mybucket/hive-ads/
output/2014-04-18/11-07-32,-d,LIBS=s3://elasticmapreduce/samples/hive-ads/libs]
```

Parámetros necesarios:

Type, Args

**Parámetros opcionales:**

Name, ActionOnFailure

**Salida:**

```
{
  "StepIds": [
    "s-XXXXXXXX",
    "s-YYYYYYYY"
  ]
}
```

**5. Para añadir Pig Steps a un clúster****Comando:**

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps Type=PIG,Name='Pig
program',ActionOnFailure=CONTINUE,Args=[-f,s3://mybucket/mypigscript.pig,-
p,INPUT=s3://mybucket/mypiginput,-p,OUTPUT=s3://mybucket/mypigoutput,arg1,arg2]
Type=PIG,Name='Pig program',Args=[-f,s3://elasticmapreduce/samples/pig-apache/do-
reports2.pig,-p,INPUT=s3://elasticmapreduce/samples/pig-apache/input,-p,OUTPUT=s3://
mybucket/pig-apache/output,arg1,arg2]
```

**Parámetros necesarios:**

Type, Args

**Parámetros opcionales:**

Name, ActionOnFailure

**Salida:**

```
{
  "StepIds": [
    "s-XXXXXXXX",
    "s-YYYYYYYY"
  ]
}
```

## 6. Para añadir escalones de Impala a un clúster

Comando:

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps Type=IMPALA,Name='Impala
program',ActionOnFailure=CONTINUE,Args=--impala-script,s3://myimpala/input,--
console-output-path,s3://myimpala/output
```

Parámetros necesarios:

Type, Args

Parámetros opcionales:

Name, ActionOnFailure

Salida:

```
{
  "StepIds": [
    "s-XXXXXXXX",
    "s-YYYYYYYY"
  ]
}
```

- Para API obtener más información, consulte [AddSteps](#) la Referencia de AWS CLI comandos.

## add-tags

En el siguiente ejemplo de código, se muestra cómo usar add-tags.

AWS CLI

### 1. Para añadir etiquetas a un clúster

Comando:

```
aws emr add-tags --resource-id j-xxxxxxx --tags name="John Doe" age=29 sex=male
address="123 East NW Seattle"
```

Salida:



None

## 2. Para enumerar las etiquetas de un clúster

--Comando:

```
aws emr describe-cluster --cluster-id j-XXXXXXYY --query Cluster.Tags
```

Salida:

```
[
  {
    "Value": "male",
    "Key": "sex"
  },
  {
    "Value": "123 East NW Seattle",
    "Key": "address"
  },
  {
    "Value": "John Doe",
    "Key": "name"
  },
  {
    "Value": "29",
    "Key": "age"
  }
]
```

- Para API obtener más información, consulte [AddTags](#) la Referencia de AWS CLI comandos.

## create-cluster-examples

En el siguiente ejemplo de código, se muestra cómo usar `create-cluster-examples`.

### AWS CLI

En la mayoría de los ejemplos siguientes se supone que has especificado tu función de EMR servicio de Amazon y tu perfil de EC2 instancia de Amazon. Si no lo ha hecho, debe especificar cada IAM función obligatoria o utilizar el `--use-default-roles` parámetro al crear el clúster. Para obtener más información sobre la especificación de IAM funciones, consulte [Configurar](#)

[IAM funciones para EMR los permisos de Amazon a los AWS servicios](#) en la Guía EMR de administración de Amazon.

Ejemplo 1: Para crear un clúster

El siguiente `create-cluster` ejemplo crea un EMR clúster simple.

```
aws emr create-cluster \  
  --release-label emr-5.14.0 \  
  --instance-type m4.large \  
  --instance-count 2
```

Este comando no genera ninguna salida.

Ejemplo 2: Para crear un EMR clúster de Amazon con InstanceProfile funciones ServiceRole y funciones predeterminadas

El siguiente `create-cluster` ejemplo crea un EMR clúster de Amazon que usa la `--instance-groups` configuración.

```
aws emr create-cluster \  
  --release-label emr-5.14.0 \  
  --service-role EMR_DefaultRole \  
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \  
  --instance-  
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

Ejemplo 3: Para crear un EMR clúster de Amazon que utilice una flota de instancias

El siguiente `create-cluster` ejemplo crea un EMR clúster de Amazon que usa la `--instance-fleets` configuración y especifica dos tipos de instancias para cada flota y dos EC2 subredes.

```
aws emr create-cluster \  
  --release-label emr-5.14.0 \  
  --service-role EMR_DefaultRole \  
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole,SubnetIds=['subnet-  
ab12345c','subnet-de67890f'] \  
  --instance-fleets  
InstanceFleetType=MASTER,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=m4.la  
InstanceFleetType=CORE,TargetSpotCapacity=11,InstanceTypeConfigs=['{InstanceType=m4.large,B
```

Ejemplo 4: Para crear un clúster con funciones predeterminadas

En el siguiente `create-cluster` ejemplo, se utiliza el `--use-default-roles` parámetro para especificar el rol de servicio y el perfil de instancia predeterminados.

```
aws emr create-cluster \  
  --release-label emr-5.9.0 \  
  --use-default-roles \  
  --instance-  
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.Large InstanceGroupType=CORE  
 \  
  --auto-terminate
```

Ejemplo 5: Para crear un clúster y especificar las aplicaciones que se van a instalar

En el siguiente `create-cluster` ejemplo, se utiliza el `--applications` parámetro para especificar las aplicaciones que EMR instala Amazon. En este ejemplo, se instalan Hadoop, Hive y Pig.

```
aws emr create-cluster \  
  --applications Name=Hadoop Name=Hive Name=Pig \  
  --release-label emr-5.9.0 \  
  --instance-  
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.Large InstanceGroupType=CORE  
 \  
  --auto-terminate
```

Ejemplo 6: Para crear un clúster que incluya Spark

En el siguiente ejemplo, se instala Spark.

```
aws emr create-cluster \  
  --release-label emr-5.9.0 \  
  --applications Name=Spark \  
  --ec2-attributes KeyName=myKey \  
  --instance-  
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.Large InstanceGroupType=CORE  
 \  
  --auto-terminate
```

Ejemplo 7: Para especificar una personalización para AMI utilizarla en las instancias de clúster

El siguiente `create-cluster` ejemplo crea una instancia de clúster basada en Amazon Linux AMI con IDami-a518e6df.

```
aws emr create-cluster \  
  --name "Cluster with My Custom AMI" \  
  --custom-ami-id ami-a518e6df \  
  --ebs-root-volume-size 20 \  
  --release-label emr-5.9.0 \  
  --use-default-roles \  
  --instance-count 2 \  
  --instance-type m4.large
```

Ejemplo 8: Para personalizar las configuraciones de las aplicaciones

En los ejemplos siguientes, se utiliza el `--configurations` parámetro para especificar un archivo de JSON configuración que contiene personalizaciones de aplicaciones para Hadoop. Para obtener más información, consulte [Configuración de aplicaciones](#) en la Amazon EMR Release Guide.

Contenidos de `configurations.json`:

```
[  
  {  
    "Classification": "mapred-site",  
    "Properties": {  
      "mapred.tasktracker.map.tasks.maximum": 2  
    }  
  },  
  {  
    "Classification": "hadoop-env",  
    "Properties": {},  
    "Configurations": [  
      {  
        "Classification": "export",  
        "Properties": {  
          "HADOOP_DATANODE_HEAPSIZE": 2048,  
          "HADOOP_NAMENODE_OPTS": "-XX:GCTimeRatio=19"  
        }  
      }  
    ]  
  }  
]
```

El siguiente ejemplo hace referencia `configurations.json` a un archivo local.

```
aws emr create-cluster \
  --configurations file://configurations.json \
  --release-label emr-5.9.0 \
  --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
\
  --auto-terminate
```

El siguiente ejemplo hace referencia `configurations.json` a un archivo en Amazon S3.

```
aws emr create-cluster \
  --configurations https://s3.amazonaws.com/myBucket/configurations.json \
  --release-label emr-5.9.0 \
  --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
\
  --auto-terminate
```

Ejemplo 9: Para crear un clúster con grupos de instancias principales, principales y de tareas

En el siguiente `create-cluster` ejemplo, `--instance-groups` se especifica el tipo y la cantidad de EC2 instancias que se van a usar para los grupos de instancias principales, principales y de tareas.

```
aws emr create-cluster \
  --release-label emr-5.9.0 \
  --instance-
groups Name=Master,InstanceGroupType=MASTER,InstanceType=m4.large,InstanceCount=1 Name=Core,
```

Ejemplo 10: Para especificar que un clúster debe terminar después de completar todos los pasos

En el siguiente `create-cluster` ejemplo, `--auto-terminate` se especifica que el clúster debe cerrarse automáticamente después de completar todos los pasos.

```
aws emr create-cluster \
  --release-label emr-5.9.0 \
  --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \
  --auto-terminate
```

Ejemplo 11: Para especificar los detalles de la configuración del clúster, como el EC2 key pair de Amazon, la configuración de la red y los grupos de seguridad

En el siguiente `create-cluster` ejemplo, se crea un clúster con el nombre del par de EC2 claves de Amazon `myKey` y el nombre de un perfil de instancia personalizado `myProfile`. Los pares de claves se utilizan para autorizar SSH las conexiones a los nodos del clúster, generalmente al nodo principal. Para obtener más información, consulte [Uso de un par de EC2 claves de Amazon para SSH las credenciales](#) en la Guía EMR de administración de Amazon.

```
aws emr create-cluster \
  --ec2-attributes KeyName=myKey,InstanceProfile=myProfile \
  --release-label emr-5.9.0 \
  --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
\
  --auto-terminate
```

El siguiente ejemplo crea un clúster en una VPC subred de Amazon.

```
aws emr create-cluster \
  --ec2-attributes SubnetId=subnet-xxxxx \
  --release-label emr-5.9.0 \
  --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
\
  --auto-terminate
```

El siguiente ejemplo crea un clúster en la zona de `us-east-1b` disponibilidad.

```
aws emr create-cluster \
  --ec2-attributes AvailabilityZone=us-east-1b \
  --release-label emr-5.9.0 \
  --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

En el siguiente ejemplo, se crea un clúster y se especifican únicamente los grupos de seguridad EMR gestionados por Amazon.

```
aws emr create-cluster \
  --release-label emr-5.9.0 \
  --service-role myServiceRole \
```

```

--ec2-attributes InstanceProfile=myRole,EmrManagedMasterSecurityGroup=sg-
master1,EmrManagedSlaveSecurityGroup=sg-slave1 \
--instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE

```

En el siguiente ejemplo, se crea un clúster y se especifican únicamente los grupos de EC2 seguridad de Amazon adicionales.

```

aws emr create-cluster \
--release-label emr-5.9.0 \
--service-role myServiceRole \
--ec2-attributes InstanceProfile=myRole,AdditionalMasterSecurityGroups=[sg-
addMaster1,sg-addMaster2,sg-addMaster3,sg-
addMaster4],AdditionalSlaveSecurityGroups=[sg-addSlave1,sg-addSlave2,sg-
addSlave3,sg-addSlave4] \
--instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE

```

En el siguiente ejemplo, se crea un clúster y se especifican los grupos de seguridad EMR gestionados, así como los grupos de seguridad adicionales.

```

aws emr create-cluster \
--release-label emr-5.9.0 \
--service-role myServiceRole \
--ec2-attributes InstanceProfile=myRole,EmrManagedMasterSecurityGroup=sg-
master1,EmrManagedSlaveSecurityGroup=sg-slave1,AdditionalMasterSecurityGroups=[sg-
addMaster1,sg-addMaster2,sg-addMaster3,sg-
addMaster4],AdditionalSlaveSecurityGroups=[sg-addSlave1,sg-addSlave2,sg-
addSlave3,sg-addSlave4] \
--instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE

```

En el siguiente ejemplo, se crea un clúster en una subred VPC privada y se utiliza un grupo de EC2 seguridad de Amazon específico para habilitar el acceso a los EMR servicios de Amazon, que es necesario para los clústeres de las subredes privadas.

```

aws emr create-cluster \
--release-label emr-5.9.0 \
--service-role myServiceRole \
--ec2-attributes InstanceProfile=myRole,ServiceAccessSecurityGroup=sg-service-
access,EmrManagedMasterSecurityGroup=sg-master,EmrManagedSlaveSecurityGroup=sg-slave
\

```

```
--instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

En el siguiente ejemplo, se especifican los parámetros de configuración del grupo de seguridad mediante un nombre de JSON archivo `ec2_attributes.json` que se almacena localmente. NOTE: JSON los argumentos deben incluir opciones y valores como elementos propios de la lista.

```
aws emr create-cluster \
  --release-label emr-5.9.0 \
  --service-role myServiceRole \
  --ec2-attributes file://ec2_attributes.json \
  --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

Contenidos de `ec2_attributes.json`:

```
[
  {
    "SubnetId": "subnet-xxxxx",
    "KeyName": "myKey",
    "InstanceProfile": "myRole",
    "EmrManagedMasterSecurityGroup": "sg-master1",
    "EmrManagedSlaveSecurityGroup": "sg-slave1",
    "ServiceAccessSecurityGroup": "sg-service-access",
    "AdditionalMasterSecurityGroups": ["sg-addMaster1", "sg-addMaster2", "sg-
addMaster3", "sg-addMaster4"],
    "AdditionalSlaveSecurityGroups": ["sg-addSlave1", "sg-addSlave2", "sg-
addSlave3", "sg-addSlave4"]
  }
]
```

Ejemplo 12: Para habilitar la depuración y especificar un registro URI

El siguiente `create-cluster` ejemplo usa el `--enable-debugging` parámetro, que le permite ver los archivos de registro más fácilmente con la herramienta de depuración de la EMR consola de Amazon. El `--log-uri` parámetro es obligatorio con `--enable-debugging`.

```
aws emr create-cluster \
  --enable-debugging \
  --log-uri s3://myBucket/myLog \
```



```

--release-label emr-5.9.0 \
--instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
\
--auto-terminate

```

Ejemplo 13: Para añadir etiquetas al crear un clúster

Las etiquetas son pares clave-valor que ayudan a identificar y gestionar los clústeres. En el siguiente `create-cluster` ejemplo, se utiliza el `--tags` parámetro para crear tres etiquetas para un clúster: una con el nombre `name` y el valor de la clave `Shirley Rodriguez`, una segunda con el nombre de la clave `age` y el valor `29`, y una tercera etiqueta con el nombre de la clave `department` y el valor `Analytics`

```

aws emr create-cluster \
--tags name="Shirley Rodriguez" age=29 department="Analytics" \
--release-label emr-5.32.0 \
--instance-type m5.xlarge \
--instance-count 3 \
--use-default-roles

```

En el siguiente ejemplo, se enumeran las etiquetas aplicadas a un clúster.

```

aws emr describe-cluster \
--cluster-id j-XXXXXXXXY \
--query Cluster.Tags

```

Ejemplo 14: Para usar una configuración de seguridad que habilite el cifrado y otras funciones de seguridad

En el siguiente `create-cluster` ejemplo, se utiliza el `--security-configuration` parámetro para especificar una configuración de seguridad para un EMR clúster. Puedes usar configuraciones de seguridad con Amazon EMR versión 4.8.0 o posterior.

```

aws emr create-cluster \
--instance-type m4.large \
--release-label emr-5.9.0 \
--security-configuration mySecurityConfiguration

```

Ejemplo 15: Para crear un clúster con volúmenes de EBS almacenamiento adicionales configurados para los grupos de instancias

Al especificar EBS volúmenes adicionales, se requieren los siguientes argumentos: `VolumeType`, `SizeInGB` si `EbsBlockDeviceConfigs` se especifica.

En el siguiente `create-cluster` ejemplo, se crea un clúster con varios EBS volúmenes adjuntos a EC2 las instancias del grupo de instancias principal.

```
aws emr create-cluster \
  --release-label emr-5.9.0 \
  --use-default-roles \
  --instance-
groups InstanceGroupType=MASTER, InstanceCount=1, InstanceType=d2.xlarge
  'InstanceGroupType=CORE, InstanceCount=2, InstanceType=d2.xlarge, EbsConfiguration={EbsOptimiz
{VolumeSpecification={VolumeType=io1, SizeInGB=100, Iops=100}, VolumesPerInstance=4}}]'
\
  --auto-terminate
```

En el siguiente ejemplo, se crea un clúster con varios EBS volúmenes adjuntos a EC2 las instancias del grupo de instancias maestro.

```
aws emr create-cluster \
  --release-label emr-5.9.0 \
  --use-default-roles \
  --instance-groups 'InstanceGroupType=MASTER, InstanceCount=1,
InstanceType=d2.xlarge, EbsConfiguration={EbsOptimized=true,
EbsBlockDeviceConfigs=[{VolumeSpecification={VolumeType=io1, SizeInGB=100,
Iops=100}},
{VolumeSpecification={VolumeType=standard, SizeInGB=50}, VolumesPerInstance=3}}]' InstanceGroup
\
  --auto-terminate
```

Ejemplo 16: Para crear un clúster con una política de escalado automático

Puede adjuntar políticas de escalado automático a los grupos de instancias principales y de tareas mediante Amazon 4.0 y EMR versiones posteriores. La política de escalado automático añade y elimina EC2 instancias de forma dinámica en respuesta a una CloudWatch métrica de Amazon. Para obtener más información, consulte [Uso del escalado automático en Amazon EMR](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-automatic-scaling.html) < <https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-automatic-scaling.html> > en la Guía de administración de Amazon. EMR

Al adjuntar una política de escalado automático, también debes especificar el rol por defecto que utilizarás el escalado automático. `--auto-scaling-role` `EMR_AutoScaling_DefaultRole`

En el siguiente `create-cluster` ejemplo, se especifica la política de escalado automático del grupo de CORE instancias mediante el `AutoScalingPolicy` argumento con una JSON estructura integrada, que especifica la configuración de la política de escalado. Los grupos de instancias con una JSON estructura integrada deben tener toda la colección de argumentos entre comillas simples. El uso de comillas simples es opcional para los grupos de instancias sin una JSON estructura incrustada.

```
aws emr create-cluster
  --release-label emr-5.9.0 \
  --use-default-roles --auto-scaling-role EMR_AutoScaling_DefaultRole \
  --instance-
groups InstanceGroupType=MASTER,InstanceType=d2.xlarge,InstanceCount=1
  'InstanceGroupType=CORE,InstanceType=d2.xlarge,InstanceCount=2,AutoScalingPolicy={Constrain
```

En el siguiente ejemplo, se usa un JSON archivo `instancegroupconfig.json`, para especificar la configuración de todos los grupos de instancias de un clúster. El JSON archivo especifica la configuración de la política de escalado automático para el grupo de instancias principal.

```
aws emr create-cluster \
  --release-label emr-5.9.0 \
  --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
  --instance-groups file://myfolder/instancegroupconfig.json \
  --auto-scaling-role EMR_AutoScaling_DefaultRole
```

Contenidos de `instancegroupconfig.json`:

```
[
  {
    "InstanceCount": 1,
    "Name": "MyMasterIG",
    "InstanceGroupType": "MASTER",
    "InstanceType": "m4.large"
  },
  {
    "InstanceCount": 2,
    "Name": "MyCoreIG",
    "InstanceGroupType": "CORE",
    "InstanceType": "m4.large",
    "AutoScalingPolicy": {
```

```

    "Constraints": {
      "MinCapacity": 2,
      "MaxCapacity": 10
    },
    "Rules": [
      {
        "Name": "Default-scale-out",
        "Description": "Replicates the default scale-out rule in the
console for YARN memory.",
        "Action": {
          "SimpleScalingPolicyConfiguration": {
            "AdjustmentType": "CHANGE_IN_CAPACITY",
            "ScalingAdjustment": 1,
            "CoolDown": 300
          }
        },
        "Trigger": {
          "CloudWatchAlarmDefinition": {
            "ComparisonOperator": "LESS_THAN",
            "EvaluationPeriods": 1,
            "MetricName": "YARNMemoryAvailablePercentage",
            "Namespace": "AWS/ElasticMapReduce",
            "Period": 300,
            "Threshold": 15,
            "Statistic": "AVERAGE",
            "Unit": "PERCENT",
            "Dimensions": [
              {
                "Key": "JobFlowId",
                "Value": "${emr.clusterId}"
              }
            ]
          }
        }
      }
    ]
  }
}
]

```

### Ejemplo 17: Agrega JAR pasos personalizados al crear un clúster

En el siguiente `create-cluster` ejemplo, se añaden pasos especificando un JAR archivo almacenado en Amazon S3. Pasos para enviar el trabajo a un clúster. La función principal definida en el JAR archivo se ejecuta una vez que se aprovisionan las EC2 instancias, se ha ejecutado cualquier acción de arranque y se han instalado las aplicaciones. Los pasos se especifican mediante. `Type=CUSTOM_JAR`

JAR Los pasos personalizados requieren el `Jar=` parámetro, que especifica la ruta y el nombre de archivo del JAR. Los parámetros opcionales son `Type` `Name` `ActionOnFailure`, `Args`, y `MainClass`. Si no se especifica la clase principal, el JAR archivo debe especificarlo `MainClass` en su archivo de manifiesto.

```
aws emr create-cluster \
  --steps Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s3://
myBucket/
mytest.jar,Args=arg1,arg2,arg3 Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s3://
myBucket/mytest.jar,MainClass=mymainclass,Args=arg1,arg2,arg3 \
  --release-label emr-5.3.1 \
  --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
\
  --auto-terminate
```

Ejemplo 18: Para añadir pasos de streaming al crear un clúster

En los siguientes `create-cluster` ejemplos, se agrega un paso de transmisión a un clúster que finaliza una vez ejecutados todos los pasos. Los pasos de transmisión requieren parámetros `Type` y `Args` Los parámetros opcionales de los pasos de transmisión son `Name` y `ActionOnFailure`.

El siguiente ejemplo especifica el paso en línea.

```
aws emr create-cluster \
  --steps Type=STREAMING,Name='Streaming Program',ActionOnFailure=CONTINUE,Args=[-
files,s3://elasticmapreduce/samples/wordcount/wordSplitter.py,-
mapper,wordSplitter.py,-reducer,aggregate,-input,s3://elasticmapreduce/samples/
wordcount/input,-output,s3://mybucket/wordcount/output] \
  --release-label emr-5.3.1 \
  --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
\
  --auto-terminate
```

En el ejemplo siguiente se utiliza un archivo de JSON configuración almacenado localmente denominado `multiplefiles.json`. La JSON configuración especifica varios archivos. Para especificar varios archivos en un paso, debe usar un archivo JSON de configuración para especificar el paso. JSON los argumentos deben incluir opciones y valores como elementos propios de la lista.

```
aws emr create-cluster \
  --steps file:///./multiplefiles.json \
  --release-label emr-5.9.0 \
  --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.Large InstanceGroupType=CORE
\
  --auto-terminate
```

Contenidos de `multiplefiles.json`:

```
[
  {
    "Name": "JSON Streaming Step",
    "Args": [
      "-files",
      "s3://elasticmapreduce/samples/wordcount/wordSplitter.py",
      "-mapper",
      "wordSplitter.py",
      "-reducer",
      "aggregate",
      "-input",
      "s3://elasticmapreduce/samples/wordcount/input",
      "-output",
      "s3://mybucket/wordcount/output"
    ],
    "ActionOnFailure": "CONTINUE",
    "Type": "STREAMING"
  }
]
```

Ejemplo 19: Para añadir los pasos de Hive al crear un clúster

En el siguiente ejemplo, se añaden los pasos de Hive al crear un clúster. Los pasos de Hive requieren parámetros `Type` y `Args`. Los parámetros opcionales de los pasos de la colmena son `Name` y `ActionOnFailure`.

```
aws emr create-cluster \
  --steps Type=HIVE,Name='Hive
  program',ActionOnFailure=CONTINUE,ActionOnFailure=TERMINATE_CLUSTER,Args=[-
  f,s3://elasticmapreduce/samples/hive-ads/libs/model-build.q,-d,INPUT=s3://
  elasticmapreduce/samples/hive-ads/tables,-d,OUTPUT=s3://mybucket/hive-ads/
  output/2014-04-18/11-07-32,-d,LIBS=s3://elasticmapreduce/samples/hive-ads/libs] \
  --applications Name=Hive \
  --release-label emr-5.3.1 \
  --instance-
  groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

Ejemplo 20: Para añadir pasos de Pig al crear un clúster

El siguiente ejemplo agrega los pasos de Pig al crear un clúster. Los parámetros necesarios de Pig Steps son Type yArgs. Los parámetros opcionales de Pig Steps son Name yActionOnFailure.

```
aws emr create-cluster \
  --steps Type=PIG,Name='Pig program',ActionOnFailure=CONTINUE,Args=[-f,s3://
  elasticmapreduce/samples/pig-apache/do-reports2.pig,-p,INPUT=s3://elasticmapreduce/
  samples/pig-apache/input,-p,OUTPUT=s3://mybucket/pig-apache/output] \
  --applications Name=Pig \
  --release-label emr-5.3.1 \
  --instance-
  groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

Ejemplo 21: Para añadir acciones de bootstrap

El siguiente create-cluster ejemplo ejecuta dos acciones de arranque definidas como scripts que se almacenan en Amazon S3.

```
aws emr create-cluster \
  --bootstrap-actions Path=s3://mybucket/
  myscript1,Name=BootstrapAction1,Args=[arg1,arg2] Path=s3://mybucket/
  myscript2,Name=BootstrapAction2,Args=[arg1,arg2] \
  --release-label emr-5.3.1 \
  --instance-
  groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
  \
  --auto-terminate
```

Ejemplo 22: Para permitir EMRFS una visualización coherente y personalizar la configuración `RetryCount` y `RetryPeriod`

El siguiente `create-cluster` ejemplo especifica el número de reintentos y el período de reintentos para una visualización EMRFS coherente. El argumento `Consistent=true` es obligatorio.

```
aws emr create-cluster \  
  --instance-type m4.large \  
  --release-label emr-5.9.0 \  
  --emrfs Consistent=true,RetryCount=6,RetryPeriod=30
```

En el ejemplo siguiente se especifica la misma EMRFS configuración que en el ejemplo anterior, mediante un archivo de JSON configuración almacenado localmente denominado `emrfsconfig.json`

```
aws emr create-cluster \  
  --instance-type m4.large \  
  --release-label emr-5.9.0 \  
  --emrfs file://emrfsconfig.json
```

Contenidos de `emrfsconfig.json`:

```
{  
  "Consistent": true,  
  "RetryCount": 6,  
  "RetryPeriod": 30  
}
```

Ejemplo 23: Para crear un clúster con Kerberos configurado

En los siguientes `create-cluster` ejemplos se crea un clúster mediante una configuración de seguridad con Kerberos activado y se establecen los parámetros de Kerberos para el clúster que se utiliza. `--kerberos-attributes`

El siguiente comando especifica los atributos de Kerberos para el clúster en línea.

```
aws emr create-cluster \  
  --instance-type m3.xlarge \  
  --release-label emr-5.10.0 \  
  --kerberos-attributes ...
```



```

--service-role EMR_DefaultRole \
--ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
--security-configuration mySecurityConfiguration \
--kerberos-
attributes Realm=EC2.INTERNAL,KdcAdminPassword=123,CrossRealmTrustPrincipalPassword=123

```

El siguiente comando especifica los mismos atributos, pero hace referencia a un JSON archivo almacenado localmente denominado. `kerberos_attributes.json` En este ejemplo, el archivo se guarda en el mismo directorio en el que se ejecuta el comando. También puede hacer referencia a un archivo de configuración guardado en Amazon S3.

```

aws emr create-cluster \
--instance-type m3.xlarge \
--release-label emr-5.10.0 \
--service-role EMR_DefaultRole \
--ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
--security-configuration mySecurityConfiguration \
--kerberos-attributes file://kerberos_attributes.json

```

Contenidos de `kerberos_attributes.json`:

```

{
  "Realm": "EC2.INTERNAL",
  "KdcAdminPassword": "123",
  "CrossRealmTrustPrincipalPassword": "123",
}

```

El siguiente `create-cluster` ejemplo crea un EMR clúster de Amazon que utiliza la `--instance-groups` configuración y tiene una política de escalado gestionada.

```

aws emr create-cluster \
--release-label emr-5.30.0 \
--service-role EMR_DefaultRole \
--ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
--instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
--managed-scaling-policy
ComputeLimits='{MinimumCapacityUnits=2,MaximumCapacityUnits=4,UnitType=Instances}'

```

El siguiente `create-cluster` ejemplo crea un EMR clúster de Amazon que usa el `<<-- log-encryption-kms-key-id` para definir el ID de KMS clave utilizado para el cifrado de registros.

```
aws emr create-cluster \
  --release-label emr-5.30.0 \
  --log-uri s3://myBucket/myLog \
  --log-encryption-kms-key-id arn:aws:kms:us-east-1:110302272565:key/dd559181-283e-45d7-99d1-66da348c4d33 \
  --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.Large InstanceGroupType=CORE,
```

El siguiente `create-cluster` ejemplo crea un EMR clúster de Amazon que usa la configuración «`--placement-group-configs`» para colocar los nodos maestros en un clúster de alta disponibilidad (HA) dentro de un grupo de EC2 ubicación mediante una estrategia de SPREAD ubicación.

```
aws emr create-cluster \
  --release-label emr-5.30.0 \
  --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
  --instance-
groups InstanceGroupType=MASTER,InstanceCount=3,InstanceType=m4.LargeInstanceGroupType=CORE,
\
  --placement-group-configs InstanceRole=MASTER
```

En el siguiente `create-cluster` ejemplo, se crea un EMR clúster de Amazon que utiliza la configuración `auto-termination-policy` «`--`» para establecer un umbral de terminación automática de inactividad para el clúster.

```
aws emr create-cluster \
  --release-label emr-5.34.0 \
  --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
  --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.Large InstanceGroupType=CORE,
\
  --auto-termination-policy IdleTimeout=100
```

El siguiente `create-cluster` ejemplo crea un EMR clúster de Amazon que usa «`--os-release-label`» para definir una versión de Amazon Linux para el lanzamiento del clúster.

```
aws emr create-cluster \
  --release-label emr-6.6.0 \
  --os-release-label 2.0.20220406.1 \
  --service-role EMR_DefaultRole \
```

```
--ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
--instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.Large InstanceGroupType=CORE
```

Ejemplo 24: Para especificar los atributos de un volumen EBS raíz: tamaño, IOPS y rendimiento para las instancias de clúster creadas con las EMR versiones 6.15.0 y posteriores

El siguiente `create-cluster` ejemplo crea un EMR clúster de Amazon que usa atributos de volumen raíz para configurar las especificaciones de los volúmenes raíz de las EC2 instancias.

```
aws emr create-cluster \
  --name "Cluster with My Custom AMI" \
  --custom-ami-id ami-a518e6df \
  --ebs-root-volume-size 20 \
  --ebs-root-volume-iops 3000 \
  --ebs-root-volume-throughput 125 \
  --release-label emr-6.15.0 \
  --use-default-roles \
  --instance-count 2 \
  --instance-type m4.Large
```

- Para API obtener más información, consulte [CreateClusterExamples](#) la Referencia de AWS CLI comandos.

## create-default-roles

En el siguiente ejemplo de código, se muestra cómo usar `create-default-roles`.

### AWS CLI

1. Para crear el IAM rol predeterminado para EC2

Comando:

```
aws emr create-default-roles
```

Salida:

```
If the role already exists then the command returns nothing.
```

```
If the role does not exist then the output will be:
```

```
[
  {
    "RolePolicy": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Action": [
            "cloudwatch:*",
            "dynamodb:*",
            "ec2:Describe*",
            "elasticmapreduce:Describe*",
            "elasticmapreduce:ListBootstrapActions",
            "elasticmapreduce:ListClusters",
            "elasticmapreduce:ListInstanceGroups",
            "elasticmapreduce:ListInstances",
            "elasticmapreduce:ListSteps",
            "kinesis:CreateStream",
            "kinesis>DeleteStream",
            "kinesis:DescribeStream",
            "kinesis:GetRecords",
            "kinesis:GetShardIterator",
            "kinesis:MergeShards",
            "kinesis:PutRecord",
            "kinesis:SplitShard",
            "rds:Describe*",
            "s3:*",
            "sdb:*",
            "sns:*",
            "sqs:*"
          ],
          "Resource": "*",
          "Effect": "Allow"
        }
      ]
    },
    "Role": {
      "AssumeRolePolicyDocument": {
        "Version": "2008-10-17",
        "Statement": [
          {
            "Action": "sts:AssumeRole",
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
```

```
        "Service": "ec2.amazonaws.com"
      }
    ]
  },
  "RoleId": "AR0AIQ5SIUGL5KMYBJX6",
  "CreateDate": "2015-06-09T17:09:04.602Z",
  "RoleName": "EMR_EC2_DefaultRole",
  "Path": "/",
  "Arn": "arn:aws:iam::176430881729:role/EMR_EC2_DefaultRole"
}
},
{
  "RolePolicy": {
    "Version": "2012-10-17",
    "Statement": [
      {
        "Action": [
          "ec2:AuthorizeSecurityGroupIngress",
          "ec2:CancelSpotInstanceRequests",
          "ec2:CreateSecurityGroup",
          "ec2:CreateTags",
          "ec2>DeleteTags",
          "ec2:DescribeAvailabilityZones",
          "ec2:DescribeAccountAttributes",
          "ec2:DescribeInstances",
          "ec2:DescribeInstanceStatus",
          "ec2:DescribeKeyPairs",
          "ec2:DescribePrefixLists",
          "ec2:DescribeRouteTables",
          "ec2:DescribeSecurityGroups",
          "ec2:DescribeSpotInstanceRequests",
          "ec2:DescribeSpotPriceHistory",
          "ec2:DescribeSubnets",
          "ec2:DescribeVpcAttribute",
          "ec2:DescribeVpcEndpoints",
          "ec2:DescribeVpcEndpointServices",
          "ec2:DescribeVpcs",
          "ec2:ModifyImageAttribute",
          "ec2:ModifyInstanceAttribute",
          "ec2:RequestSpotInstances",
          "ec2:RunInstances",
          "ec2:TerminateInstances",
          "iam:GetRole",
```

```

        "iam:GetRolePolicy",
        "iam:ListInstanceProfiles",
        "iam:ListRolePolicies",
        "iam:PassRole",
        "s3:CreateBucket",
        "s3:Get*",
        "s3:List*",
        "sdb:BatchPutAttributes",
        "sdb:Select",
        "sqs:CreateQueue",
        "sqs>Delete*",
        "sqs:GetQueue*",
        "sqs:ReceiveMessage"
    ],
    "Resource": "*",
    "Effect": "Allow"
}
]
},
"Role": {
    "AssumeRolePolicyDocument": {
        "Version": "2008-10-17",
        "Statement": [
            {
                "Action": "sts:AssumeRole",
                "Sid": "",
                "Effect": "Allow",
                "Principal": {
                    "Service": "elasticmapreduce.amazonaws.com"
                }
            }
        ]
    },
    "RoleId": "AROAI3SRVPPVSRDLARBPY",
    "CreateDate": "2015-06-09T17:09:10.401Z",
    "RoleName": "EMR_DefaultRole",
    "Path": "/",
    "Arn": "arn:aws:iam::176430881729:role/EMR_DefaultRole"
}
]

```

- Para API obtener más información, consulte [CreateDefaultRoles](#) la Referencia de AWS CLI comandos.

## create-security-configuration

En el siguiente ejemplo de código, se muestra cómo usar `create-security-configuration`.

### AWS CLI

1. Para crear una configuración de seguridad con el cifrado en tránsito habilitado PEM para el proveedor de certificados y el cifrado en reposo habilitado con SSE -S3 para el cifrado S3 y AWS-KMS para el proveedor de claves de disco local

Comando:

```
aws emr create-security-configuration --name MySecurityConfig --security-configuration '{
  "EncryptionConfiguration": {
    "EnableInTransitEncryption" : true,
    "EnableAtRestEncryption" : true,
    "InTransitEncryptionConfiguration" : {
      "TLSCertificateConfiguration" : {
        "CertificateProviderType" : "PEM",
        "S3Object" : "s3://mycertstore/artifacts/MyCerts.zip"
      }
    },
    "AtRestEncryptionConfiguration" : {
      "S3EncryptionConfiguration" : {
        "EncryptionMode" : "SSE-S3"
      },
      "LocalDiskEncryptionConfiguration" : {
        "EncryptionKeyProviderType" : "AwsKms",
        "AwsKmsKey" : "arn:aws:kms:us-east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
      }
    }
  }
}'
```

Salida:

```
{
  "CreationDateTime": 1474070889.129,
  "Name": "MySecurityConfig"
}
```

JSONequivalente (contenido de security\_configuration.json):

```
{
  "EncryptionConfiguration": {
    "EnableInTransitEncryption": true,
    "EnableAtRestEncryption": true,
    "InTransitEncryptionConfiguration": {
      "TLSCertificateConfiguration": {
        "CertificateProviderType": "PEM",
        "S3Object": "s3://mycertstore/artifacts/MyCerts.zip"
      }
    },
    "AtRestEncryptionConfiguration": {
      "S3EncryptionConfiguration": {
        "EncryptionMode": "SSE-S3"
      },
      "LocalDiskEncryptionConfiguration": {
        "EncryptionKeyProviderType": "AwsKms",
        "AwsKmsKey": "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
      }
    }
  }
}
```

Comando (usando security\_configuration.json):

```
aws emr create-security-configuration --name "MySecurityConfig" --security-
configuration file://./security_configuration.json
```

Salida:

```
{
  "CreationDateTime": 1474070889.129,
  "Name": "MySecurityConfig"
}
```



## 2. Para crear una configuración de seguridad con Kerberos habilitado mediante la confianza entre dominios y dedicada al clúster KDC

Comando:

```
aws emr create-security-configuration --name MySecurityConfig --security-configuration '{
  "AuthenticationConfiguration": {
    "KerberosConfiguration": {
      "Provider": "ClusterDedicatedKdc",
      "ClusterDedicatedKdcConfiguration": {
        "TicketLifetimeInHours": 24,
        "CrossRealmTrustConfiguration": {
          "Realm": "AD.DOMAIN.COM",
          "Domain": "ad.domain.com",
          "AdminServer": "ad.domain.com",
          "KdcServer": "ad.domain.com"
        }
      }
    }
  }
}'
```

Salida:

```
{
  "CreationDateTime": 1490225558.982,
  "Name": "MySecurityConfig"
}
```

JSONequivalente (contenido de security\_configuration.json):

```
{
  "AuthenticationConfiguration": {
    "KerberosConfiguration": {
      "Provider": "ClusterDedicatedKdc",
      "ClusterDedicatedKdcConfiguration": {
        "TicketLifetimeInHours": 24,
        "CrossRealmTrustConfiguration": {
          "Realm": "AD.DOMAIN.COM",
          "Domain": "ad.domain.com",
          "AdminServer": "ad.domain.com",
```

```
    "KdcServer": "ad.domain.com"
  }
}
}
```

Comando (usando security\_configuration.json):

```
aws emr create-security-configuration --name "MySecurityConfig" --security-configuration file://./security_configuration.json
```

Salida:

```
{
  "CreationDateTime": 1490225558.982,
  "Name": "MySecurityConfig"
}
```

- Para obtener API más información, consulte la Referencia de comandos.  
[CreateSecurityConfiguration](#) AWS CLI

## delete-security-configuration

En el siguiente ejemplo de código, se muestra cómo usar delete-security-configuration.

AWS CLI

Para eliminar una configuración de seguridad en la región actual

Comando:

```
aws emr delete-security-configuration --name MySecurityConfig
```

Salida:

```
None
```

- Para API obtener más información, consulte [DeleteSecurityConfiguration](#) la Referencia de AWS CLI comandos.

## describe-cluster

En el siguiente ejemplo de código, se muestra cómo usar `describe-cluster`.

### AWS CLI

Comando:

```
aws emr describe-cluster --cluster-id j-XXXXXXXX
```

Salida:

For release-label based uniform instance groups cluster:

```
{
  "Cluster": {
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1436475075.199,
        "CreationDateTime": 1436474656.563,
      },
      "State": "WAITING",
      "StateChangeReason": {
        "Message": "Waiting for steps to run"
      }
    },
    "Ec2InstanceAttributes": {
      "ServiceAccessSecurityGroup": "sg-xxxxxxx",
      "EmrManagedMasterSecurityGroup": "sg-xxxxxxx",
      "IamInstanceProfile": "EMR_EC2_DefaultRole",
      "Ec2KeyName": "myKey",
      "Ec2AvailabilityZone": "us-east-1c",
      "EmrManagedSlaveSecurityGroup": "sg-yyyyyyyyyy"
    },
    "Name": "My Cluster",
    "ServiceRole": "EMR_DefaultRole",
    "Tags": [],
    "TerminationProtected": true,
    "UnhealthyNodeReplacement": true,
    "ReleaseLabel": "emr-4.0.0",
    "NormalizedInstanceHours": 96,
    "InstanceGroups": [
      {
```

```
    "RequestedInstanceCount": 2,
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1436475074.245,
        "CreationDateTime": 1436474656.564,
        "EndDateTime": 1436638158.387
      },
      "State": "RUNNING",
      "StateChangeReason": {
        "Message": "",
      }
    },
    "Name": "CORE",
    "InstanceGroupType": "CORE",
    "Id": "ig-YYYYYYYY",
    "Configurations": [],
    "InstanceType": "m3.large",
    "Market": "ON_DEMAND",
    "RunningInstanceCount": 2
  },
  {
    "RequestedInstanceCount": 1,
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1436475074.245,
        "CreationDateTime": 1436474656.564,
        "EndDateTime": 1436638158.387
      },
      "State": "RUNNING",
      "StateChangeReason": {
        "Message": "",
      }
    },
    "Name": "MASTER",
    "InstanceGroupType": "MASTER",
    "Id": "ig-XXXXXXXXXX",
    "Configurations": [],
    "InstanceType": "m3.large",
    "Market": "ON_DEMAND",
    "RunningInstanceCount": 1
  }
],
"Applications": [
  {
```

```

        "Name": "Hadoop"
      }
    ],
    "VisibleToAllUsers": true,
    "BootstrapActions": [],
    "MasterPublicDnsName": "ec2-54-147-144-78.compute-1.amazonaws.com",
    "AutoTerminate": false,
    "Id": "j-XXXXXXXX",
    "Configurations": [
      {
        "Properties": {
          "fs.s3.consistent.retryPeriodSeconds": "20",
          "fs.s3.enableServerSideEncryption": "true",
          "fs.s3.consistent": "false",
          "fs.s3.consistent.retryCount": "2"
        },
        "Classification": "emrfs-site"
      }
    ]
  }
}

```

For release-label based instance fleet cluster:

```

{
  "Cluster": {
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1487897289.705,
        "CreationDateTime": 1487896933.942
      },
      "State": "WAITING",
      "StateChangeReason": {
        "Message": "Waiting for steps to run"
      }
    },
    "Ec2InstanceAttributes": {
      "EmrManagedMasterSecurityGroup": "sg-xxxxx",
      "RequestedEc2AvailabilityZones": [],
      "RequestedEc2SubnetIds": [],
      "IamInstanceProfile": "EMR_EC2_DefaultRole",
      "Ec2AvailabilityZone": "us-east-1a",
      "EmrManagedSlaveSecurityGroup": "sg-xxxxx"
    },
  },
}

```

```
"Name": "My Cluster",
"ServiceRole": "EMR_DefaultRole",
"Tags": [],
"TerminationProtected": false,
"UnhealthyNodeReplacement": false,
"ReleaseLabel": "emr-5.2.0",
"NormalizedInstanceHours": 472,
"InstanceCollectionType": "INSTANCE_FLEET",
"InstanceFleets": [
  {
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1487897212.74,
        "CreationDateTime": 1487896933.948
      },
      "State": "RUNNING",
      "StateChangeReason": {
        "Message": ""
      }
    },
    "ProvisionedSpotCapacity": 1,
    "Name": "MASTER",
    "InstanceFleetType": "MASTER",
    "LaunchSpecifications": {
      "SpotSpecification": {
        "TimeoutDurationMinutes": 60,
        "TimeoutAction": "TERMINATE_CLUSTER"
      }
    },
    "TargetSpotCapacity": 1,
    "ProvisionedOnDemandCapacity": 0,
    "InstanceTypeSpecifications": [
      {
        "BidPrice": "0.5",
        "InstanceType": "m3.xlarge",
        "WeightedCapacity": 1
      }
    ],
    "Id": "if-xxxxxxx",
    "TargetOnDemandCapacity": 0
  }
],
"Applications": [
  {
```

```

        "Version": "2.7.3",
        "Name": "Hadoop"
    }
],
"ScaleDownBehavior": "TERMINATE_AT_INSTANCE_HOUR",
"VisibleToAllUsers": true,
"BootstrapActions": [],
"MasterPublicDnsName": "ec2-xxx-xx-xxx-xx.compute-1.amazonaws.com",
"AutoTerminate": false,
"Id": "j-xxxxx",
"Configurations": []
}
}

```

For ami based uniform instance group cluster:

```

{
  "Cluster": {
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1399400564.432,
        "CreationDateTime": 1399400268.62
      },
      "State": "WAITING",
      "StateChangeReason": {
        "Message": "Waiting for steps to run"
      }
    },
    "Ec2InstanceAttributes": {
      "IamInstanceProfile": "EMR_EC2_DefaultRole",
      "Ec2AvailabilityZone": "us-east-1c"
    },
    "Name": "My Cluster",
    "Tags": [],
    "TerminationProtected": true,
    "UnhealthyNodeReplacement": true,
    "RunningAmiVersion": "2.5.4",
    "InstanceGroups": [
      {
        "RequestedInstanceCount": 1,
        "Status": {
          "Timeline": {
            "ReadyDateTime": 1399400558.848,
            "CreationDateTime": 1399400268.621
          }
        }
      }
    ]
  }
}

```

```
    },
    "State": "RUNNING",
    "StateChangeReason": {
      "Message": ""
    }
  },
  "Name": "Master instance group",
  "InstanceGroupType": "MASTER",
  "InstanceType": "m1.small",
  "Id": "ig-ABCD",
  "Market": "ON_DEMAND",
  "RunningInstanceCount": 1
},
{
  "RequestedInstanceCount": 2,
  "Status": {
    "Timeline": {
      "ReadyDateTime": 1399400564.439,
      "CreationDateTime": 1399400268.621
    },
    "State": "RUNNING",
    "StateChangeReason": {
      "Message": ""
    }
  },
  "Name": "Core instance group",
  "InstanceGroupType": "CORE",
  "InstanceType": "m1.small",
  "Id": "ig-DEF",
  "Market": "ON_DEMAND",
  "RunningInstanceCount": 2
}
],
"Applications": [
  {
    "Version": "1.0.3",
    "Name": "hadoop"
  }
],
"BootstrapActions": [],
"VisibleToAllUsers": false,
"RequestedAmiVersion": "2.4.2",
"LogUri": "s3://myLogUri/",
"AutoTerminate": false,
```



```

    "Id": "j-XXXXXXXX"
  }
}

```

- Para API obtener más información, consulte [DescribeCluster](#) la Referencia de AWS CLI comandos.

## describe-step

En el siguiente ejemplo de código, se muestra cómo usar `describe-step`.

### AWS CLI

El siguiente comando describe un paso con el ID del paso `s-3LZC0QUT43AM` en un clúster con el ID de clúster `j-3SD91U2E1L2QX`:

```
aws emr describe-step --cluster-id j-3SD91U2E1L2QX --step-id s-3LZC0QUT43AM
```

Salida:

```

{
  "Step": {
    "Status": {
      "Timeline": {
        "EndTime": 1433200470.481,
        "CreationTime": 1433199926.597,
        "StartTime": 1433200404.959
      },
      "State": "COMPLETED",
      "StateChangeReason": {}
    },
    "Config": {
      "Args": [
        "s3://us-west-2.elasticmapreduce/libs/hive/hive-script",
        "--base-path",
        "s3://us-west-2.elasticmapreduce/libs/hive/",
        "--install-hive",
        "--hive-versions",
        "0.13.1"
      ],
      "Jar": "s3://us-west-2.elasticmapreduce/libs/script-runner/script-
runner.jar",

```

```

        "Properties": {}
    },
    "Id": "s-3LZC0QUT43AM",
    "ActionOnFailure": "TERMINATE_CLUSTER",
    "Name": "Setup hive"
}
}

```

- Para API obtener más información, consulte [DescribeStep](#) la Referencia de AWS CLI comandos.

## get

En el siguiente ejemplo de código, se muestra cómo usar `get`.

### AWS CLI

A continuación, se descarga el `hadoop-examples.jar` archivo de la instancia maestra de un clúster con el ID del clúster `j-3SD91U2E1L2QX`:

```
aws emr get --cluster-id j-3SD91U2E1L2QX --key-pair-file ~/.ssh/mykey.pem --src /home/hadoop-examples.jar --dest ~
```

- Para API obtener más información, consulta la referencia de [Get](#) in AWS CLI Command.

## list-clusters

En el siguiente ejemplo de código, se muestra cómo usar `list-clusters`.

### AWS CLI

El siguiente comando muestra todos los EMR clústeres activos de la región actual:

```
aws emr list-clusters --active
```

Salida:

```

{
  "Clusters": [
    {
      "Status": {

```

```

        "Timeline": {
            "ReadyDateTime": 1433200405.353,
            "CreationDateTime": 1433199926.596
        },
        "State": "WAITING",
        "StateChangeReason": {
            "Message": "Waiting after step completed"
        }
    },
    "NormalizedInstanceHours": 6,
    "Id": "j-3SD91U2E1L2QX",
    "Name": "my-cluster"
}
]
}

```

- Para API obtener más información, consulte [ListClusters](#) la Referencia de AWS CLI comandos.

## list-instance-fleets

En el siguiente ejemplo de código, se muestra cómo usar `list-instance-fleets`.

### AWS CLI

Para obtener los detalles de configuración de las flotas de instancias de un clúster

En este ejemplo, se muestran los detalles de las flotas de instancias del clúster especificado.

Comando:

```
list-instance-fleets --cluster-id 'j-12ABCDEFGH134JK'
```

Salida:

```

{
  "InstanceFleets": [
    {
      "Status": {
        "Timeline": {
          "ReadyDateTime": 1488759094.637,
          "CreationDateTime": 1488758719.817
        },
        "State": "RUNNING",

```

```
        "StateChangeReason": {
            "Message": ""
        }
    },
    "ProvisionedSpotCapacity": 6,
    "Name": "CORE",
    "InstanceFleetType": "CORE",
    "LaunchSpecifications": {
        "SpotSpecification": {
            "TimeoutDurationMinutes": 60,
            "TimeoutAction": "TERMINATE_CLUSTER"
        }
    },
    "ProvisionedOnDemandCapacity": 2,
    "InstanceTypeSpecifications": [
        {
            "BidPrice": "0.5",
            "InstanceType": "m3.xlarge",
            "WeightedCapacity": 2
        }
    ],
    "Id": "if-1ABC2DEFGHIJ3"
},
{
    "Status": {
        "Timeline": {
            "ReadyDateTime": 1488759058.598,
            "CreationDateTime": 1488758719.811
        },
        "State": "RUNNING",
        "StateChangeReason": {
            "Message": ""
        }
    },
    "ProvisionedSpotCapacity": 0,
    "Name": "MASTER",
    "InstanceFleetType": "MASTER",
    "ProvisionedOnDemandCapacity": 1,
    "InstanceTypeSpecifications": [
        {
            "BidPriceAsPercentageOfOnDemandPrice": 100.0,
            "InstanceType": "m3.xlarge",
            "WeightedCapacity": 1
        }
    ]
}
```

```

    ],
    "Id": "if-2ABC4DEFGHIJ4"
  }
]
}

```

- Para API obtener más información, consulte [ListInstanceFleets](#) la Referencia de AWS CLI comandos.

## list-instances

En el siguiente ejemplo de código, se muestra cómo usar `list-instances`.

### AWS CLI

El siguiente comando muestra todas las instancias de un clúster con el ID del clúster `j-3C6XNQ39VR9WL`:

```
aws emr list-instances --cluster-id j-3C6XNQ39VR9WL
```

### Salida:

```

For a uniform instance group based cluster
{
  "Instances": [
    {
      "Status": {
        "Timeline": {
          "ReadyDateTime": 1433200400.03,
          "CreationDateTime": 1433199960.152
        },
        "State": "RUNNING",
        "StateChangeReason": {}
      },
      "Ec2InstanceId": "i-f19ecfee",
      "PublicDnsName": "ec2-52-52-41-150.us-west-2.compute.amazonaws.com",
      "PrivateDnsName": "ip-172-21-11-216.us-west-2.compute.internal",
      "PublicIpAddress": "52.52.41.150",
      "Id": "ci-3NNHQUQ2TWB6Y",
      "PrivateIpAddress": "172.21.11.216"
    },
    {

```

```

    "Status": {
      "Timeline": {
        "ReadyDateTime": 1433200400.031,
        "CreationDateTime": 1433199949.102
      },
      "State": "RUNNING",
      "StateChangeReason": {}
    },
    "Ec2InstanceId": "i-1f4ee4c2",
    "PublicDnsName": "ec2-52-63-246-32.us-west-2.compute.amazonaws.com",
    "PrivateDnsName": "ip-172-31-24-130.us-west-2.compute.internal",
    "PublicIpAddress": "52.63.246.32",
    "Id": "ci-GAOCMKNKDCV7",
    "PrivateIpAddress": "172.21.11.215"
  },
  {
    "Status": {
      "Timeline": {
        "ReadyDateTime": 1433200400.031,
        "CreationDateTime": 1433199949.102
      },
      "State": "RUNNING",
      "StateChangeReason": {}
    },
    "Ec2InstanceId": "i-15cfeee3",
    "PublicDnsName": "ec2-52-25-246-63.us-west-2.compute.amazonaws.com",
    "PrivateDnsName": "ip-172-31-24-129.us-west-2.compute.internal",
    "PublicIpAddress": "52.25.246.63",
    "Id": "ci-2W3TDFFB47UAD",
    "PrivateIpAddress": "172.21.11.214"
  }
]
}

```

For a fleet based cluster:

```

{
  "Instances": [
    {
      "Status": {
        "Timeline": {
          "ReadyDateTime": 1487810810.878,
          "CreationDateTime": 1487810588.367,
          "EndDateTime": 1488022990.924
        }
      }
    }
  ]
}

```

```

    },
    "State": "TERMINATED",
    "StateChangeReason": {
        "Message": "Instance was terminated."
    }
},
"Ec2InstanceId": "i-xxxxx",
"InstanceFleetId": "if-xxxxx",
"EbsVolumes": [],
"PublicDnsName": "ec2-xx-xxx-xxx-xxx.compute-1.amazonaws.com",
"InstanceType": "m3.xlarge",
"PrivateDnsName": "ip-xx-xx-xxx-xx.ec2.internal",
"Market": "SPOT",
"PublicIpAddress": "xx.xx.xxx.xxx",
"Id": "ci-xxxxx",
"PrivateIpAddress": "10.47.191.80"
}
]
}

```

- Para API obtener más información, consulta [ListInstances](#) la Referencia de AWS CLI comandos.

## list-security-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-security-configurations`.

### AWS CLI

Para enumerar las configuraciones de seguridad de la región actual

Comando:

```
aws emr list-security-configurations
```

Salida:

```

{
  "SecurityConfigurations": [
    {
      "CreationDateTime": 1473889697.417,
      "Name": "MySecurityConfig-1"
    },
    {

```

```
        "CreationDateTime": 1473889697.417,  
        "Name": "MySecurityConfig-2"  
    }  
]  
}
```

- Para API obtener más información, consulte [ListSecurityConfigurations](#) la Referencia de AWS CLI comandos.

## list-steps

En el siguiente ejemplo de código, se muestra cómo usar `list-steps`.

### AWS CLI

El siguiente comando muestra todos los pasos de un clúster con el ID del clúster `j-3SD91U2E1L2QX`:

```
aws emr list-steps --cluster-id j-3SD91U2E1L2QX
```

- Para API obtener más información, consulte [ListSteps](#) la Referencia de AWS CLI comandos.

## modify-cluster-attributes

En el siguiente ejemplo de código, se muestra cómo usar `modify-cluster-attributes`.

### AWS CLI

El siguiente comando establece la visibilidad de un EMR clúster con el ID `j-301CDNY0J5XM4` para todos los usuarios:

```
aws emr modify-cluster-attributes --cluster-id j-301CDNY0J5XM4 --visible-to-all-users
```

- Para API obtener más información, consulte [ModifyClusterAttributes](#) la Referencia de AWS CLI comandos.

## modify-instance-fleet

En el siguiente ejemplo de código, se muestra cómo usar `modify-instance-fleet`.



## AWS CLI

Para cambiar las capacidades objetivo de una flota de instancias

En este ejemplo, se cambian las capacidades objetivo bajo demanda y puntual a 1 para la flota de instancias especificada.

Comando:

```
aws emr modify-instance-fleet --cluster-id 'j-12ABCDEFGHI34JK' --instance-fleet InstanceFleetId='if-2ABC4DEFGHIJ4',TargetOnDemandCapacity=1,TargetSpotCapacity=1
```

- Para API obtener más información, consulte [ModifyInstanceFleet](#) la Referencia de AWS CLI comandos.

## put

En el siguiente ejemplo de código, se muestra cómo usar put.

## AWS CLI

El siguiente comando carga un archivo con el nombre `healthcheck.sh` de la instancia maestra de un clúster con el ID del clúster: `j-3SD91U2E1L2QX`

```
aws emr put --cluster-id j-3SD91U2E1L2QX --key-pair-file ~/.ssh/mykey.pem --src ~/scripts/healthcheck.sh --dest /home/hadoop/bin/healthcheck.sh
```

- Para API obtener más información, consulta la referencia de [Put](#) in AWS CLI Command.

## remove-tags

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags`.

## AWS CLI

El siguiente comando elimina una etiqueta con la clave `prod` de un clúster con el ID del clúster `j-3SD91U2E1L2QX`:

```
aws emr remove-tags --resource-id j-3SD91U2E1L2QX --tag-keys prod
```

- Para API obtener más información, consulte [RemoveTags](#) la Referencia de AWS CLI comandos.

## schedule-hbase-backup

En el siguiente ejemplo de código, se muestra cómo usar `schedule-hbase-backup`.

### AWS CLI

Nota: Este comando solo se puede utilizar HBase en las AMI versiones 2.x y 3.x

1. Para programar una HBase copia de seguridad completa >>>>>>  
06ab6d6e13564b5733d75abaf3b599f93cf39a23

Comando:

```
aws emr schedule-hbase-backup --cluster-id j-XXXXXXYY --type full --dir
s3://myBucket/backup --interval 10 --unit hours --start-time
2014-04-21T05:26:10Z --consistent
```

Salida:

```
None
```

2. Para programar una copia de HBase seguridad incremental

Comando:

```
aws emr schedule-hbase-backup --cluster-id j-XXXXXXYY --type incremental
--dir s3://myBucket/backup --interval 30 --unit minutes --start-time
2014-04-21T05:26:10Z --consistent
```

Salida:

```
None
```

- Para API obtener más información, consulte [ScheduleHbaseBackup](#) la Referencia de AWS CLI comandos.

## socks

En el siguiente ejemplo de código, se muestra cómo usar socks.

### AWS CLI

El siguiente comando abre una conexión Socks con la instancia maestra de un clúster con el ID del clúster `j-3SD91U2E1L2QX`:

```
aws emr socks --cluster-id j-3SD91U2E1L2QX --key-pair-file ~/.ssh/mykey.pem
```

La opción de archivo de pares de claves toma una ruta local a un archivo de clave privada.

- Para API obtener más información, consulte la referencia de [Socks](#) in AWS CLI Command.

## ssh

En el siguiente ejemplo de código, se muestra cómo usar ssh.

### AWS CLI

El siguiente comando abre una conexión ssh con la instancia maestra de un clúster con el ID del clúster: `j-3SD91U2E1L2QX`

```
aws emr ssh --cluster-id j-3SD91U2E1L2QX --key-pair-file ~/.ssh/mykey.pem
```

La opción de archivo de pares de claves toma una ruta local a un archivo de clave privada.

Salida:

```
ssh -o StrictHostKeyChecking=no -o ServerAliveInterval=10 -i /home/local/user/.ssh/mykey.pem hadoop@ec2-52-52-41-150.us-west-2.compute.amazonaws.com
Warning: Permanently added 'ec2-52-52-41-150.us-west-2.compute.amazonaws.com,52.52.41.150' (ECDSA) to the list of known hosts.
Last login: Mon Jun  1 23:15:38 2015

  _|  _|_  )
  _| (    /  Amazon Linux AMI
  _|\__|__|

https://aws.amazon.com/amazon-linux-ami/2015.03-release-notes/
26 package(s) needed for security, out of 39 available
```

```
Run "sudo yum update" to apply all updates.

-----

Welcome to Amazon Elastic MapReduce running Hadoop and Amazon Linux.

Hadoop is installed in /home/hadoop. Log files are in /mnt/var/log/hadoop. Check
/mnt/var/log/hadoop/steps for diagnosing step failures.

The Hadoop UI can be accessed via the following commands:

ResourceManager    lynx http://ip-172-21-11-216:9026/
NameNode           lynx http://ip-172-21-11-216:9101/

-----

[hadoop@ip-172-31-16-216 ~]$
```

- Para API obtener más información, consulte [Ssh](#) en la Referencia de AWS CLI comandos.

## Amazon EMR sobre EKS ejemplos que utilizan AWS CLI

En los siguientes ejemplos de código, se muestra cómo realizar acciones e implementar situaciones comunes mediante el uso AWS Command Line Interface de Amazon EMR onEKS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde encontrarás instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **update-role-trust-policy**

En el siguiente ejemplo de código, se muestra cómo usar `update-role-trust-policy`.

## AWS CLI

Para actualizar la política de confianza de un IAM rol que se va a utilizar con Amazon EMR el EKS

Este comando de ejemplo actualiza la política de confianza de un rol denominado `example_iam_role` para que pueda usarse con Amazon EMR o con el espacio de nombres `example_namespace` de un EKS clúster llamado `example_cluster`. EKS

Comando:

```
aws emr-containers update-role-trust-policy \  
  --cluster example_cluster \  
  --namespace example_namespace \  
  --role-name example_iam_role
```

Salida:

```
If the trust policy has already been updated, then the output will be:  
Trust policy statement already exists for role example_iam_role. No  
changes were made!
```

```
If the trust policy has not been updated yet, then the output will be:  
Successfully updated trust policy of role example_iam_role.
```

- API Para obtener [UpdateRoleTrustPolicy](#) más información, consulte la Referencia de comandos. AWS CLI

## EventBridge ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with EventBridge.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **delete-rule**

En el siguiente ejemplo de código, se muestra cómo usar `delete-rule`.

#### AWS CLI

Para eliminar una regla de CloudWatch eventos

En este ejemplo, se elimina la regla denominada `EC2InstanceStateChanges`:

```
aws events delete-rule --name "EC2InstanceStateChanges"
```

- Para API obtener más información, consulte [DeleteRule](#) la Referencia de AWS CLI comandos.

### **describe-rule**

En el siguiente ejemplo de código, se muestra cómo usar `describe-rule`.

#### AWS CLI

Para mostrar información sobre una regla de CloudWatch eventos

En este ejemplo se muestra información sobre la regla denominada `DailyLambdaFunction`:

```
aws events describe-rule --name "DailyLambdaFunction"
```

- Para API obtener más información, consulte [DescribeRule](#) la Referencia de AWS CLI comandos.

### **disable-rule**

En el siguiente ejemplo de código, se muestra cómo usar `disable-rule`.

#### AWS CLI

Para deshabilitar una regla de CloudWatch eventos

En este ejemplo, se deshabilita la regla denominada `DailyLambdaFunction`. La regla no se elimina:

```
aws events disable-rule --name "DailyLambdaFunction"
```

- Para API obtener más información, consulte [DisableRule](#) la Referencia de AWS CLI comandos.

## **enable-rule**

En el siguiente ejemplo de código, se muestra cómo usar `enable-rule`.

### AWS CLI

Para habilitar una regla de CloudWatch eventos

Este ejemplo habilita la regla denominada `DailyLambdaFunction`, que estaba deshabilitada anteriormente:

```
aws events enable-rule --name "DailyLambdaFunction"
```

- Para API obtener más información, consulte [EnableRule](#) la Referencia de AWS CLI comandos.

## **list-rule-names-by-target**

En el siguiente ejemplo de código, se muestra cómo usar `list-rule-names-by-target`.

### AWS CLI

Visualización de una lista de todas las reglas que tienen un destino especificado

En este ejemplo se muestran todas las reglas que tienen como destino la función Lambda denominada `MyFunctionName` "»:

```
aws events list-rule-names-by-target --target-arn "arn:aws:lambda:us-east-1:123456789012:function:MyFunctionName"
```

- Para API obtener más información, consulte [ListRuleNamesByTarget](#) la Referencia de AWS CLI comandos.

## list-rules

En el siguiente ejemplo de código, se muestra cómo usar `list-rules`.

### AWS CLI

Para mostrar una lista de todas las reglas de CloudWatch eventos

En este ejemplo se muestran todas las reglas de CloudWatch eventos de la región:

```
aws events list-rules
```

Para mostrar una lista de reglas de CloudWatch eventos que comiencen por una cadena determinada.

En este ejemplo, se muestran todas las reglas de CloudWatch eventos de la región cuyo nombre comience por «Daily»:

```
aws events list-rules --name-prefix "Daily"
```

- Para API obtener más información, consulte [ListRules](#) la Referencia de AWS CLI comandos.

## list-targets-by-rule

En el siguiente ejemplo de código, se muestra cómo usar `list-targets-by-rule`.

### AWS CLI

Para mostrar todos los objetivos de una regla de CloudWatch eventos

En este ejemplo se muestran todos los objetivos de la regla denominada `DailyLambdaFunction`:

```
aws events list-targets-by-rule --rule "DailyLambdaFunction"
```

- Para API obtener más información, consulte [ListTargetsByRule](#) la Referencia de AWS CLI comandos.

## put-events

En el siguiente ejemplo de código, se muestra cómo usar `put-events`.



## AWS CLI

Para enviar un evento personalizado a CloudWatch Events

En este ejemplo, se envía un evento personalizado a CloudWatch Events. El evento está incluido en el archivo `putevents.json`:

```
aws events put-events --entries file://putevents.json
```

Este es el contenido del archivo `putevents.json`:

```
[
  {
    "Source": "com.mycompany.myapp",
    "Detail": "{ \"key1\": \"value1\", \"key2\": \"value2\" }",
    "Resources": [
      "resource1",
      "resource2"
    ],
    "DetailType": "myDetailType"
  },
  {
    "Source": "com.mycompany.myapp",
    "Detail": "{ \"key1\": \"value3\", \"key2\": \"value4\" }",
    "Resources": [
      "resource1",
      "resource2"
    ],
    "DetailType": "myDetailType"
  }
]
```

- Para API obtener más información, consulte [PutEvents](#) la Referencia de AWS CLI comandos.

## **put-rule**

En el siguiente ejemplo de código, se muestra cómo usar `put-rule`.

## AWS CLI

Para crear reglas de CloudWatch eventos

En este ejemplo, se crea una regla que se activa todos los días a las 9:00 a.m. (UTC). Si utiliza `put-targets` para añadir una función de Lambda como destino de esta regla, puede ejecutar la función de Lambda todos los días a la hora especificada:

```
aws events put-rule --name "DailyLambdaFunction" --schedule-expression "cron(0 9 * * ? *)"
```

En este ejemplo, se crea una regla que se activa cuando cualquier EC2 instancia de la región cambia de estado:

```
aws events put-rule --name "EC2InstanceStateChanges" --event-pattern "{\"source\": [\"aws.ec2\"], \"detail-type\": [\"EC2 Instance State-change Notification\"]}" --role-arn "arn:aws:iam::123456789012:role/MyRoleForThisRule"
```

En este ejemplo, se crea una regla que se activa cuando se detiene o finaliza una EC2 instancia de la región:

```
aws events put-rule --name "EC2InstanceStateChangeStopOrTerminate" --event-pattern "{\"source\": [\"aws.ec2\"], \"detail-type\": [\"EC2 Instance State-change Notification\"], \"detail\": {\"state\": [\"stopped\", \"terminated\"]}}" --role-arn "arn:aws:iam::123456789012:role/MyRoleForThisRule"
```

- Para API obtener más información, consulte [PutRule](#) la Referencia de AWS CLI comandos.

## put-targets

En el siguiente ejemplo de código, se muestra cómo usar `put-targets`.

### AWS CLI

Para añadir objetivos a las reglas de CloudWatch eventos

En este ejemplo, se añade una función de Lambda como destino de una regla:

```
aws events put-targets --rule DailyLambdaFunction --targets "Id"="1", "Arn"="arn:aws:lambda:us-east-1:123456789012:function:MyFunctionName"
```

En este ejemplo, se establece una transmisión de Amazon Kinesis como destino, de modo que los eventos detectados por esta regla se retransmitan a la transmisión:

```
aws events put-targets --rule EC2InstanceStateChanges --targets
  "Id"="1", "Arn"="arn:aws:kinesis:us-east-1:123456789012:stream/
  MyStream", "RoleArn"="arn:aws:iam::123456789012:role/MyRoleForThisRule"
```

En este ejemplo, se establecen dos transmisiones de Amazon Kinesis como destinos para una regla:

```
aws events put-targets --rule DailyLambdaFunction --targets
  "Id"="Target1", "Arn"="arn:aws:kinesis:us-east-1:379642911888:stream/
  MyStream1", "RoleArn"="arn:aws:iam::379642911888:role/ MyRoleToAccessLambda"
  "Id"="Target2", "Arn"="arn:aws:kinesis:us-east-1:379642911888:stream/
  MyStream2", "RoleArn"="arn:aws:iam::379642911888:role/MyRoleToAccessLambda"
```

- Para API obtener más información, consulte [PutTargets](#) la Referencia de AWS CLI comandos.

## remove-targets

En el siguiente ejemplo de código, se muestra cómo usar `remove-targets`.

### AWS CLI

Eliminación de un destino de un evento

En este ejemplo, la transmisión de Amazon Kinesis denominada MyStream 1 deja de ser el objetivo de la regla. `DailyLambdaFunction` Cuando `DailyLambdaFunction` se creó, esta transmisión se estableció como un objetivo con un ID de `Target1`:

```
aws events remove-targets --rule "DailyLambdaFunction" --ids "Target1"
```

- Para API obtener más información, consulte [RemoveTargets](#) la Referencia de AWS CLI comandos.

## test-event-pattern

En el siguiente ejemplo de código, se muestra cómo usar `test-event-pattern`.

### AWS CLI

Para comprobar si un patrón de eventos coincide con un evento específico

En este ejemplo se comprueba si el patrón «source:com.mycompany.myapp» coincide con el evento especificado. En este ejemplo, el resultado sería «verdadero»:

```
aws events test-event-pattern --event-pattern "{\"source\": [\"com.mycompany.myapp\n\"]}" --event "{\"id\": \"1\", \"source\": \"com.mycompany.myapp\", \"detail-type\":\n\"myDetailType\", \"account\": \"123456789012\", \"region\": \"us-east-1\", \"time\":\n\"2017-04-11T20:11:04Z\"}"
```

- Para API obtener más información, consulte [TestEventPattern](#) la Referencia de AWS CLI comandos.

## Ejemplos de Firewall Manager que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el AWS Command Line Interface uso de Firewall Manager.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **associate-admin-account**

En el siguiente ejemplo de código, se muestra cómo usar `associate-admin-account`.

### AWS CLI

Para configurar la cuenta de administrador de Firewall Manager

En el siguiente `associate-admin-account` ejemplo, se establece la cuenta de administrador de Firewall Manager.

```
aws fms associate-admin-account \
```

```
--admin-account 123456789012
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configurar la cuenta de administrador de AWS Firewall Manager](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [AssociateAdminAccount](#) la Referencia de AWS CLI comandos.

## **delete-notification-channel**

En el siguiente ejemplo de código, se muestra cómo usar `delete-notification-channel`.

AWS CLI

Para eliminar la información del SNS tema de los registros del Firewall Manager

En el siguiente `delete-notification-channel` ejemplo, se elimina la información del SNS tema.

```
aws fms delete-notification-channel
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de Amazon SNS Notifications y Amazon CloudWatch Alarms](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [DeleteNotificationChannel](#) la Referencia de AWS CLI comandos.

## **delete-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-policy`.

AWS CLI

Para eliminar una política de Firewall Manager

En el siguiente `delete-policy` ejemplo, se elimina la política con el ID especificado, junto con todos sus recursos.

```
aws fms delete-policy \  
  --policy-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --delete-all-policy-resources
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de políticas de AWS Firewall Manager](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [DeletePolicy](#) la Referencia de AWS CLI comandos.

## **disassociate-admin-account**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-admin-account`.

### AWS CLI

Para eliminar la cuenta de administrador de Firewall Manager

En el siguiente `disassociate-admin-account` ejemplo, se elimina la asociación de cuentas de administrador actual de Firewall Manager.

```
aws fms disassociate-admin-account
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configurar la cuenta de administrador de AWS Firewall Manager](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [DisassociateAdminAccount](#) la Referencia de AWS CLI comandos.

## **get-admin-account**

En el siguiente ejemplo de código, se muestra cómo usar `get-admin-account`.

## AWS CLI

Para recuperar la cuenta de administrador de Firewall Manager

En el siguiente `get-admin-account` ejemplo, se recupera la cuenta de administrador.

```
aws fms get-admin-account
```

Salida:

```
{
  "AdminAccount": "123456789012",
  "RoleStatus": "READY"
}
```

Para obtener más información, consulte los [requisitos previos de AWS Firewall Manager](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [GetAdminAccount](#) la Referencia de AWS CLI comandos.

## get-compliance-detail

En el siguiente ejemplo de código, se muestra cómo usar `get-compliance-detail`.

## AWS CLI

Para recuperar la información de cumplimiento de una cuenta

En el siguiente `get-compliance-detail` ejemplo, se recupera la información de cumplimiento de la política y la cuenta de miembro especificadas.

```
aws fms get-compliance-detail \
  --policy-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --member-account 123456789012
```

Salida:

```
{
  "PolicyComplianceDetail": {
    "EvaluationLimitExceeded": false,
```

```
"IssueInfoMap": {},
"MemberAccount": "123456789012",
"PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"PolicyOwner": "123456789012",
"Violators": []
}
```

Para obtener más información, consulte [Cómo comprobar la conformidad de los recursos con una política](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [GetComplianceDetail](#) la Referencia de AWS CLI comandos.

## get-notification-channel

En el siguiente ejemplo de código, se muestra cómo usar `get-notification-channel`.

### AWS CLI

Para recuperar la información del SNS tema de los registros del Firewall Manager

En el siguiente `get-notification-channel` ejemplo, se recupera la información del SNS tema.

```
aws fms get-notification-channel
```

Salida:

```
{
  "SnsTopicArn": "arn:aws:sns:us-west-2:123456789012:us-west-2-fms",
  "SnsRoleName": "arn:aws:iam::123456789012:role/aws-service-role/
fms.amazonaws.com/AWSServiceRoleForFMS"
}
```

Para obtener más información, consulte [Configuración de Amazon SNS Notifications y Amazon CloudWatch Alarms](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [GetNotificationChannel](#) la Referencia de AWS CLI comandos.



## get-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-policy`.

### AWS CLI

Para recuperar una política de Firewall Manager

En el siguiente `get-policy` ejemplo, se recupera la política con el ID especificado.

```
aws fms get-policy \  
  --policy-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "Policy": {  
    "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "PolicyName": "test",  
    "PolicyUpdateToken": "1:p+2RpKR4wPFx7mcrL1U0QQ==",  
    "SecurityServicePolicyData": {  
      "Type": "SECURITY_GROUPS_COMMON",  
      "ManagedServiceData": "{\"type\":\"SECURITY_GROUPS_COMMON\",  
\"revertManualSecurityGroupChanges\":true,\"exclusiveResourceSecurityGroupManagement  
\":false,\"securityGroups\":[{\\"id\":\\"sg-045c43ccc9724e63e\"]]}"  
    },  
    "ResourceType": "AWS::EC2::Instance",  
    "ResourceTags": [],  
    "ExcludeResourceTags": false,  
    "RemediationEnabled": false  
  },  
  "PolicyArn": "arn:aws:fms:us-west-2:123456789012:policy/d1ac59b8-938e-42b3-  
b2e0-7c620422ddc2"  
}
```

Para obtener más información, consulte [Uso de políticas de AWS Firewall Manager](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [GetPolicy](#) la Referencia de AWS CLI comandos.

## list-compliance-status

En el siguiente ejemplo de código, se muestra cómo usar `list-compliance-status`.

## AWS CLI

Para recuperar la información de cumplimiento de las políticas de las cuentas de los miembros

En el siguiente `list-compliance-status` ejemplo, se recupera la información de conformidad de las cuentas de los miembros para la política especificada.

```
aws fms list-compliance-status \  
  --policy-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "PolicyComplianceStatusList": [  
    {  
      "PolicyOwner": "123456789012",  
      "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "PolicyName": "test",  
      "MemberAccount": "123456789012",  
      "EvaluationResults": [  
        {  
          "ComplianceStatus": "COMPLIANT",  
          "ViolatorCount": 0,  
          "EvaluationLimitExceeded": false  
        },  
        {  
          "ComplianceStatus": "NON_COMPLIANT",  
          "ViolatorCount": 2,  
          "EvaluationLimitExceeded": false  
        }  
      ],  
      "LastUpdated": 1576283774.0,  
      "IssueInfoMap": {}  
    }  
  ]  
}
```

Para obtener más información, consulte Cómo [comprobar la conformidad de los recursos con una política](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [ListComplianceStatus](#) la Referencia de AWS CLI comandos.

## list-member-accounts

En el siguiente ejemplo de código, se muestra cómo usar `list-member-accounts`.

### AWS CLI

Para recuperar las cuentas de los miembros de la organización

En el siguiente `list-member-accounts` ejemplo, se enumeran todas las cuentas de los miembros que se encuentran en la organización del administrador del Firewall Manager.

```
aws fms list-member-accounts
```

Salida:

```
{
  "MemberAccounts": [
    "222222222222",
    "333333333333",
    "444444444444"
  ]
}
```

Para obtener más información, consulte [AWS Firewall Manager](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager and AWS Shield.

- Para API obtener más información, consulte [ListMemberAccounts](#) la Referencia de AWS CLI comandos.

## list-policies

En el siguiente ejemplo de código, se muestra cómo usar `list-policies`.

### AWS CLI

Para recuperar todas las políticas de Firewall Manager

En el siguiente `list-policies` ejemplo, se recupera la lista de políticas de la cuenta. En este ejemplo, la salida está limitada a dos resultados por solicitud. Cada llamada devuelve un valor `NextToken` que se puede utilizar como valor del `--starting-token` parámetro en la siguiente `list-policies` llamada para obtener el siguiente conjunto de resultados de la lista.

```
aws fms list-policies \  
  --max-items 2
```

Salida:

```
{  
  "PolicyList": [  
    {  
      "PolicyArn": "arn:aws:fms:us-west-2:123456789012:policy/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "PolicyName": "test",  
      "ResourceType": "AWS::EC2::Instance",  
      "SecurityServiceType": "SECURITY_GROUPS_COMMON",  
      "RemediationEnabled": false  
    },  
    {  
      "PolicyArn": "arn:aws:fms:us-west-2:123456789012:policy/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "PolicyId": "457c9b21-fc94-406c-ae63-21217395ba72",  
      "PolicyName": "test",  
      "ResourceType": "AWS::EC2::Instance",  
      "SecurityServiceType": "SECURITY_GROUPS_COMMON",  
      "RemediationEnabled": false  
    }  
  ],  
  "NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="  
}
```

Para obtener más información, consulte [Uso de políticas de AWS Firewall Manager](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [ListPolicies](#) la Referencia de AWS CLI comandos.

## put-notification-channel

En el siguiente ejemplo de código, se muestra cómo usar `put-notification-channel`.

### AWS CLI

Para configurar la información del SNS tema para los registros del Firewall Manager

En el siguiente `put-notification-channel` ejemplo, se establece la información del SNS tema.

```
aws fms put-notification-channel \  
  --sns-topic-arn arn:aws:sns:us-west-2:123456789012:us-west-2-fms \  
  --sns-role-name arn:aws:iam::123456789012:role/aws-service-role/  
fms.amazonaws.com/AWSServiceRoleForFMS
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de Amazon SNS Notifications y Amazon CloudWatch Alarms](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [PutNotificationChannel](#) la Referencia de AWS CLI comandos.

## put-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-policy`.

### AWS CLI

Para crear una política de Firewall Manager

En el siguiente `put-policy` ejemplo, se crea una política de grupo de seguridad de Firewall Manager.

```
aws fms put-policy \  
  --cli-input-json file://policy.json
```

Contenidos de `policy.json`:

```
{  
  "Policy": {  
    "PolicyName": "test",  
    "SecurityServicePolicyData": {  
      "Type": "SECURITY_GROUPS_USAGE_AUDIT",  
      "ManagedServiceData": "{\"type\":\"SECURITY_GROUPS_USAGE_AUDIT\",  
\"deleteUnusedSecurityGroups\":false,\"coalesceRedundantSecurityGroups\":true}"
```

```

    },
    "ResourceType": "AWS::EC2::SecurityGroup",
    "ResourceTags": [],
    "ExcludeResourceTags": false,
    "RemediationEnabled": false
  },
  "TagList": [
    {
      "Key": "foo",
      "Value": "foo"
    }
  ]
}

```

Salida:

```

{
  "Policy": {
    "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "PolicyName": "test",
    "PolicyUpdateToken": "1:X9QGexP7HASDlsFp+G31Iw==",
    "SecurityServicePolicyData": {
      "Type": "SECURITY_GROUPS_USAGE_AUDIT",
      "ManagedServiceData": "{\"type\":\"SECURITY_GROUPS_USAGE_AUDIT\",
\\\"deleteUnusedSecurityGroups\\\":false,\\\"coalesceRedundantSecurityGroups\\\":true,
\\\"optionalDelayForUnusedInMinutes\\\":null}"
    },
    "ResourceType": "AWS::EC2::SecurityGroup",
    "ResourceTags": [],
    "ExcludeResourceTags": false,
    "RemediationEnabled": false
  },
  "PolicyArn": "arn:aws:fms:us-west-2:123456789012:policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}

```

Para obtener más información, consulte [Uso de políticas de AWS Firewall Manager](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [PutPolicy](#) la Referencia de AWS CLI comandos.

## AWS FIS ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS FIS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-experiment-template**

En el siguiente ejemplo de código, se muestra cómo usar `create-experiment-template`.

AWS CLI

Para crear una plantilla de experimento

En el siguiente `create-experiment-template` ejemplo, se crea una plantilla de experimento en tu AWS FIS cuenta.

```
aws fis create-experiment-template \  
  --cli-input-json file://myfile.json
```

Contenidos de `myfile.json`:

```
{  
  "description": "experimentTemplate",  
  "stopConditions": [  
    {  
      "source": "aws:cloudwatch:alarm",  
      "value": "arn:aws:cloudwatch:us-west-2:123456789012:alarm:alarmName"  
    }  
  ]  
}
```

```

],
"targets": {
  "Instances-Target-1": {
    "resourceType": "aws:ec2:instance",
    "resourceArns": [
      "arn:aws:ec2:us-west-2:123456789012:instance/i-12a3b4c56d78e9012"
    ],
    "selectionMode": "ALL"
  }
},
"actions": {
  "reboot": {
    "actionId": "aws:ec2:reboot-instances",
    "description": "reboot",
    "parameters": {},
    "targets": {
      "Instances": "Instances-Target-1"
    }
  }
},
"roleArn": "arn:aws:iam::123456789012:role/myRole"
}

```

**Salida:**

```

{
  "experimentTemplate": {
    "id": "ABCDE1fgHIJkLmNop",
    "description": "experimentTemplate",
    "targets": {
      "Instances-Target-1": {
        "resourceType": "aws:ec2:instance",
        "resourceArns": [
          "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
        ],
        "selectionMode": "ALL"
      }
    },
    "actions": {
      "reboot": {
        "actionId": "aws:ec2:reboot-instances",
        "description": "reboot",

```



```

        "parameters": {},
        "targets": {
            "Instances": "Instances-Target-1"
        }
    },
    "stopConditions": [
        {
            "source": "aws:cloudwatch:alarm",
            "value": "arn:aws:cloudwatch:us-west-2:123456789012:alarm:alarmName"
        }
    ],
    "creationTime": 1616434850.659,
    "lastUpdateTime": 1616434850.659,
    "roleArn": "arn:aws:iam::123456789012:role/myRole",
    "tags": {}
}
}

```

Para obtener más información, consulte [Crear una plantilla de experimento](#) en la Guía del usuario del simulador de inyección de AWS fallas.

- Para API obtener más información, consulte [CreateExperimentTemplate](#) la Referencia de AWS CLI comandos.

## delete-experiment-template

En el siguiente ejemplo de código, se muestra cómo usar delete-experiment-template.

### AWS CLI

Para eliminar una plantilla de experimento

En el siguiente delete-experiment-template ejemplo, se elimina la plantilla de experimento especificada.

```
aws fis delete-experiment-template \
  --id ABCDE1fgHIJKLmNop
```

Salida:

```
{
```

```

"experimentTemplate": {
  "id": "ABCDE1fgHIJKLmNop",
  "description": "myExperimentTemplate",
  "targets": {
    "Instances-Target-1": {
      "resourceType": "aws:ec2:instance",
      "resourceArns": [
        "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
      ],
      "selectionMode": "ALL"
    }
  },
  "actions": {
    "testaction": {
      "actionId": "aws:ec2:stop-instances",
      "parameters": {},
      "targets": {
        "Instances": "Instances-Target-1"
      }
    }
  },
  "stopConditions": [
    {
      "source": "none"
    }
  ],
  "creationTime": 1616017191.124,
  "lastUpdateTime": 1616017859.607,
  "roleArn": "arn:aws:iam::123456789012:role/FISRole"
}
}

```

Para obtener más información, consulte [Eliminar una plantilla de experimento](#) en la Guía del usuario del simulador de inyección de AWS fallas.

- Para API obtener más información, consulte [DeleteExperimentTemplate](#) la Referencia de AWS CLI comandos.

## get-action

En el siguiente ejemplo de código, se muestra cómo usar `get-action`.

## AWS CLI

Para obtener los detalles de la acción

En el `get-action` ejemplo siguiente se obtienen los detalles de la acción especificada.

```
aws fis get-action \  
  --id aws:ec2:stop-instances
```

Salida:

```
{  
  "action": {  
    "id": "aws:ec2:stop-instances",  
    "description": "Stop the specified EC2 instances.",  
    "parameters": {  
      "startInstancesAfterDuration": {  
        "description": "The time to wait before restarting the instances  
(ISO 8601 duration).",  
        "required": false  
      }  
    },  
    "targets": {  
      "Instances": {  
        "resourceType": "aws:ec2:instance"  
      }  
    },  
    "tags": {}  
  }  
}
```

Para obtener más información, consulte [Acciones](#) en la guía del usuario del simulador de inyección de AWS fallos.

- Para API obtener más información, consulte [GetAction](#) la Referencia de AWS CLI comandos.

## **get-experiment-template**

En el siguiente ejemplo de código, se muestra cómo usar `get-experiment-template`.

## AWS CLI

Para obtener los detalles de la plantilla del experimento

En el siguiente `get-experiment-template` ejemplo, se obtienen los detalles de la plantilla de experimento especificada.

```
aws fis get-experiment-template \  
  --id ABCDE1fgHIJkLmNop
```

Salida:

```
{  
  "experimentTemplate": {  
    "id": "ABCDE1fgHIJkLmNop",  
    "description": "myExperimentTemplate",  
    "targets": {  
      "Instances-Target-1": {  
        "resourceType": "aws:ec2:instance",  
        "resourceArns": [  
          "arn:aws:ec2:us-west-2:123456789012:instance/  
i-12a3b4c56d78e9012"  
        ],  
        "selectionMode": "ALL"  
      }  
    },  
    "actions": {  
      "testaction": {  
        "actionId": "aws:ec2:stop-instances",  
        "parameters": {},  
        "targets": {  
          "Instances": "Instances-Target-1"  
        }  
      }  
    },  
    "stopConditions": [  
      {  
        "source": "none"  
      }  
    ],  
    "creationTime": 1616017191.124,  
    "lastUpdateTime": 1616017331.51,  
    "roleArn": "arn:aws:iam::123456789012:role/FISRole",  
    "tags": {  
      "key": "value"  
    }  
  }  
}
```

```
}
```

Para obtener más información, consulte [las plantillas de experimentos](#) en la Guía del usuario del simulador de inyección de AWS fallas.

- Para API obtener más información, consulte [GetExperimentTemplate](#) la Referencia de AWS CLI comandos.

## get-experiment

En el siguiente ejemplo de código, se muestra cómo usar `get-experiment`.

### AWS CLI

Para obtener detalles del experimento

En el siguiente `get-experiment` ejemplo se obtienen los detalles del experimento especificado.

```
aws fis get-experiment \  
  --id ABC12DeFGhI3jKLMNOP
```

Salida:

```
{  
  "experiment": {  
    "id": "ABC12DeFGhI3jKLMNOP",  
    "experimentTemplateId": "ABCDE1fgHIJkLmNop",  
    "roleArn": "arn:aws:iam::123456789012:role/myRole",  
    "state": {  
      "status": "completed",  
      "reason": "Experiment completed."  
    },  
    "targets": {  
      "Instances-Target-1": {  
        "resourceType": "aws:ec2:instance",  
        "resourceArns": [  
          "arn:aws:ec2:us-west-2:123456789012:instance/  
i-12a3b4c56d78e9012"  
        ],  
        "selectionMode": "ALL"  
      }  
    },  
  },  
}
```

```
    "actions": {
      "reboot": {
        "actionId": "aws:ec2:reboot-instances",
        "parameters": {},
        "targets": {
          "Instances": "Instances-Target-1"
        },
        "state": {
          "status": "completed",
          "reason": "Action was completed."
        }
      }
    },
    "stopConditions": [
      {
        "source": "none"
      }
    ],
    "creationTime": 1616432509.662,
    "startTime": 1616432509.962,
    "endTime": 1616432522.307,
    "tags": {}
  }
}
```

Para obtener más información, consulte la guía del usuario de [Experiments for AWS FIS](#) the AWS Fault Injection Simulator.

- Para API obtener más información, consulte [GetExperiment](#) la Referencia de AWS CLI comandos.

## list-actions

En el siguiente ejemplo de código, se muestra cómo usar `list-actions`.

### AWS CLI

Para enumerar las acciones

En el siguiente `list-actions` ejemplo, se enumeran las acciones disponibles.

```
aws fis list-actions
```

**Salida:**

```
{
  "actions": [
    {
      "id": "aws:ec2:reboot-instances",
      "description": "Reboot the specified EC2 instances.",
      "targets": {
        "Instances": {
          "resourceType": "aws:ec2:instance"
        }
      },
      "tags": {}
    },
    {
      "id": "aws:ec2:stop-instances",
      "description": "Stop the specified EC2 instances.",
      "targets": {
        "Instances": {
          "resourceType": "aws:ec2:instance"
        }
      },
      "tags": {}
    },
    {
      "id": "aws:ec2:terminate-instances",
      "description": "Terminate the specified EC2 instances.",
      "targets": {
        "Instances": {
          "resourceType": "aws:ec2:instance"
        }
      },
      "tags": {}
    },
    {
      "id": "aws:ecs:drain-container-instances",
      "description": "Drain percentage of underlying EC2 instances on an ECS
cluster.",
      "targets": {
        "Clusters": {
          "resourceType": "aws:ecs:cluster"
        }
      },
      "tags": {}
    }
  ]
}
```

```
    },
    {
      "id": "aws:eks:terminate-nodegroup-instances",
      "description": "Terminates a percentage of the underlying EC2 instances
in an EKS cluster.",
      "targets": {
        "Nodegroups": {
          "resourceType": "aws:eks:nodegroup"
        }
      },
      "tags": {}
    },
    {
      "id": "aws:fis:inject-api-internal-error",
      "description": "Cause an AWS service to return internal error responses
for specific callers and operations.",
      "targets": {
        "Roles": {
          "resourceType": "aws:iam:role"
        }
      },
      "tags": {}
    },
    {
      "id": "aws:fis:inject-api-throttle-error",
      "description": "Cause an AWS service to return throttled responses for
specific callers and operations.",
      "targets": {
        "Roles": {
          "resourceType": "aws:iam:role"
        }
      },
      "tags": {}
    },
    {
      "id": "aws:fis:inject-api-unavailable-error",
      "description": "Cause an AWS service to return unavailable error
responses for specific callers and operations.",
      "targets": {
        "Roles": {
          "resourceType": "aws:iam:role"
        }
      },
      "tags": {}
    }
  ],
  "tags": {}
}
```



```
    },
    {
      "id": "aws:fis:wait",
      "description": "Wait for the specified duration. Stop condition
monitoring will continue during this time.",
      "tags": {}
    },
    {
      "id": "aws:rds:failover-db-cluster",
      "description": "Failover a DB Cluster to one of the replicas.",
      "targets": {
        "Clusters": {
          "resourceType": "aws:rds:cluster"
        }
      },
      "tags": {}
    },
    {
      "id": "aws:rds:reboot-db-instances",
      "description": "Reboot the specified DB instances.",
      "targets": {
        "DBInstances": {
          "resourceType": "aws:rds:db"
        }
      },
      "tags": {}
    },
    {
      "id": "aws:ssm:send-command",
      "description": "Run the specified SSM document.",
      "targets": {
        "Instances": {
          "resourceType": "aws:ec2:instance"
        }
      },
      "tags": {}
    }
  ]
}
```

Para obtener más información, consulte [Acciones](#) en la guía del usuario del simulador de inyección de AWS fallas.

- Para API obtener más información, consulte [ListActions](#) la Referencia de AWS CLI comandos.

## list-experiment-templates

En el siguiente ejemplo de código, se muestra cómo usar `list-experiment-templates`.

### AWS CLI

Para enumerar las plantillas de experimentos

En el siguiente `list-experiment-templates` ejemplo, se muestran las plantillas de experimentos de tu AWS cuenta.

```
aws fis list-experiment-templates
```

Salida:

```
{
  "experimentTemplates": [
    {
      "id": "ABCDE1fgHIJkLmNop",
      "description": "myExperimentTemplate",
      "creationTime": 1616017191.124,
      "lastUpdateTime": 1616017191.124,
      "tags": {
        "key": "value"
      }
    }
  ]
}
```

Para obtener más información, consulte [las plantillas de experimentos](#) en la guía del usuario del simulador de inyección de AWS fallas.

- Para API obtener más información, consulte [ListExperimentTemplates](#) la Referencia de AWS CLI comandos.

## list-experiments

En el siguiente ejemplo de código, se muestra cómo usar `list-experiments`.

### AWS CLI

Para enumerar los experimentos

En el siguiente `list-experiments` ejemplo, se enumeran los experimentos de tu AWS cuenta.

```
aws fis list-experiments
```

Salida:

```
{
  "experiments": [
    {
      "id": "ABCdeF1GHijKLM23NO",
      "experimentTemplateId": "ABCDE1fgHIJkLmNop",
      "state": {
        "status": "running",
        "reason": "Experiment is running."
      },
      "creationTime": 1616017341.197,
      "tags": {
        "key": "value"
      }
    }
  ]
}
```

Para obtener más información, consulte la Guía del usuario de [Experimentos](#) en el simulador de inyección de AWS fallas.

- Para API obtener más información, consulte [ListExperiments](#) la Referencia de AWS CLI comandos.

## **list-tags-for-resource**

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

AWS CLI

Para enumerar las etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas del recurso especificado.

```
aws fis list-tags-for-resource \
```

```
--resource-arn arn:aws:fis:us-west-2:123456789012:experiment/ABC12DeFGhI3jKLMNOP
```

Salida:

```
{
  "tags": {
    "key1": "value1",
    "key2": "value2"
  }
}
```

Para obtener más información, consulte [Etiquete sus AWS FIS recursos](#) en la Guía del usuario del simulador de inyección de AWS fallas.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## start-experiment

En el siguiente ejemplo de código, se muestra cómo usar `start-experiment`.

AWS CLI

Para iniciar un experimento

En el siguiente `start-experiment` ejemplo se inicia el experimento especificado.

```
aws fis start-experiment \
  --experiment-template-id ABCDE1fgHIJkLmNop
```

Salida:

```
{
  "experiment": {
    "id": "ABC12DeFGhI3jKLMNOP",
    "experimentTemplateId": "ABCDE1fgHIJkLmNop",
    "roleArn": "arn:aws:iam::123456789012:role/myRole",
    "state": {
      "status": "initiating",
      "reason": "Experiment is initiating."
    }
  },
}
```

```
    "targets": {
      "Instances-Target-1": {
        "resourceType": "aws:ec2:instance",
        "resourceArns": [
          "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
        ],
        "selectionMode": "ALL"
      }
    },
    "actions": {
      "reboot": {
        "actionId": "aws:ec2:reboot-instances",
        "parameters": {},
        "targets": {
          "Instances": "Instances-Target-1"
        },
        "state": {
          "status": "pending",
          "reason": "Initial state"
        }
      }
    },
    "stopConditions": [
      {
        "source": "none"
      }
    ],
    "creationTime": 1616432464.025,
    "startTime": 1616432464.374,
    "tags": {}
  }
}
```

Para obtener más información, consulte la guía del usuario AWS FIS de [Experimentos para el simulador de inyección de AWS fallas](#).

- Para API obtener más información, consulte [StartExperiment](#) la Referencia de AWS CLI comandos.

## stop-experiment

En el siguiente ejemplo de código, se muestra cómo usar stop-experiment.

## AWS CLI

Para detener un experimento

El siguiente stop-experiment ejemplo detiene la ejecución del experimento especificado.

```
aws fis stop-experiment \  
  --id ABC12DeFGhI3jKLMNOP
```

Salida:

```
{  
  "experiment": {  
    "id": "ABC12DeFGhI3jKLMNOP",  
    "experimentTemplateId": "ABCDE1fgHIJkLmNop",  
    "roleArn": "arn:aws:iam::123456789012:role/myRole",  
    "state": {  
      "status": "stopping",  
      "reason": "Stopping Experiment."  
    },  
    "targets": {  
      "Instances-Target-1": {  
        "resourceType": "aws:ec2:instance",  
        "resourceArns": [  
          "arn:aws:ec2:us-west-2:123456789012:instance/  
i-12a3b4c56d78e9012"  
        ],  
        "selectionMode": "ALL"  
      }  
    },  
    "actions": {  
      "reboot": {  
        "actionId": "aws:ec2:reboot-instances",  
        "parameters": {},  
        "targets": {  
          "Instances": "Instances-Target-1"  
        },  
        "startAfter": [  
          "wait"  
        ],  
        "state": {  
          "status": "pending",  
          "reason": "Initial state."  
        }  
      }  
    }  
  }  
}
```

```

    }
  },
  "wait": {
    "actionId": "aws:fis:wait",
    "parameters": {
      "duration": "PT5M"
    },
    "state": {
      "status": "running",
      "reason": ""
    }
  }
},
"stopConditions": [
  {
    "source": "none"
  }
],
"creationTime": 1616432680.927,
"startTime": 1616432681.177,
"tags": {}
}
}

```

Para obtener más información, consulte la guía del usuario de [Experiments for AWS FIS](#) the AWS Fault Injection Simulator.

- Para API obtener más información, consulte [StopExperiment](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

### AWS CLI

Para etiquetar un recurso

El siguiente tag-resource ejemplo etiqueta el recurso especificado.

```

aws fis tag-resource \
  --resource-arn arn:aws:fis:us-west-2:123456789012:experiment/ABC12DeFGhI3jKLMNOP \

```

```
--tags key1=value1, key2=value2
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquete sus AWS FIS recursos](#) en la Guía del usuario del simulador de inyección de AWS fallas.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

AWS CLI

Para quitar la etiqueta de un recurso

En el siguiente `untag-resource` ejemplo, se eliminan las etiquetas del recurso especificado.

```
aws fis untag-resource \  
  --resource-arn arn:aws:fis:us-west-2:123456789012:experiment/ABC12DeFGhI3jKLMNOP
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquete sus AWS FIS recursos](#) en la Guía del usuario del simulador de inyección de AWS fallas.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-experiment-template

En el siguiente ejemplo de código, se muestra cómo usar `update-experiment-template`.

AWS CLI

Para actualizar una plantilla de experimento

El siguiente `update-experiment-template` ejemplo actualiza la descripción de la plantilla de experimento especificada.



```
aws fis update-experiment-template \  
  --id ABCDE1fgHIJkLmNop \  
  ---description myExperimentTemplate
```

Salida:

```
{  
  "experimentTemplate": {  
    "id": "ABCDE1fgHIJkLmNop",  
    "description": "myExperimentTemplate",  
    "targets": {  
      "Instances-Target-1": {  
        "resourceType": "aws:ec2:instance",  
        "resourceArns": [  
          "arn:aws:ec2:us-west-2:123456789012:instance/  
i-12a3b4c56d78e9012"  
        ],  
        "selectionMode": "ALL"  
      }  
    },  
    "actions": {  
      "testaction": {  
        "actionId": "aws:ec2:stop-instances",  
        "parameters": {},  
        "targets": {  
          "Instances": "Instances-Target-1"  
        }  
      }  
    },  
    "stopConditions": [  
      {  
        "source": "none"  
      }  
    ],  
    "creationTime": 1616017191.124,  
    "lastUpdateTime": 1616017859.607,  
    "roleArn": "arn:aws:iam::123456789012:role/FISRole",  
    "tags": {  
      "key": "value"  
    }  
  }  
}
```

Para obtener más información, consulte [Actualizar una plantilla de experimento](#) en la Guía del usuario del simulador de inyección de AWS fallas.

- Para API obtener más información, consulte [UpdateExperimentTemplate](#) la Referencia de AWS CLI comandos.

## GameLift Ejemplos de Amazon que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Amazon GameLift.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-build**

En el siguiente ejemplo de código, se muestra cómo usar `create-build`.

AWS CLI

Ejemplo 1: Para crear una compilación de juegos a partir de los archivos de un bucket de S3

En el siguiente `create-build` ejemplo, se crea un recurso de creación de juegos personalizado. Utiliza archivos comprimidos que se almacenan en una ubicación S3 en una AWS cuenta que tú controlas. En este ejemplo, se supone que ya has creado un IAM rol que da GameLift permiso a Amazon para acceder a la ubicación de S3. Como la solicitud no especifica un sistema operativo, el recurso de nueva compilación se establece de forma predeterminada en `WINDOWS_2012`.

```
aws gamelift create-build \  
  --storage-location file://storage-loc.json \  
  --name MegaFrogRaceServer.NA \  
  --operating-system WINDOWS_2012
```

```
--build-version 12345.678
```

Contenidos de storage-loc.json:

```
{
  "Bucket": "MegaFrogRaceServer_NA_build_files"
  "Key": "MegaFrogRaceServer_build_123.zip"
  "RoleArn": "arn:aws:iam::123456789012:role/gamelift"
}
```

Salida:

```
{
  "Build": {
    "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "CreationTime": 1496708916.18,
    "Name": "MegaFrogRaceServer.NA",
    "OperatingSystem": "WINDOWS_2012",
    "SizeOnDisk": 479303,
    "Status": "INITIALIZED",
    "Version": "12345.678"
  },
  "StorageLocation": {
    "Bucket": "MegaFrogRaceServer_NA_build_files",
    "Key": "MegaFrogRaceServer_build_123.zip"
  }
}
```

Ejemplo 2: Para crear un recurso de compilación de juegos para cargar archivos manualmente en GameLift

En el siguiente create-build ejemplo, se crea un nuevo recurso de compilación. También cuenta con una ubicación de almacenamiento y credenciales temporales que le permiten cargar manualmente la versión del juego a la GameLift ubicación de Amazon S3. Una vez que haya cargado correctamente la compilación, el GameLift servicio la validará y actualizará el estado de la nueva.

```
aws gamelift create-build \  
  --name MegaFrogRaceServer.NA \  
  --build-version 12345.678 \  
  --storage-location storage-loc \  
  --role-arn arn:aws:iam::123456789012:role/gamelift
```

```
--operating-system AMAZON_LINUX
```

Salida:

```
{
  "Build": {
    "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "CreationTime": 1496708916.18,
    "Name": "MegaFrogRaceServer.NA",
    "OperatingSystem": "AMAZON_LINUX",
    "SizeOnDisk": 0,
    "Status": "INITIALIZED",
    "Version": "12345.678"
  },
  "StorageLocation": {
    "Bucket": "gamelift-builds-us-west-2",
    "Key": "123456789012/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  },
  "UploadCredentials": {
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY",
    "SessionToken": "AgoGb3JpZ2luENZ...EXAMPLETOKEN=="
  }
}
```

Para obtener más información, consulta [Cómo subir una versión de servidor personalizada GameLift en la Guía para GameLift desarrolladores](#) de Amazon.

- Para API obtener más información, consulte [CreateBuild](#) la Referencia de AWS CLI comandos.

## create-fleet

En el siguiente ejemplo de código, se muestra cómo usar `create-fleet`.

### AWS CLI

Ejemplo 1: Para crear una flota básica de Linux

En el siguiente `create-fleet` ejemplo, se crea una flota de instancias de Linux bajo demanda con una configuración mínima para alojar una compilación de servidor personalizada. Para completar la configuración, utilice `update-fleet`.

```
aws gamelift create-fleet \
  --name MegaFrogRaceServer.NA.v2 \
  --description 'Hosts for v2 North America' \
  --build-id build-1111aaaa-22bb-33cc-44dd-5555eeee66ff \
  --certificate-configuration 'CertificateType=GENERATED' \
  --ec2-instance-type c4.large \
  --fleet-type ON_DEMAND \
  --runtime-configuration 'ServerProcesses=[{LaunchPath=/local/game/release-na/MegaFrogRace_Server.exe,ConcurrentExecutions=1}]'
```

Salida:

```
{
  "FleetAttributes": {
    "BuildId": "build-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
    "CertificateConfiguration": {
      "CertificateType": "GENERATED"
    },
    "CreationTime": 1496365885.44,
    "Description": "Hosts for v2 North America",
    "FleetArn": "arn:aws:gamelift:us-west-2:444455556666:fleet/fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "FleetType": "ON_DEMAND",
    "InstanceType": "c4.large",
    "MetricGroups": ["default"],
    "Name": "MegaFrogRace.NA.v2",
    "NewGameSessionProtectionPolicy": "NoProtection",
    "OperatingSystem": "AMAZON_LINUX",
    "ServerLaunchPath": "/local/game/release-na/MegaFrogRace_Server.exe",
    "Status": "NEW"
  }
}
```

Ejemplo 2: Para crear una flota básica de Windows

En el siguiente create-fleet ejemplo, se crea una flota de instancias puntuales de Windows con una configuración mínima para alojar una compilación de servidor personalizada. Para completar la configuración, utilice update-fleet.

```
aws gamelift create-fleet \
  --name MegaFrogRace.NA.v2 \
```

```

--description 'Hosts for v2 North America' \
--build-id build-2222aaaa-33bb-44cc-55dd-6666eeee77ff \
--certificate-configuration 'CertificateType=GENERATED' \
--ec2-instance-type c4.large \
--fleet-type SPOT \
--runtime-configuration 'ServerProcesses=[{LaunchPath=C:\game
\Bin64.Release.Dedicated\MegaFrogRace_Server.exe, ConcurrentExecutions=1}]'

```

Salida:

```

{
  "FleetAttributes": {
    "BuildId": "build-2222aaaa-33bb-44cc-55dd-6666eeee77ff",
    "CertificateConfiguration": {
      "CertificateType": "GENERATED"
    },
    "CreationTime": 1496365885.44,
    "Description": "Hosts for v2 North America",
    "FleetArn": "arn:aws:gamelift:us-west-2:444455556666:fleet/
fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "FleetType": "SPOT",
    "InstanceType": "c4.large",
    "MetricGroups": ["default"],
    "Name": "MegaFrogRace.NA.v2",
    "NewGameSessionProtectionPolicy": "NoProtection",
    "OperatingSystem": "WINDOWS_2012",
    "ServerLaunchPath": "C:\game\Bin64.Release.Dedicated
\MegaFrogRace_Server.exe",
    "Status": "NEW"
  }
}

```

Ejemplo 3: Para crear una flota completamente configurada

En el siguiente create-fleet ejemplo, se crea una flota de instancias de Spot Windows para una versión de servidor personalizada, con los ajustes de configuración más utilizados.

```

aws gamelift create-fleet \
  --name MegaFrogRace.NA.v2 \
  --description 'Hosts for v2 North America' \
  --build-id build-2222aaaa-33bb-44cc-55dd-6666eeee77ff \
  --certificate-configuration 'CertificateType=GENERATED' \

```

```

--ec2-instance-type c4.large \
--ec2-inbound-permissions
'FromPort=33435,ToPort=33435,IpRange=10.24.34.0/23,Protocol=UDP' \
--fleet-type SPOT \
--new-game-session-protection-policy FullProtection \
--runtime-configuration file://runtime-config.json \
--metric-groups default \
--instance-role-arn 'arn:aws:iam::444455556666:role/GameLiftS3Access'

```

Contenidos de runtime-config.json:

```

GameSessionActivationTimeoutSeconds=300,
MaxConcurrentGameSessionActivations=2,
ServerProcesses=[
  {LaunchPath=C:\game\Bin64.Release.Dedicated\MegaFrogRace_Server.exe,Parameters=-
debug,ConcurrentExecutions=1},
  {LaunchPath=C:\game\Bin64.Release.Dedicated
\MegaFrogRace_Server.exe,ConcurrentExecutions=1}]

```

Salida:

```

{
  "FleetAttributes": {
    "InstanceRoleArn": "arn:aws:iam::444455556666:role/GameLiftS3Access",
    "Status": "NEW",
    "InstanceType": "c4.large",
    "FleetArn": "arn:aws:gamelift:us-west-2:444455556666:fleet/
fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "Description": "Hosts for v2 North America",
    "FleetType": "SPOT",
    "OperatingSystem": "WINDOWS_2012",
    "Name": "MegaFrogRace.NA.v2",
    "CreationTime": 1569309011.11,
    "MetricGroups": [
      "default"
    ],
    "BuildId": "build-2222aaaa-33bb-44cc-55dd-6666eeee77ff",
    "ServerLaunchParameters": "abc",
    "ServerLaunchPath": "C:\\game\\Bin64.Release.Dedicated\\
MegaFrogRace_Server.exe",
    "NewGameSessionProtectionPolicy": "FullProtection",
    "CertificateConfiguration": {

```

```

        "CertificateType": "GENERATED"
    }
}
}

```

#### Ejemplo 4: Para crear una flota de Realtime Servers

En el siguiente `create-fleet` ejemplo, se crea una flota de instancias puntuales con un script de configuración en tiempo real que se ha cargado en Amazon GameLift. Todos los servidores Realtime se implementan en máquinas Linux. A los efectos de este ejemplo, supongamos que el script Realtime cargado incluye varios archivos de script y que la `Init()` función se encuentra en el archivo de script se llama `MainScript.js`. Como se muestra, este archivo se identifica como el script de inicio en la configuración del tiempo de ejecución.

```

aws gamelift create-fleet \
  --name MegaFrogRace.NA.realtime \
  --description 'Mega Frog Race Realtime fleet' \
  --script-id script-1111aaaa-22bb-33cc-44dd-5555eeee66ff \
  --ec2-instance-type c4.large \
  --fleet-type SPOT \
  --certificate-configuration 'CertificateType=GENERATED' --runtime-configuration
'ServerProcesses=[{LaunchPath=/local/game/MainScript.js,Parameters=+map
Winter444,ConcurrentExecutions=5}]'

```

Salida:

```

{
  "FleetAttributes": {
    "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "Status": "NEW",
    "CreationTime": 1569310745.212,
    "InstanceType": "c4.large",
    "NewGameSessionProtectionPolicy": "NoProtection",
    "CertificateConfiguration": {
      "CertificateType": "GENERATED"
    },
    "Name": "MegaFrogRace.NA.realtime",
    "ScriptId": "script-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
    "FleetArn": "arn:aws:gamelift:us-west-2:444455556666:fleet/
fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "FleetType": "SPOT",
    "MetricGroups": [

```



```

        "default"
    ],
    "Description": "Mega Frog Race Realtime fleet",
    "OperatingSystem": "AMAZON_LINUX"
}
}

```

- Para API obtener más información, consulte [CreateFleet](#) la Referencia de AWS CLI comandos.

## create-game-session-queue

En el siguiente ejemplo de código, se muestra cómo usar `create-game-session-queue`.

### AWS CLI

Ejemplo 1: Para configurar una cola de sesiones de juego ordenada

El siguiente `create-game-session-queue` ejemplo crea una nueva cola de sesiones de juego con destinos en dos regiones. También configura la cola para que las sesiones de juego soliciten tiempo de espera después de esperar 10 minutos para ser colocadas. Como no se ha definido ninguna política de latencia, GameLift intenta colocar todas las sesiones de juego en el primer destino de la lista.

```

aws gamelift create-game-session-queue \
  --name MegaFrogRaceServer-NA \
  --destinations file://destinations.json \
  --timeout-in-seconds 600

```

Contenidos de `destinations.json`:

```

{
  "Destinations": [
    {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" },
    {"DestinationArn": "arn:aws:gamelift:us-west-1::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222" }
  ]
}

```

Salida:

```
{
  "GameSessionQueues": [
    {
      "Name": "MegaFrogRaceServer-NA",
      "GameSessionQueueArn": "arn:aws:gamelift:us-west-2:123456789012:gamesessionqueue/MegaFrogRaceServer-NA",
      "TimeoutInSeconds": 600,
      "Destinations": [
        {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"},
        {"DestinationArn": "arn:aws:gamelift:us-west-1::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"}
      ]
    }
  ]
}
```

Ejemplo 2: Para configurar una cola de sesiones de juego con las políticas de latencia de los jugadores

En el siguiente `create-game-session-queue` ejemplo, se crea una nueva cola de sesiones de juego con políticas de latencia para dos jugadores. La primera política establece un límite de latencia de 100 ms que se aplica durante el primer minuto del intento de ubicación de una sesión de juego. La segunda política aumenta el límite de latencia a 200 ms hasta que se agote el tiempo de espera de la solicitud de colocación, que es de 3 minutos.

```
aws gamelift create-game-session-queue \
  --name MegaFrogRaceServer-NA \
  --destinations file://destinations.json \
  --player-latency-policies file://latency-policies.json \
  --timeout-in-seconds 180
```

Contenidos de `destinations.json`:

```
{
  "Destinations": [
    { "DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" },
    { "DestinationArn": "arn:aws:gamelift:us-east-1::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222" }
  ]
}
```

```
}

```

Contenidos de `latency-policies.json`:

```
{
  "PlayerLatencyPolicies": [
    {"MaximumIndividualPlayerLatencyMilliseconds": 200},
    {"MaximumIndividualPlayerLatencyMilliseconds": 100, "PolicyDurationSeconds":
60}
  ]
}
```

Salida:

```
{
  "GameSessionQueue": {
    "Name": "MegaFrogRaceServer-NA",
    "GameSessionQueueArn": "arn:aws:gamelift:us-
west-2:111122223333:gamesessionqueue/MegaFrogRaceServer-NA",
    "TimeoutInSeconds": 600,
    "PlayerLatencyPolicies": [
      {
        "MaximumIndividualPlayerLatencyMilliseconds": 100,
        "PolicyDurationSeconds": 60
      },
      {
        "MaximumIndividualPlayerLatencyMilliseconds": 200
      }
    ]
    "Destinations": [
      {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"},
      {"DestinationArn": "arn:aws:gamelift:us-east-1::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"}
    ],
  }
}
```

Para obtener más información, consulta [Crear una cola](#) en la Guía para GameLift desarrolladores de Amazon.

- Para API obtener más información, consulte [CreateGameSessionQueue](#) la Referencia de AWS CLI comandos.

## delete-build

En el siguiente ejemplo de código, se muestra cómo usar `delete-build`.

### AWS CLI

Para eliminar una versión de juego personalizada

En el siguiente `delete-build` ejemplo, se elimina una compilación de tu GameLift cuenta de Amazon. Una vez eliminada la compilación, no podrás usarla para crear nuevas flotas. Esta operación no se puede deshacer.

```
aws gamelift delete-build \  
  --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteBuild](#) la Referencia de AWS CLI comandos.

## delete-fleet

En el siguiente ejemplo de código, se muestra cómo usar `delete-fleet`.

### AWS CLI

Para eliminar una flota que ya no está en uso

En el siguiente `delete-fleet` ejemplo, se elimina una flota que se ha reducido a cero instancias. Si la capacidad de la flota es superior a cero, la solicitud falla y arroja un error de HTTP 400.

```
aws gamelift delete-fleet \  
  --fleet-id Fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Administrar GameLift flotas](#) en la Guía para GameLift desarrolladores de Amazon.

- Para API obtener más información, consulte [DeleteFleet](#) la Referencia de AWS CLI comandos.

## delete-game-session-queue

En el siguiente ejemplo de código, se muestra cómo usar `delete-game-session-queue`.

### AWS CLI

Para eliminar una cola de sesiones de juego

En el siguiente `delete-game-session-queue` ejemplo, se elimina una cola de sesiones de juego especificada.

```
aws gamelift delete-game-session-queue \  
  --name MegaFrogRace-NA
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulta la Referencia [DeleteGameSessionQueue](#) de AWS CLI comandos.

## describe-build

En el siguiente ejemplo de código, se muestra cómo usar `describe-build`.

### AWS CLI

Para obtener información sobre una versión de juego personalizada

En el siguiente `describe-build` ejemplo, se recuperan las propiedades de un recurso de compilación de un servidor de juegos.

```
aws gamelift describe-build \  
  --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "Build": {  
    "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "CreationTime": 1496708916.18,
```

```
    "Name": "My_Game_Server_Build_One",
    "OperatingSystem": "AMAZON_LINUX",
    "SizeOnDisk": 1304924,
    "Status": "READY",
    "Version": "12345.678"
  }
}
```

Para obtener más información, consulta [Cómo subir una versión de servidor personalizada GameLift en la Guía para GameLift desarrolladores](#) de Amazon.

- Para API obtener más información, consulte [DescribeBuild](#) la Referencia de AWS CLI comandos.

## describe-ec2-instance-limits

En el siguiente ejemplo de código, se muestra cómo usar `describe-ec2-instance-limits`.

### AWS CLI

Para recuperar los límites de servicio de un tipo de EC2 instancia

En el siguiente `describe-ec2-instance-limits` ejemplo, se muestran las instancias máximas permitidas y las instancias actuales que se utilizan para el tipo de EC2 instancia especificado en la región actual. El resultado indica que solo se utilizan cinco de las veinte instancias permitidas.

```
aws gamelift describe-ec2-instance-limits \
  --ec2-instance-type m5.Large
```

Salida:

```
{
  "EC2InstanceLimits": [
    {
      "EC2InstanceType": "m5.large",
      "CurrentInstances": 5,
      "InstanceLimit": 20
    }
  ]
}
```

Para obtener más información, consulte [Choose Computing Resources](#) en la Guía para GameLift desarrolladores de Amazon.

- Para API obtener más información, consulte la [DescribeEcsección 2 InstanceLimits](#) en la Referencia de AWS CLI comandos.

## describe-fleet-attributes

En el siguiente ejemplo de código, se muestra cómo usar `describe-fleet-attributes`.

### AWS CLI

Ejemplo 1: Para ver los atributos de una lista de flotas

En el siguiente `describe-fleet-attributes` ejemplo, se recuperan los atributos de flota de dos flotas especificadas. Como se muestra, las flotas solicitadas se implementan con la misma compilación, una para las instancias bajo demanda y otra para las instancias puntuales, con algunas pequeñas diferencias de configuración.

```
aws gamelift describe-fleet-attributes \  
  --fleet-ids arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Salida:

```
{  
  "FleetAttributes": [  
    {  
      "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "FleetArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "FleetType": "ON_DEMAND",  
      "InstanceType": "c4.large",  
      "Description": "On-demand hosts for v2 North America",  
      "Name": "MegaFrogRaceServer.NA.v2-od",  
      "CreationTime": 1568836191.995,  
      "Status": "ACTIVE",  
      "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",  
      "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",  
      "ServerLaunchPath": "C:\\\\game\\\\MegaFrogRace_Server.exe",  
      "ServerLaunchParameters": "+gamelift_start_server",
```

```

    "NewGameSessionProtectionPolicy": "NoProtection",
    "OperatingSystem": "WINDOWS_2012",
    "MetricGroups": [
      "default"
    ],
    "CertificateConfiguration": {
      "CertificateType": "DISABLED"
    }
  },
  {
    "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "FleetArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222",
    "FleetType": "SPOT",
    "InstanceType": "c4.large",
    "Description": "On-demand hosts for v2 North America",
    "Name": "MegaFrogRaceServer.NA.v2-spot",
    "CreationTime": 1568838275.379,
    "Status": "ACTIVATING",
    "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE33333",
    "ServerLaunchPath": "C:\\\\game\\\\MegaFrogRace_Server.exe",
    "NewGameSessionProtectionPolicy": "NoProtection",
    "OperatingSystem": "WINDOWS_2012",
    "MetricGroups": [
      "default"
    ],
    "CertificateConfiguration": {
      "CertificateType": "GENERATED"
    }
  }
]
}

```

## Ejemplo 2: Para solicitar atributos para todas las flotas

A continuación se describe `fleet-attributes` muestran los atributos de todas las flotas con cualquier estado. Este ejemplo ilustra el uso de parámetros de paginación para devolver una flota a la vez.

```

aws gamelift describe-fleet-attributes \
  --limit 1

```



**Salida:**

```
{
  "FleetAttributes": [
    {
      "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "FleetArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "FleetType": "SPOT",
      "InstanceType": "c4.large",
      "Description": "On-demand hosts for v2 North America",
      "Name": "MegaFrogRaceServer.NA.v2-spot",
      "CreationTime": 1568838275.379,
      "Status": "ACTIVATING",
      "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
      "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
      "ServerLaunchPath": "C:\\\\game\\\\MegaFrogRace_Server.exe",
      "NewGameSessionProtectionPolicy": "NoProtection",
      "OperatingSystem": "WINDOWS_2012",
      "MetricGroups": [
        "default"
      ],
      "CertificateConfiguration": {
        "CertificateType": "GENERATED"
      }
    }
  ],
  "NextToken":
  "eyJhd3NBWY2NvdW50SWQlOmsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS"
}
```

El resultado incluye un NextToken valor que puede utilizar al llamar al comando por segunda vez. Pase el valor al --next-token parámetro para especificar dónde recoger la salida. El siguiente comando devuelve el segundo resultado de la salida.

```
aws gamelift describe-fleet-attributes \
  --limit 1 \
  --next-
token eyJhd3NBWY2NvdW50SWQlOmsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS
```

Repita el procedimiento hasta que la respuesta no incluya ningún NextToken valor.

Para obtener más información, consulte [Configuración de GameLift flotas](#) en la Guía para GameLift desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeFleetAttributes](#) la Referencia de AWS CLI comandos.

## describe-fleet-capacity

En el siguiente ejemplo de código, se muestra cómo usar `describe-fleet-capacity`.

### AWS CLI

Para ver el estado de la capacidad de una lista de flotas

El siguiente `describe-fleet-capacity` ejemplo recupera la capacidad actual de dos flotas especificadas.

```
aws gamelift describe-fleet-capacity \  
  --fleet-ids arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Salida:

```
{  
  "FleetCapacity": [  
    {  
      "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "InstanceType": "c5.large",  
      "InstanceCounts": {  
        "DESIRED": 10,  
        "MINIMUM": 1,  
        "MAXIMUM": 20,  
        "PENDING": 0,  
        "ACTIVE": 10,  
        "IDLE": 3,  
        "TERMINATING": 0  
      }  
    },  
    {  
      "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "InstanceType": "c5.large",  
      "InstanceCounts": {  
        "DESIRED": 13,
```

```

        "MINIMUM": 1,
        "MAXIMUM": 20,
        "PENDING": 0,
        "ACTIVE": 15,
        "IDLE": 2,
        "TERMINATING": 2
    }
}
]
}

```

Para obtener más información, consulta [GameLift Metrics for Fleets](#) en la Guía para GameLift desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeFleetCapacity](#) la Referencia de AWS CLI comandos.

## describe-fleet-events

En el siguiente ejemplo de código, se muestra cómo usar `describe-fleet-events`.

### AWS CLI

Para solicitar eventos durante un período de tiempo específico

En el siguiente `describe-fleet-events` ejemplo, se muestran los detalles de todos los eventos relacionados con la flota que se produjeron durante el período de tiempo especificado.

```

aws gamelift describe-fleet-events \
  --fleet-id arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --start-time 1579647600 \
  --end-time 1579649400 \
  --limit 5

```

Salida:

```

{
  "Events": [
    {
      "EventId": "a37b6892-5d07-4d3b-8b47-80244ecf66b9",
      "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",

```

```

    "EventCode": "FLEET_STATE_ACTIVE",
    "Message": "Fleet fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 changed
state to ACTIVE",
    "EventTime": 1579649342.191
  },
  {
    "EventId": "67da4ec9-92a3-4d95-886a-5d6772c24063",
    "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "EventCode": "FLEET_STATE_ACTIVATING",
    "Message": "Fleet fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 changed
state to ACTIVATING",
    "EventTime": 1579649321.427
  },
  {
    "EventId": "23813a46-a9e6-4a53-8847-f12e6a8381ac",
    "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "EventCode": "FLEET_STATE_BUILDING",
    "Message": "Fleet fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 changed
state to BUILDING",
    "EventTime": 1579649321.243
  },
  {
    "EventId": "3bf217d0-1d44-42f9-9202-433ed475d2e8",
    "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "EventCode": "FLEET_STATE_VALIDATING",
    "Message": "Fleet fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 changed
state to VALIDATING",
    "EventTime": 1579649197.449
  },
  {
    "EventId": "2ecd0130-5986-44eb-99a7-62df27741084",
    "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "EventCode": "FLEET_VALIDATION_LAUNCH_PATH_NOT_FOUND",
    "Message": "Failed to find a valid path",
    "EventTime": 1569319075.839,
    "PreSignedLogUrl": "https://gamelift-event-logs-prod-
us-west-2.s3.us-west-2.amazonaws.com/logs/fleet-83422059-8329-42a2-
a4d6-c4444386a6f8/events/2ecd0130-5986-44eb-99a7-62df27741084/
FLEET_VALIDATION_LAUNCH_PATH_NOT_FOUND.txt?X-Amz-Security-
Token=IQoJb3JpZ2luX2VjEB8aCXVzLXdlc3QtMiJHMEUCIHV5K%2FLPx8h310D
%2FAvx0%2FZxsDy5XA3cJ0wPdu3T0eBa%2FAiEA1yovokcZYy%2FV4CWW6126aFyiSH0
%2Bxz%2FBMAhEHYHMqNcqkQMImP%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F
%2FARAAGgw3NDEwNjE1OTIxNzEiDI8rsZtzLzlwEDQhXSrlAt15Ae
%2Fgo6FCIzqXPbXfB0nSvFyqeDlriZarEpKqKrUt8mXQv9iqHResqCph9AKo49lwgSYTT2QoSxnrD7%2FUgV

```

```

%2BZm2pVuczvuKtUA0fcx6s0GxpjIAzdIE%2F5P%2FB7B9M%2BVZ
%2F9KF82hbJi0HTE6Y7BjKsEgFCvk4UXILhfjtan9iQl8%2F21ZTurAcJbm7Y5tuLF9SWSK3%2BEa7VX0cCK4D401sMj
%2FIaXoHkNvg0RVTa0hIqdvpaDQlSsBNdqTXbjHTu6fETE9Y9Ky%2BiJK5KiUG
%2F59GjCpDcvS1FqKeLUEmKT7wysGmvjMc2n%2Fr
%2F9VxQfte7w9srXw1LAQuwhiXAAyI5ICMZ5JvzjzQwTqD4CHTVKUUDwL
%2BRZzbuuqkJ0bZm102CkRGp%2B74RTAzLbWptVqZTIIfzctiCTmWxb
%2FmKyELRYsVLrwNJ%2BGJ7%2BCrN0RC%2FjlgfLYIZyeAqjPgAu5HjgX
%2BM7jCo9M7wBTrnAXK0FQuf9dvA84SuwX0JFp17LYGjrHMKv0qC3GfbTMrZ6kzeNV9awKCpXB2Gnx9z2KvIlJdqirWV
%2F9C6%2B4jIZPME3jXmZcEHqqw5uvAVF7aeIavtUZU8pxpDIWT0YE4p3Kriy2AA7ziCRktVfjV839InyLk8LUjsioWK
%2BYUq8%2FDT1Lxqj1S%2Fi04TI0Wo7ilAo%2FKKWWF4guuNDexj8E00ynSp1yImB
%2BZf2Fua3044W4eEXAMPLE33333&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20170621T231808Z&X-Amz-SignedHeaders=host&X-Amz-Expires=900&X-Amz-
Credential=AKIAIOSFODNN7EXAMPLE%2F20170621%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY"
    }
  ],
  "NextToken":
    "eyJhd3NBZjY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjpw7InMiOiJidWlsZC01NWYxZTZmMS"
}

```

Para obtener más información, consulta [Debug GameLift Fleet Issues](#) en la Guía para GameLift desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeFleetEvents](#) la Referencia de AWS CLI comandos.

## describe-fleet-port-settings

En el siguiente ejemplo de código, se muestra cómo usar `describe-fleet-port-settings`.

### AWS CLI

Para ver los permisos de conexión entrante de una flota

En el siguiente `describe-fleet-port-settings` ejemplo, se recupera la configuración de conexión de una flota específica.

```

aws gamelift describe-fleet-port-settings \
  --fleet-id arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

```

Salida:

```
{
  "InboundPermissions": [
    {
      "FromPort": 33400,
      "ToPort": 33500,
      "IpRange": "0.0.0.0/0",
      "Protocol": "UDP"
    },
    {
      "FromPort": 1900,
      "ToPort": 2000,
      "IpRange": "0.0.0.0/0",
      "Protocol": "TCP"
    }
  ]
}
```

Para obtener más información, consulte [Configuración de GameLift flotas](#) en la Guía para GameLift desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeFleetPortSettings](#) la Referencia de AWS CLI comandos.

## describe-fleet-utilization

En el siguiente ejemplo de código, se muestra cómo usar `describe-fleet-utilization`.

### AWS CLI

Ejemplo 1: Para ver los datos de uso de una lista de flotas

El siguiente `describe-fleet-utilization` ejemplo recupera la información de uso actual de una flota específica.

```
aws gamelift describe-fleet-utilization \
  --fleet-ids arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{
```

```

    "FleetUtilization": [
      {
        "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "ActiveServerProcessCount": 100,
        "ActiveGameSessionCount": 62,
        "CurrentPlayerSessionCount": 329,
        "MaximumPlayerSessionCount": 1000
      }
    ]
  }

```

## Ejemplo 2: Para solicitar datos de uso de todas las flotas

A continuación se muestran `describe-fleet-utilization` los datos de uso de la flota de todas las flotas con cualquier estado. En este ejemplo, se utilizan parámetros de paginación para devolver datos de dos flotas a la vez.

```

aws gamelift describe-fleet-utilization \
  --limit 2

```

Salida:

```

{
  "FleetUtilization": [
    {
      "FleetId": "fleet-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
      "ActiveServerProcessCount": 100,
      "ActiveGameSessionCount": 13,
      "CurrentPlayerSessionCount": 98,
      "MaximumPlayerSessionCount": 1000
    },
    {
      "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
      "ActiveServerProcessCount": 100,
      "ActiveGameSessionCount": 62,
      "CurrentPlayerSessionCount": 329,
      "MaximumPlayerSessionCount": 1000
    }
  ],
  "NextToken":
  "eyJhd3NBZjY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS"
}

```

Llame al comando por segunda vez y pase el NextToken valor como argumento al `--next-token` parámetro para ver los dos resultados siguientes.

```
aws gamelift describe-fleet-utilization \  
  --limit 2 \  
  --next-  
token eyJhd3NBY2NvdW50SWQi0nsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZT
```

Repita el procedimiento hasta que la respuesta deje de incluir un NextToken valor en la salida.

Para obtener más información, consulta [GameLift Metrics for Fleets](#) en la Guía para GameLift desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeFleetUtilization](#) la Referencia de AWS CLI comandos.

## describe-game-session-queues

En el siguiente ejemplo de código, se muestra cómo usar `describe-game-session-queues`.

### AWS CLI

Para ver las colas de las sesiones del juego

En el siguiente `describe-game-session-queues` ejemplo, se recuperan las propiedades de dos colas especificadas.

```
aws gamelift describe-game-session-queues \  
  --names MegaFrogRace-NA MegaFrogRace-EU
```

Salida:

```
{  
  "GameSessionQueues": [{  
    "Destinations": [{  
      "DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
    },  
    {  
      "DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-  
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"  
    }  
  ]  
}
```



```

    ],
    "Name": "MegaFrogRace-NA",
    "TimeoutInSeconds": 600,
    "GameSessionQueueArn": "arn:aws:gamelift:us-west-2::gamesessionqueue/
MegaFrogRace-NA",
    "PlayerLatencyPolicies": [{
        "MaximumIndividualPlayerLatencyMilliseconds": 200
    },
    {
        "MaximumIndividualPlayerLatencyMilliseconds": 100,
        "PolicyDurationSeconds": 60
    }
    ],
    "FilterConfiguration": {
        "AllowedLocations": ["us-west-2", "ap-south-1", "us-east-1"]
    },
    "PriorityConfiguration": {
        "PriorityOrder": ["LOCATION", "FLEET_TYPE", "DESTINATION"],
        "LocationOrder": ["us-west-2", "ap-south-1", "us-east-1"]
    }
},
{
    "Destinations": [{
        "DestinationArn": "arn:aws:gamelift:eu-west-3::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
    }],
    "Name": "MegaFrogRace-EU",
    "TimeoutInSeconds": 600,
    "GameSessionQueueArn": "arn:aws:gamelift:us-west-2::gamesessionqueue/
MegaFrogRace-EU"
    }
]
}

```

Para obtener más información, consulte [Uso de colas multirregionales](#) en la Guía para GameLift desarrolladores de Amazon.

- Para API obtener más información, consulte la Referencia [DescribeGameSessionQueues](#) de AWS CLI comandos.

## describe-runtime-configuration

En el siguiente ejemplo de código, se muestra cómo usar `describe-runtime-configuration`.

## AWS CLI

Para solicitar la configuración de tiempo de ejecución de una flota

El siguiente `describe-runtime-configuration` ejemplo recupera detalles sobre la configuración de tiempo de ejecución actual de una flota específica.

```
aws gamelift describe-runtime-configuration \  
--fleet-id fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "RuntimeConfiguration": {  
    "ServerProcesses": [  
      {  
        "LaunchPath": "C:\\game\\Bin64.Release.Dedicated  
\\MegaFrogRace_Server.exe",  
        "Parameters": "+gamelift_start_server",  
        "ConcurrentExecutions": 3  
      },  
      {  
        "LaunchPath": "C:\\game\\Bin64.Release.Dedicated  
\\MegaFrogRace_Server.exe",  
        "Parameters": "+gamelift_start_server +debug",  
        "ConcurrentExecutions": 1  
      }  
    ],  
    "MaxConcurrentGameSessionActivations": 2147483647,  
    "GameSessionActivationTimeoutSeconds": 300  
  }  
}
```

Para obtener más información, consulte [Ejecutar varios procesos en una flota](#) en la Guía para GameLift desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeRuntimeConfiguration](#) la Referencia de AWS CLI comandos.

## list-builds

En el siguiente ejemplo de código, se muestra cómo usar `list-builds`.

## AWS CLI

Ejemplo 1: Para obtener una lista de versiones de juegos personalizadas

En el siguiente `list-builds` ejemplo, se recuperan las propiedades de todas las versiones de servidores de juegos de la región actual. La solicitud de muestra muestra cómo utilizar los parámetros de paginación `Limit` y `NextToken` cómo recuperar los resultados en conjuntos secuenciales. El primer comando recupera las dos primeras compilaciones. Como hay más de dos disponibles, la respuesta incluye una `NextToken` para indicar que hay más resultados disponibles.

```
aws gamelift list-builds \  
  --limit 2
```

Salida:

```
{  
  "Builds": [  
    {  
      "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "CreationTime": 1495664528.723,  
      "Name": "My_Game_Server_Build_One",  
      "OperatingSystem": "WINDOWS_2012",  
      "SizeOnDisk": 8567781,  
      "Status": "READY",  
      "Version": "12345.678"  
    },  
    {  
      "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "CreationTime": 1495528748.555,  
      "Name": "My_Game_Server_Build_Two",  
      "OperatingSystem": "AMAZON_LINUX_2",  
      "SizeOnDisk": 8567781,  
      "Status": "FAILED",  
      "Version": "23456.789"  
    }  
  ],  
  "NextToken": "EXAMPLE_NEXT_TOKEN"  
}
```

```

    "NextToken":
    "eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS
  }

```

A continuación, puede volver a llamar al comando con el `--next-token` parámetro siguiente para ver las dos compilaciones siguientes.

```

aws gamelift list-builds \
  --limit 2
  --next-
token eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS

```

Repita el procedimiento hasta que la respuesta no incluya ningún `NextToken` valor.

Ejemplo 2: Para obtener una lista de compilaciones de juegos personalizadas en estado de error

En el siguiente `list-builds` ejemplo, se recuperan las propiedades de todas las versiones de servidores de juegos de la región actual que tienen el estado actual. `FAILED`

```

aws gamelift list-builds \
  --status FAILED

```

Salida:

```

{
  "Builds": [
    {
      "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "CreationTime": 1495528748.555,
      "Name": "My_Game_Server_Build_Two",
      "OperatingSystem": "AMAZON_LINUX_2",
      "SizeOnDisk": 8567781,
      "Status": "FAILED",
      "Version": "23456.789"
    }
  ]
}

```

- Para API obtener más información, consulta [ListBuilds](#) la Referencia de AWS CLI comandos.

## list-fleets

En el siguiente ejemplo de código, se muestra cómo usar `list-fleets`.

### AWS CLI

Ejemplo 1: Para obtener una lista de todas las flotas de una región

En el siguiente `list-fleets` ejemplo, se muestra la flota IDs de todas las flotas de la región actual. En este ejemplo, se utilizan parámetros de paginación para recuperar dos flotas IDs a la vez. La respuesta incluye un `next-token` atributo que indica que hay más resultados que recuperar.

```
aws gamelift list-fleets \  
  --limit 2
```

Salida:

```
{  
  "FleetIds": [  
    "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"  
  ],  
  "NextToken":  
    "eyJhd3NBWY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMmM"  
}
```

Puede pasar el `NextToken` valor de la respuesta anterior al siguiente comando, como se muestra aquí, para obtener los dos resultados siguientes.

```
aws gamelift list-fleets \  
  --limit 2 \  
  --next-  
token eyJhd3NBWY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC00NDRLZj
```

Ejemplo 2: Para obtener una lista de todas las flotas de una región con una compilación o un script específicos

En el siguiente `list-builds` ejemplo, se recuperan IDs las flotas que se han desplegado con la versión de juego especificada. Si trabaja con Realtime Servers, puede proporcionar un ID de

script en lugar de un ID de compilación. Como en este ejemplo no se especifica el parámetro de límite, los resultados pueden incluir hasta 16 flotasIDs.

```
aws gamelift list-fleets \  
  --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "FleetIds": [  
    "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
    "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",  
    "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE44444"  
  ]  
}
```

- Para API obtener más información, consulte [ListFleets](#) la Referencia de AWS CLI comandos.

## request-upload-credentials

En el siguiente ejemplo de código, se muestra cómo usar `request-upload-credentials`.

AWS CLI

Para actualizar las credenciales de acceso para cargar una compilación

El siguiente `create-build` ejemplo obtiene credenciales de acceso nuevas y válidas para cargar un archivo de GameLift compilación en una ubicación de Amazon S3. Las credenciales tienen una vida útil limitada. El ID de compilación se obtiene de la respuesta a la `CreateBuild` solicitud original.

```
aws gamelift request-upload-credentials \  
  --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "StorageLocation": {  
    "Bucket": "gamelift-builds-us-west-2",
```

```
    "Key": "123456789012/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  },
  "UploadCredentials": {
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY",
    "SessionToken": "AgoGb3JpZ2luENz...EXAMPLETOKEN=="
  }
}
```

Para obtener más información, consulta [Cómo subir una versión de servidor personalizada GameLift en la Guía para GameLift desarrolladores](#) de Amazon.

- Para API obtener más información, consulte [RequestUploadCredentials](#) la Referencia de AWS CLI comandos.

## start-fleet-actions

En el siguiente ejemplo de código, se muestra cómo usar `start-fleet-actions`.

### AWS CLI

Para reiniciar la actividad de escalado automático de la flota

En el siguiente `start-fleet-actions` ejemplo, se reanuda el uso de todas las políticas de escalado que se definieron para la flota especificada, pero que se interrumpieron al `stop-fleet-actions` llamar `` ``. Tras empezar, las políticas de escalado comienzan inmediatamente a realizar un seguimiento de sus métricas respectivas.

```
aws gamelift start-fleet-actions \
  --fleet-id fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --actions AUTO_SCALING
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [StartFleetActions](#) la Referencia de AWS CLI comandos.

## stop-fleet-actions

En el siguiente ejemplo de código, se muestra cómo usar `stop-fleet-actions`.

## AWS CLI

Para detener la actividad de escalado automático de una flota

El siguiente `stop-fleet-actions` ejemplo detiene el uso de todas las políticas de escalado definidas para la flota especificada. Una vez suspendidas las políticas, la capacidad de la flota permanece con el mismo número de instancias activas, a menos que se ajuste manualmente.

```
aws gamelift start-fleet-actions \  
  --fleet-id fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --actions AUTO_SCALING
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [StopFleetActions](#) la Referencia de AWS CLI comandos.

## update-build

En el siguiente ejemplo de código, se muestra cómo usar `update-build`.

## AWS CLI

Para actualizar una versión de juego personalizada

En el siguiente `update-build` ejemplo, se cambia el nombre y la información de versión asociada a un recurso de compilación específico. El objeto de compilación devuelto verifica que los cambios se hayan realizado correctamente.

```
aws gamelift update-build \  
  --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --name MegaFrogRaceServer.NA.east \  
  --build-version 12345.east
```

Salida:

```
{  
  "Build": {  
    "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
```



```

    "CreationTime": 1496708916.18,
    "Name": "MegaFrogRaceServer.NA.east",
    "OperatingSystem": "AMAZON_LINUX_2",
    "SizeOnDisk": 1304924,
    "Status": "READY",
    "Version": "12345.east"
  }
}

```

Para obtener más información, consulta [Actualizar tus archivos de compilación](#) en la Guía para GameLift desarrolladores de Amazon.

- Para API obtener más información, consulte [UpdateBuild](#) la Referencia de AWS CLI comandos.

## update-game-session-queue

En el siguiente ejemplo de código, se muestra cómo usar `update-game-session-queue`.

### AWS CLI

Para actualizar la configuración de la cola de una sesión de juego

En el siguiente `update-game-session-queue` ejemplo, se añade un nuevo destino y se actualizan las políticas de latencia de los jugadores para una cola de sesiones de juego existente.

```

aws gamelift update-game-session-queue \
  --name MegaFrogRace-NA \
  --destinations file://destinations.json \
  --player-latency-policies file://latency-policies.json

```

Contenidos de `destinations.json`:

```

{
  "Destinations": [
    {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-1a2b3c4d-5e6f-7a8b-9c0d-1e2f3a4b5c6d"},
    {"DestinationArn": "arn:aws:gamelift:us-east-1::fleet/fleet-5c6d3c4d-5e6f-7a8b-9c0d-1e2f3a4b5a2b"},
    {"DestinationArn": "arn:aws:gamelift:us-east-1::alias/alias-11aa22bb-3c4d-5e6f-000a-1111aaaa22bb"}
  ]
}

```

## Contenidos de latency-policies.json:

```
{
  "PlayerLatencyPolicies": [
    {"MaximumIndividualPlayerLatencyMilliseconds": 200},
    {"MaximumIndividualPlayerLatencyMilliseconds": 150, "PolicyDurationSeconds":
120},
    {"MaximumIndividualPlayerLatencyMilliseconds": 100, "PolicyDurationSeconds":
120}
  ]
}
```

## Salida:

```
{
  "GameSessionQueue": {
    "Destinations": [
      {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/
fleet-1a2b3c4d-5e6f-7a8b-9c0d-1e2f3a4b5c6d"},
      {"DestinationArn": "arn:aws:gamelift:us-east-1::fleet/
fleet-5c6d3c4d-5e6f-7a8b-9c0d-1e2f3a4b5a2b"},
      {"DestinationArn": "arn:aws:gamelift:us-east-1::alias/
alias-11aa22bb-3c4d-5e6f-000a-1111aaaa22bb"}
    ],
    "GameSessionQueueArn": "arn:aws:gamelift:us-
west-2:111122223333:gamesessionqueue/MegaFrogRace-NA",
    "Name": "MegaFrogRace-NA",
    "TimeoutInSeconds": 600,
    "PlayerLatencyPolicies": [
      {"MaximumIndividualPlayerLatencyMilliseconds": 200},
      {"MaximumIndividualPlayerLatencyMilliseconds": 150,
"PolicyDurationSeconds": 120},
      {"MaximumIndividualPlayerLatencyMilliseconds": 100,
"PolicyDurationSeconds": 120}
    ]
  }
}
```

Para obtener más información, consulte [Uso de colas multirregionales](#) en la Guía para GameLift desarrolladores de Amazon.

- Para API obtener más información, consulte la Referencia [UpdateGameSessionQueue](#) de AWS CLI comandos.

## upload-build

En el siguiente ejemplo de código, se muestra cómo usar `upload-build`.

### AWS CLI

Ejemplo 1: Para cargar una versión de servidor de juegos Linux

En el siguiente `upload-build` ejemplo, se cargan los archivos de compilación de un servidor de juegos Linux desde un directorio de archivos al GameLift servicio y se crea un recurso de compilación.

```
aws gamelift upload-build \  
  --name MegaFrogRaceServer.NA \  
  --build-version 2.0.1 \  
  --build-root ~/MegaFrogRace_Server/release-na \  
  --operating-system AMAZON_LINUX_2 \  
  --server-sdk-version 4.0.2
```

Salida:

```
Uploading ~/MegaFrogRace_Server/release-na: 16.0 KiB / 74.6 KiB (21.45%)  
Uploading ~/MegaFrogRace_Server/release-na: 32.0 KiB / 74.6 KiB (42.89%)  
Uploading ~/MegaFrogRace_Server/release-na: 48.0 KiB / 74.6 KiB (64.34%)  
Uploading ~/MegaFrogRace_Server/release-na: 64.0 KiB / 74.6 KiB (85.79%)  
Uploading ~/MegaFrogRace_Server/release-na: 74.6 KiB / 74.6 KiB (100.00%)  
Successfully uploaded ~/MegaFrogRace_Server/release-na to AWS GameLift  
Build ID: build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Ejemplo 2: Para cargar una compilación de un servidor de juegos de Windows

En el siguiente `upload-build` ejemplo, se cargan los archivos de compilación del servidor de juegos de Windows desde un directorio al GameLift servicio y se crea un registro de compilación.

```
aws gamelift upload-build \  
  --name MegaFrogRaceServer.NA \  
  --build-version 2.0.1 \  
  --build-root C:\MegaFrogRace_Server\release-na \  
  --operating-system WINDOWS_2012 \  
  --server-sdk-version 4.0.2
```

Salida:

```
Uploading C:\MegaFrogRace_Server\release-na: 16.0 KiB / 74.6 KiB (21.45%)
Uploading C:\MegaFrogRace_Server\release-na: 32.0 KiB / 74.6 KiB (42.89%)
Uploading C:\MegaFrogRace_Server\release-na: 48.0 KiB / 74.6 KiB (64.34%)
Uploading C:\MegaFrogRace_Server\release-na: 64.0 KiB / 74.6 KiB (85.79%)
Uploading C:\MegaFrogRace_Server\release-na: 74.6 KiB / 74.6 KiB (100.00%)
Successfully uploaded C:\MegaFrogRace_Server\release-na to AWS GameLift
Build ID: build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Para obtener más información, consulta [Cómo subir una versión de servidor personalizada GameLift en la Guía para GameLift desarrolladores](#) de Amazon.

- Para API obtener más información, consulte [UploadBuild](#) la Referencia de AWS CLI comandos.

## Ejemplos de Global Accelerator utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el AWS Command Line Interface uso de Global Accelerator.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **add-custom-routing-endpoints**

En el siguiente ejemplo de código, se muestra cómo usar `add-custom-routing-endpoints`.

### AWS CLI

Para agregar un punto final de VPC subred a un grupo de puntos finales para un acelerador de enrutamiento personalizado

En el siguiente `add-custom-routing-endpoints` ejemplo, se agrega un punto final de VPC subred a un grupo de puntos finales para un acelerador de enrutamiento personalizado.

```
aws globalaccelerator add-custom-routing-endpoints \
  --endpoint-group-
  arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
  abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/4321abcd \
  --endpoint-configurations "EndpointId=subnet-1234567890abcdef0"
```

Salida:

```
{
  "EndpointDescriptions": [
    {
      "EndpointId": "subnet-1234567890abcdef0"
    }
  ],
  "EndpointGroupArn": "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
  abcd-1234-abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/4321abcd"
}
```

Para obtener más información, consulte los [puntos finales de VPC subred para ver los aceleradores de enrutamiento personalizados en Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).AWS

- Para obtener API más información, consulte la Referencia de comandos. [AddCustomRoutingEndpoints](#)AWS CLI

## advertise-byoip-cidr

En el siguiente ejemplo de código, se muestra cómo usar `advertise-byoip-cidr`.

AWS CLI

Para anunciar un rango de direcciones

En el siguiente `advertise-byoip-cidr` ejemplo, AWS se solicita anunciar un rango de direcciones que has aprovisionado para usarlo con tus AWS recursos.

```
aws globalaccelerator advertise-byoip-cidr \
  --cidr 198.51.100.0/24
```

Salida:

```
{
  "ByoipCidr": {
    "Cidr": "198.51.100.0/24",
    "State": "PENDING_ADVERTISING"
  }
}
```

Para obtener más información, consulte Incluya [su propia dirección IP en AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia [AdvertiseByoipCidr](#) de AWS CLI comandos.

## allow-custom-routing-traffic

En el siguiente ejemplo de código, se muestra cómo usar `allow-custom-routing-traffic`.

### AWS CLI

Para permitir el tráfico a destinos específicos de EC2 instancias de Amazon en una VPC subred para un acelerador de enrutamiento personalizado

El siguiente `allow-custom-routing-traffic` ejemplo especifica que se permite el tráfico a determinadas direcciones IP (destino) de EC2 instancias de Amazon y que los puertos de un punto final de VPC subred en un acelerador de enrutamiento personalizado pueden recibir tráfico.

```
aws globalaccelerator allow-custom-routing-traffic \
  --endpoint-group-
  arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
  abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/ab8888example \
  --endpoint-id subnet-abcd123example \
  --destination-addresses "172.31.200.6" "172.31.200.7" \
  --destination-ports 80 81
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [puntos de enlace de VPC subred para ver los aceleradores de enrutamiento personalizados en Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).AWS

- Para obtener API más información, consulte la Referencia de comandos. [AllowCustomRoutingTraffic](#) AWS CLI

## create-accelerator

En el siguiente ejemplo de código, se muestra cómo usar `create-accelerator`.

### AWS CLI

Para crear un acelerador

El siguiente `create-accelerator` ejemplo crea un acelerador con dos etiquetas con dos direcciones IP BYOIP estáticas. Debe especificar la US-West-2 (Oregon) región para crear o actualizar un acelerador.

```
aws globalaccelerator create-accelerator \  
  --name ExampleAccelerator \  
  --tags Key="Name",Value="Example Name" Key="Project",Value="Example Project" \  
  --ip-addresses 192.0.2.250 198.51.100.52
```

Salida:

```
{  
  "Accelerator": {  
    "AcceleratorArn":  
    "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-  
abcd-1234abcdefgh",  
    "IpAddressType": "IPv4",  
    "Name": "ExampleAccelerator",  
    "Enabled": true,  
    "Status": "IN_PROGRESS",  
    "IpSets": [  
      {  
        "IpAddresses": [  
          "192.0.2.250",  
          "198.51.100.52"  
        ],  
        "IpFamily": "IPv4"  
      }  
    ],  
    "DnsName": "a1234567890abcdef.awsglobalaccelerator.com",  
    "CreatedTime": 1542394847.0,  
    "LastModifiedTime": 1542394847.0  
  }  
}
```

Para obtener más información, consulte [Accelerators in AWS Global Accelerator en](#) la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia de [CreateAccelerator](#) comandos AWS CLI .

## **create-custom-routing-accelerator**

En el siguiente ejemplo de código, se muestra cómo usar `create-custom-routing-accelerator`.

### AWS CLI

Para crear un acelerador de enrutamiento personalizado

El siguiente `create-custom-routing-accelerator` ejemplo crea un acelerador de enrutamiento personalizado con las etiquetas `Name` y `Project`

```
aws globalaccelerator create-custom-routing-accelerator \
  --name ExampleCustomRoutingAccelerator \
  --tags Key="Name",Value="Example Name" Key="Project",Value="Example Project" \
  --ip-addresses 192.0.2.250 198.51.100.52
```

Salida:

```
{
  "Accelerator": {
    "AcceleratorArn":
      "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
      abcd-1234abcdefg",
    "IpAddressType": "IPV4",
    "Name": "ExampleCustomRoutingAccelerator",
    "Enabled": true,
    "Status": "IN_PROGRESS",
    "IpSets": [
      {
        "IpAddresses": [
          "192.0.2.250",
          "198.51.100.52"
        ],
        "IpFamily": "IPv4"
      }
    ]
  }
}
```



```

    ],
    "DnsName": "a1234567890abcdef.awsglobalaccelerator.com",
    "CreatedTime": 1542394847.0,
    "LastModifiedTime": 1542394847.0
  }
}

```

Para obtener más información, consulte los [aceleradores de enrutamiento personalizados en AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia de [CreateCustomRoutingAccelerator](#) comandos AWS CLI .

## create-custom-routing-endpoint-group

En el siguiente ejemplo de código, se muestra cómo usar `create-custom-routing-endpoint-group`.

### AWS CLI

Para crear un grupo de puntos finales para un acelerador de enrutamiento personalizado

El siguiente `create-custom-routing-endpoint-group` ejemplo crea un grupo de puntos finales para un acelerador de enrutamiento personalizado.

```

aws globalaccelerator create-custom-routing-endpoint-group \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh/listener/0123vxyz \
  --endpoint-group-region us-east-2 \
  --destination-configurations "FromPort=80,ToPort=81,Protocols=TCP,UDP"

```

Salida:

```

{
  "EndpointGroup": {
    "EndpointGroupArn":
      "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-  
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/4321abcd",
    "EndpointGroupRegion": "us-east-2",
    "DestinationDescriptions": [
      {

```

```

        "FromPort": 80,
        "ToPort": 81,
        "Protocols": [
            "TCP",
            "UDP"
        ]
    },
    ],
    "EndpointDescriptions": []
}
}

```

Para obtener más información, consulte [los grupos de puntos finales para obtener aceleradores de enrutamiento personalizados en AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia de [CreateCustomRoutingEndpointGroup](#) comandos AWS CLI .

## create-custom-routing-listener

En el siguiente ejemplo de código, se muestra cómo usar `create-custom-routing-listener`.

### AWS CLI

Para crear un detector para un acelerador de enrutamiento personalizado

El siguiente `create-custom-routing-listener` ejemplo crea un listener con un rango de puertos de 5000 a 10000 para un acelerador de enrutamiento personalizado.

```

aws globalaccelerator create-custom-routing-listener \
  --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh \
  --port-ranges FromPort=5000, ToPort=10000

```

Salida:

```

{
  "Listener": {
    "PortRange": [
      "FromPort": 5000,
      "ToPort": 10000
    ]
  }
}

```

```

    ],
    "ListenerArn":
    "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz"
  }
}

```

Para obtener más información, consulte [Listeners para obtener aceleradores de enrutamiento personalizados en Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).AWS

- Para obtener API más información, consulte la Referencia de comandos [CreateCustomRoutingListener](#).AWS CLI

## create-endpoint-group

En el siguiente ejemplo de código, se muestra cómo usar create-endpoint-group.

AWS CLI

Para crear un grupo de puntos finales

En el siguiente create-endpoint-group ejemplo, se crea un grupo de puntos finales con un punto final.

```

aws globalaccelerator create-endpoint-group \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/0123vxyz \
  --endpoint-group-region us-east-1 \
  --endpoint-configurations EndpointId=i-1234567890abcdef0,Weight=128

```

Salida:

```

{
  "EndpointGroup": {
    "TrafficDialPercentage": 100.0,
    "EndpointDescriptions": [
      {
        "Weight": 128,
        "EndpointId": "i-1234567890abcdef0"
      }
    ]
  },
}

```

```

    "EndpointGroupArn":
      "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/098765zyxwvu",
    "EndpointGroupRegion": "us-east-1"
  }
}

```

Para obtener más información, consulte [los grupos de puntos finales de AWS Global Accelerator](#) en la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia [CreateEndpointGroup](#) de AWS CLI comandos.

## create-listener

En el siguiente ejemplo de código, se muestra cómo usar `create-listener`.

### AWS CLI

Para crear un oyente

En el siguiente `create-listener` ejemplo, se crea un oyente con dos puertos.

```

aws globalaccelerator create-listener \
  --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh \
  --port-ranges FromPort=80,ToPort=80 FromPort=81,ToPort=81 \
  --protocol TCP

```

Salida:

```

{
  "Listener": {
    "PortRanges": [
      {
        "ToPort": 80,
        "FromPort": 80
      },
      {
        "ToPort": 81,
        "FromPort": 81
      }
    ],
  },
}

```

```

    "ClientAffinity": "NONE",
    "Protocol": "TCP",
    "ListenerArn":
      "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
      abcd-1234abcdefgh/listener/0123vxyz"
  }
}

```

Para obtener más información, consulte [Listeners in AWS Global Accelerator](#) en la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia de [CreateListener](#) comandos AWS CLI .

## deny-custom-routing-traffic

En el siguiente ejemplo de código, se muestra cómo usar `deny-custom-routing-traffic`.

### AWS CLI

Para especificar una dirección de destino que no pueda recibir tráfico en un acelerador de enrutamiento personalizado

El siguiente `deny-custom-routing-traffic` ejemplo especifica la dirección o direcciones de destino de un punto final de subred que no pueden recibir tráfico para un acelerador de enrutamiento personalizado. Para especificar más de una dirección de destino, separe las direcciones con un espacio. No hay respuesta si la `deny-custom-routing-traffic` llamada se ha realizado correctamente.

```

aws globalaccelerator deny-custom-routing-traffic \
  --endpoint-group-
  arn "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
  abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/ab8888example" \
  --endpoint-id "subnet-abcd123example" \
  --destination-addresses "198.51.100.52"

```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [puntos finales de VPC subred para ver los aceleradores de enrutamiento personalizados en Global Accelerator](#) en la Guía para desarrolladores de AWSAWS Global Accelerator.

- Para obtener API más información, consulte la Referencia de comandos.  
[DenyCustomRoutingTraffic](#) AWS CLI

## deprovision-byoip-cidr

En el siguiente ejemplo de código, se muestra cómo usar `deprovision-byoip-cidr`.

### AWS CLI

Para desaproveccionar un rango de direcciones

En el siguiente `deprovision-byoip-cidr` ejemplo, se publica el rango de direcciones especificado que ha aprovisionado para usarlo con sus AWS recursos.

```
aws globalaccelerator deprovision-byoip-cidr \  
  --cidr "198.51.100.0/24"
```

Salida:

```
{  
  "ByoipCidr": {  
    "Cidr": "198.51.100.0/24",  
    "State": "PENDING_DEPROVISIONING"  
  }  
}
```

Para obtener más información, consulte Incluya [su propia dirección IP en AWS Global Accelerator](#) en la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia [DeprovisionByoipCidr](#) de AWS CLI comandos.

## describe-accelerator-attributes

En el siguiente ejemplo de código, se muestra cómo usar `describe-accelerator-attributes`.

### AWS CLI

Para describir los atributos de un acelerador

En el siguiente `describe-accelerator-attributes` ejemplo, se recuperan los detalles de los atributos de un acelerador.

```
aws globalaccelerator describe-accelerator-attributes \  
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh
```

Salida:

```
{  
  "AcceleratorAttributes": {  
    "FlowLogsEnabled": true  
    "FlowLogsS3Bucket": flowlogs-abc  
    "FlowLogsS3Prefix": bucketprefix-abc  
  }  
}
```

Para obtener más información, consulte [Accelerators in AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia de [DescribeAcceleratorAttributes](#) comandos AWS CLI .

## describe-accelerator

En el siguiente ejemplo de código, se muestra cómo usar describe-accelerator.

### AWS CLI

Para describir un acelerador

En el siguiente describe-accelerator ejemplo, se recuperan los detalles sobre el acelerador especificado.

```
aws globalaccelerator describe-accelerator \  
  --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh
```

Salida:

```
{  
  "Accelerator": {  
    "AcceleratorArn":  
    "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-  
abcd-1234abcdefgh",
```

```

    "IpAddressType": "IPV4",
    "Name": "ExampleAccelerator",
    "Enabled": true,
    "Status": "IN_PROGRESS",
    "IpSets": [
      {
        "IpAddresses": [
          "192.0.2.250",
          "198.51.100.52"
        ],
        "IpFamily": "IPv4"
      }
    ],
    "DnsName": "a1234567890abcdef.awsglobalaccelerator.com",
    "CreatedTime": 1542394847,
    "LastModifiedTime": 1542395013
  }
}

```

Para obtener más información, consulte [Accelerators in AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia de [DescribeAccelerator](#) comandos AWS CLI .

## describe-custom-routing-accelerator-attributes

En el siguiente ejemplo de código, se muestra cómo usar `describe-custom-routing-accelerator-attributes`.

AWS CLI

Para describir los atributos de un acelerador de enrutamiento personalizado

El siguiente `describe-custom-routing-accelerator-attributes` ejemplo describe los atributos de un acelerador de enrutamiento personalizado.

```

aws globalaccelerator describe-custom-routing-accelerator-attributes \
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh

```

Salida:



```
{
  "AcceleratorAttributes": {
    "FlowLogsEnabled": false
  }
}
```

Para obtener más información, consulte los [aceleradores de enrutamiento personalizados en AWS Global Accelerator en](#) la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia de [DescribeCustomRoutingAcceleratorAttributes](#) comandos AWS CLI .

## describe-custom-routing-accelerator

En el siguiente ejemplo de código, se muestra cómo usar `describe-custom-routing-accelerator`.

### AWS CLI

Para describir un acelerador de enrutamiento personalizado

El siguiente `describe-custom-routing-accelerator` ejemplo recupera los detalles sobre el acelerador de enrutamiento personalizado especificado.

```
aws globalaccelerator describe-custom-routing-accelerator \
  --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh
```

Salida:

```
{
  "Accelerator": {
    "AcceleratorArn":
      "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh",
    "IpAddressType": "IPV4",
    "Name": "ExampleCustomRoutingAccelerator",
    "Enabled": true,
    "Status": "IN_PROGRESS",
    "IpSets": [
      {
        "IpAddresses": [
```

```

        "192.0.2.250",
        "198.51.100.52"
    ],
    "IpFamily": "IPv4"
}
],
"DnsName": "a1234567890abcdef.awsglobalaccelerator.com",
"CreatedTime": 1542394847,
"LastModifiedTime": 1542395013
}
}

```

Para obtener más información, consulte los [aceleradores de enrutamiento personalizados en AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia de [DescribeCustomRoutingAccelerator](#) comandos AWS CLI .

## describe-custom-routing-endpoint-group

En el siguiente ejemplo de código, se muestra cómo usar `describe-custom-routing-endpoint-group`.

### AWS CLI

Para describir un grupo de puntos finales para un acelerador de enrutamiento personalizado

El siguiente `describe-custom-routing-endpoint-group` ejemplo describe un grupo de puntos finales para un acelerador de enrutamiento personalizado.

```

aws globalaccelerator describe-custom-routing-endpoint-group \
  --endpoint-group-
  arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
  abcd-1234abcdefgh/listener/6789vxyz/endpoint-group/ab8888example

```

Salida:

```

{
  "EndpointGroup": {
    "EndpointGroupArn":
    "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
    abcd-1234abcdefgh/listener/6789vxyz/endpoint-group/ab8888example",

```

```

    "EndpointGroupRegion": "us-east-2",
    "DestinationDescriptions": [
      {
        "FromPort": 5000,
        "ToPort": 10000,
        "Protocols": [
          "UDP"
        ]
      }
    ],
    "EndpointDescriptions": [
      {
        "EndpointId": "subnet-1234567890abcdef0"
      }
    ]
  }
}

```

Para obtener más información, consulte [los grupos de puntos finales para obtener aceleradores de enrutamiento personalizados en AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia de [DescribeCustomRoutingEndpointGroup](#) comandos AWS CLI .

## describe-custom-routing-listener

En el siguiente ejemplo de código, se muestra cómo usar `describe-custom-routing-listener`.

### AWS CLI

Para describir el agente de escucha de un acelerador de enrutamiento personalizado

El siguiente `describe-custom-routing-listener` ejemplo describe un agente de escucha para un acelerador de enrutamiento personalizado.

```

aws globalaccelerator describe-custom-routing-listener \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/abcdef1234

```

Salida:

```
{
  "Listener": {
    "PortRanges": [
      "FromPort": 5000,
      "ToPort": 10000
    ],
    "ListenerArn":
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234"
  }
}
```

Para obtener más información, consulte [Listeners para obtener aceleradores de enrutamiento personalizados en Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).AWS

- Para obtener API más información, consulte la Referencia de comandos [DescribeCustomRoutingListener](#).AWS CLI

## describe-endpoint-group

En el siguiente ejemplo de código, se muestra cómo usar `describe-endpoint-group`.

### AWS CLI

Para describir un grupo de puntos finales

El siguiente `describe-endpoint-group` ejemplo recupera detalles sobre un grupo de puntos de enlace con los siguientes puntos de enlace: una EC2 instancia de AmazonALB, un y un. NLB

```
aws globalaccelerator describe-endpoint-group \
  --endpoint-group-
arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/6789vxyz-vxyz-6789-vxyz-6789lmnopqrs/endpoint-group/
ab8888example
```

Salida:

```
{
  "EndpointGroup": {
    "TrafficDialPercentage": 100.0,
    "EndpointDescriptions": [
```

```

    {
      "Weight": 128,
      "EndpointId": "i-1234567890abcdef0"
    },
    {
      "Weight": 128,
      "EndpointId": "arn:aws:elasticloadbalancing:us-
east-1:000123456789:loadbalancer/app/ALBTesting/alb01234567890xyz"
    },
    {
      "Weight": 128,
      "EndpointId": "arn:aws:elasticloadbalancing:us-
east-1:000123456789:loadbalancer/net/NLBTesting/alb01234567890qrs"
    }
  ],
  "EndpointGroupArn":
  "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/6789vxyz-vxyz-6789-vxyz-6789lmnopqrs/endpoint-
group/4321abcd-abcd-4321-abcd-4321abcdefg",
  "EndpointGroupRegion": "us-east-1"
}
}

```

Para obtener más información, consulte [los grupos de puntos finales de AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia [DescribeEndpointGroup](#) de AWS CLI comandos.

## describe-listener

En el siguiente ejemplo de código, se muestra cómo usar `describe-listener`.

### AWS CLI

Para describir a un oyente

El siguiente `describe-listener` ejemplo describe un oyente.

```

aws globalaccelerator describe-listener \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/abcdef1234

```

Salida:

```
{
  "Listener": {
    "ListenerArn":
    "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234",
    "PortRanges": [
      {
        "FromPort": 80,
        "ToPort": 80
      }
    ],
    "Protocol": "TCP",
    "ClientAffinity": "NONE"
  }
}
```

Para obtener más información, consulte [Listeners in AWS Global Accelerator](#) en la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia de [DescribeListener](#) comandos AWS CLI .

## list-accelerators

En el siguiente ejemplo de código, se muestra cómo usar `list-accelerators`.

AWS CLI

Para ver una lista de sus aceleradores

En el siguiente `list-accelerators` ejemplo, se muestran los aceleradores de tu cuenta. AWS Esta cuenta tiene dos aceleradores.

```
aws globalaccelerator list-accelerators
```

Salida:

```
{
  "Accelerators": [
    {
```



Para obtener más información, consulte [los aceleradores de AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia de [ListAccelerators](#) comandos AWS CLI .

## **list-byoip-cidr**

En el siguiente ejemplo de código, se muestra cómo usar `list-byoip-cidr`.

### AWS CLI

Para enumerar sus rangos de direcciones

En el siguiente `list-byoip-cidr` ejemplo, se enumeran los rangos de direcciones «traiga su propia dirección IP» (BYOIP) que ha aprovisionado para su uso con Global Accelerator.

```
aws globalaccelerator list-byoip-cidrs
```

Salida:

```
{
  "ByoipCidrs": [
    {
      "Cidr": "198.51.100.0/24",
      "State": "READY"
    }
    {
      "Cidr": "203.0.113.25/24",
      "State": "READY"
    }
  ]
}
```

Para obtener más información, consulte Incluya [su propia dirección IP en AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia [ListByoipCidr](#) de AWS CLI comandos.







Salida:

```
{
  "EndpointGroups": [
    {
      "EndpointGroupArn":
      "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234/endpoint-group/ab88888example",
      "EndpointGroupRegion": "eu-central-1",
      "DestinationDescriptions": [
        {
          "FromPort": 80,
          "ToPort": 80,
          "Protocols": [
            "TCP",
            "UDP"
          ]
        }
      ]
      "EndpointDescriptions": [
        {
          "EndpointId": "subnet-abcd123example"
        }
      ]
    }
  ]
}
```

Para obtener más información, consulte [los grupos de puntos finales para ver los aceleradores de enrutamiento personalizados en AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia de [ListCustomRoutingEndpointGroups](#) comandos AWS CLI .

## **list-custom-routing-listeners**

En el siguiente ejemplo de código, se muestra cómo usar `list-custom-routing-listeners`.

AWS CLI

Para enumerar los oyentes de los aceleradores de enrutamiento personalizados

En el siguiente `list-custom-routing-listeners` ejemplo, se enumeran los oyentes de un acelerador de enrutamiento personalizado.

```
aws globalaccelerator list-custom-routing-listeners \
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh
```

Salida:

```
{
  "Listeners": [
    {
      "ListenerArn":
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234",
      "PortRanges": [
        {
          "FromPort": 5000,
          "ToPort": 10000
        }
      ],
      "Protocol": "TCP"
    }
  ]
}
```

Para obtener más información, consulte [Listeners para obtener aceleradores de enrutamiento personalizados en Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).AWS

- Para obtener API más información, consulte la Referencia de comandos [ListCustomRoutingListeners](#).AWS CLI

## **list-custom-routing-port-mappings-by-destination**

En el siguiente ejemplo de código, se muestra cómo usar `list-custom-routing-port-mappings-by-destination`.

AWS CLI

Para enumerar las asignaciones de puertos para un destino específico de un acelerador de enrutamiento personalizado

El siguiente `list-custom-routing-port-mappings-by-destination` ejemplo proporciona las asignaciones de puertos para un EC2 servidor de destino específico (en la dirección de destino) para un acelerador de enrutamiento personalizado.

```
aws globalaccelerator list-custom-routing-port-mappings-by-destination \  
  --endpoint-id subnet-abcd123example \  
  --destination-address 198.51.100.52
```

Salida:

```
{  
  "DestinationPortMappings": [  
    {  
      "AcceleratorArn":  
        "arn:aws:globalaccelerator::402092451327:accelerator/24ea29b8-  
d750-4489-8919-3095f3c4b0a7",  
      "AcceleratorSocketAddresses": [  
        {  
          "IpAddress": "192.0.2.250",  
          "Port": 65514  
        },  
        {  
          "IpAddress": "192.10.100.99",  
          "Port": 65514  
        }  
      ],  
      "EndpointGroupArn":  
        "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-  
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/ab88888example",  
      "EndpointId": "subnet-abcd123example",  
      "EndpointGroupRegion": "us-west-2",  
      "DestinationSocketAddress": {  
        "IpAddress": "198.51.100.52",  
        "Port": 80  
      },  
      "IpAddressType": "IPv4",  
      "DestinationTrafficState": "ALLOW"  
    }  
  ]  
}
```

Para obtener más información, consulte [Cómo funcionan los aceleradores de enrutamiento personalizados en Global Accelerator en la AWS Guía para desarrolladores de Global Accelerator](#).AWS

- Para API obtener más información, consulte la Referencia de [ListCustomRoutingPortMappingsByDestination](#) comandos AWS CLI .

## list-custom-routing-port-mappings

En el siguiente ejemplo de código, se muestra cómo usar `list-custom-routing-port-mappings`.

### AWS CLI

Para enumerar las asignaciones de puertos en un acelerador de enrutamiento personalizado

El siguiente `list-custom-routing-port-mappings` ejemplo proporciona una lista parcial de las asignaciones de puertos en un acelerador de enrutamiento personalizado.

```
aws globalaccelerator list-custom-routing-port-mappings \
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
  abcd-1234-abcd-1234abcdefgh
```

Salida:

```
{
  "PortMappings": [
    {
      "AcceleratorPort": 40480,
      "EndpointGroupArn":
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/098765zyxwvu",
      "EndpointId": "subnet-1234567890abcdef0",
      "DestinationSocketAddress": {
        "IpAddress": "192.0.2.250",
        "Port": 80
      },
      "Protocols": [
        "TCP",
        "UDP"
      ],
      "DestinationTrafficState": "ALLOW"
    }
  ]
}
```

```

    }
    {
      "AcceleratorPort": 40481,
      "EndpointGroupArn":
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/098765zyxwvu",
      "EndpointId": "subnet-1234567890abcdef0",
      "DestinationSocketAddress": {
        "IpAddress": "192.0.2.251",
        "Port": 80
      },
      "Protocols": [
        "TCP",
        "UDP"
      ],
      "DestinationTrafficState": "ALLOW"
    }
  ]
}

```

Para obtener más información, consulte [Cómo funcionan los aceleradores de enrutamiento personalizados en Global Accelerator en la AWS Guía para desarrolladores de Global Accelerator](#).AWS

- Para API obtener más información, consulte la Referencia de [ListCustomRoutingPortMappings](#) comandos AWS CLI .

## list-endpoint-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-endpoint-groups`.

### AWS CLI

Para enumerar los grupos de puntos finales

En el siguiente `list-endpoint-groups` ejemplo, se enumeran los grupos de puntos finales de un listener. Este oyente tiene dos grupos de puntos finales.

```

aws globalaccelerator --region us-west-2 list-endpoint-groups \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/abcdef1234

```

Salida:

```
{
  "EndpointGroups": [
    {
      "EndpointGroupArn":
        "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
        abcd-1234abcdefgh/listener/abcdef1234/endpoint-group/ab88888example",
      "EndpointGroupRegion": "eu-central-1",
      "EndpointDescriptions": [],
      "TrafficDialPercentage": 100.0,
      "HealthCheckPort": 80,
      "HealthCheckProtocol": "TCP",
      "HealthCheckIntervalSeconds": 30,
      "ThresholdCount": 3
    }
    {
      "EndpointGroupArn":
        "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
        abcd-1234abcdefgh/listener/abcdef1234/endpoint-group/ab99999example",
      "EndpointGroupRegion": "us-east-1",
      "EndpointDescriptions": [],
      "TrafficDialPercentage": 50.0,
      "HealthCheckPort": 80,
      "HealthCheckProtocol": "TCP",
      "HealthCheckIntervalSeconds": 30,
      "ThresholdCount": 3
    }
  ]
}
```

Para obtener más información, consulte [los grupos de puntos finales de AWS Global Accelerator](#) en la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia [ListEndpointGroups](#) de AWS CLI comandos.

## list-listeners

En el siguiente ejemplo de código, se muestra cómo usar `list-listeners`.



## AWS CLI

Para enumerar los oyentes

En el siguiente `list-listeners` ejemplo, se enumeran los oyentes de un acelerador.

```
aws globalaccelerator list-listeners \
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh
```

Salida:

```
{
  "Listeners": [
    {
      "ListenerArn":
"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234",
      "PortRanges": [
        {
          "FromPort": 80,
          "ToPort": 80
        }
      ],
      "Protocol": "TCP",
      "ClientAffinity": "NONE"
    }
  ]
}
```

Para obtener más información, consulte [Listeners in AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia de [ListListeners](#) comandos AWS CLI .

## **list-tags-for-resource**

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

## AWS CLI

Para enumerar las etiquetas de un acelerador

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas de un acelerador específico.

```
aws globalaccelerator list-tags-for-resource \  
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh
```

Salida:

```
{  
  "Tags": [  
    {  
      "Key": "Project",  
      "Value": "A123456"  
    }  
  ]  
}
```

Para obtener más información, consulte [Etiquetar en AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia de [ListTagsForResource](#) comandos AWS CLI .

## provision-byoip-cidr

En el siguiente ejemplo de código, se muestra cómo usar `provision-byoip-cidr`.

AWS CLI

Para aprovisionar un rango de direcciones

En el siguiente `provision-byoip-cidr` ejemplo, se proporciona el rango de direcciones especificado para usarlo con AWS los recursos.

```
aws globalaccelerator provision-byoip-cidr \  
  --cidr 192.0.2.250/24 \  
  --cidr-authorization-context Message="$text_message",Signature="$signed_message"
```

Salida:

```
{
  "ByoipCidr": {
    "Cidr": "192.0.2.250/24",
    "State": "PENDING_PROVISIONING"
  }
}
```

Para obtener más información, consulte [Incluya su propia dirección IP en AWS Global Accelerator](#) en la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia [ProvisionByoipCidr](#) de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar un acelerador

En el siguiente `tag-resource` ejemplo, se añaden las etiquetas Nombre y Proyecto a un acelerador, junto con los valores correspondientes a cada una de ellas.

```
aws globalaccelerator tag-resource \
  --resource-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh \
  --tags Key="Name",Value="Example Name" Key="Project",Value="Example Project"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado en AWS Global Accelerator en](#) la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia de [TagResource](#) comandos AWS CLI .

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

## AWS CLI

Para eliminar una etiqueta de un acelerador

En el siguiente `untag-resource` ejemplo, se eliminan las etiquetas Nombre y Proyecto de un acelerador.

```
aws globalaccelerator untag-resource \  
  --resource-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh \  
  --tag-keys Key="Name" Key="Project"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar en AWS Global Accelerator en](#) la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia de [UntagResource](#) comandos AWS CLI .

## update-accelerator-attributes

En el siguiente ejemplo de código, se muestra cómo usar `update-accelerator-attributes`.

### AWS CLI

Para actualizar los atributos de un acelerador

En el siguiente `update-accelerator-attributes` ejemplo, se actualiza un acelerador para habilitar los registros de flujo. Debe especificar la US-West-2 (Oregon) región para crear o actualizar los atributos del acelerador.

```
aws globalaccelerator update-accelerator-attributes \  
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh \  
  --flow-logs-enabled \  
  --flow-logs-s3-bucket flowlogs-abc \  
  --flow-logs-s3-prefix bucketprefix-abc
```

Salida:

```
{
```

```
"AcceleratorAttributes": {
  "FlowLogsEnabled": true
  "FlowLogsS3Bucket": flowlogs-abc
  "FlowLogsS3Prefix": bucketprefix-abc
}
```

Para obtener más información, consulte [Accelerators in AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia de [UpdateAcceleratorAttributes](#) comandos AWS CLI .

## update-accelerator

En el siguiente ejemplo de código, se muestra cómo usar update-accelerator.

### AWS CLI

Para actualizar un acelerador

En el siguiente update-accelerator ejemplo, se modifica un acelerador para cambiarle el nombre a. ExampleAcceleratorNew Debe especificar la US-West-2 (Oregon) región para crear o actualizar los aceleradores.

```
aws globalaccelerator update-accelerator \
  --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh \
  --name ExampleAcceleratorNew
```

Salida:

```
{
  "Accelerator": {
    "AcceleratorArn":
"arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh",
    "IpAddressType": "IPV4",
    "Name": "ExampleAcceleratorNew",
    "Enabled": true,
    "Status": "IN_PROGRESS",
    "IpSets": [
```

```

    {
      "IpAddresses": [
        "192.0.2.250",
        "198.51.100.52"
      ],
      "IpFamily": "IPv4"
    }
  ],
  "DnsName": "a1234567890abcdef.awsglobalaccelerator.com",
  "CreatedTime": 1232394847,
  "LastModifiedTime": 1232395654
}
}

```

Para obtener más información, consulte [los aceleradores de AWS Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia de [UpdateAccelerator](#) comandos AWS CLI .

## update-custom-routing-accelerator-attributes

En el siguiente ejemplo de código, se muestra cómo usar `update-custom-routing-accelerator-attributes`.

### AWS CLI

Para actualizar los atributos de un acelerador de enrutamiento personalizado

El siguiente `update-custom-routing-accelerator-attributes` ejemplo actualiza un acelerador de enrutamiento personalizado para habilitar los registros de flujo.

```

aws globalaccelerator update-custom-routing-accelerator-attributes \
  --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh \
  --flow-logs-enabled \
  --flow-logs-s3-bucket flowlogs-abc \
  --flow-logs-s3-prefix bucketprefix-abc

```

Salida:

```
{
```

```

    "AcceleratorAttributes": {
      "FlowLogsEnabled": true
      "FlowLogsS3Bucket": flowlogs-abc
      "FlowLogsS3Prefix": bucketprefix-abc
    }
  }
}

```

Para obtener más información, consulte los [aceleradores de enrutamiento personalizados en AWS Global Accelerator](#) en la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia de [UpdateCustomRoutingAcceleratorAttributes](#) comandos AWS CLI .

## update-custom-routing-accelerator

En el siguiente ejemplo de código, se muestra cómo usar `update-custom-routing-accelerator`.

### AWS CLI

Para actualizar un acelerador de enrutamiento personalizado

El siguiente `update-custom-routing-accelerator` ejemplo modifica un acelerador de enrutamiento personalizado para cambiar el nombre del acelerador.

```

aws globalaccelerator --region us-west-2 update-custom-routing-accelerator \
  --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh \
  --name ExampleCustomRoutingAcceleratorNew

```

Salida:

```

{
  "Accelerator": {
    "AcceleratorArn":
    "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh",
    "IpAddressType": "IPV4",
    "Name": "ExampleCustomRoutingAcceleratorNew",
    "Enabled": true,
    "Status": "IN_PROGRESS",
    "IpSets": [

```

```

    {
      "IpAddresses": [
        "192.0.2.250",
        "198.51.100.52"
      ],
      "IpFamily": "IPv4"
    }
  ],
  "DnsName": "a1234567890abcdef.awsglobalaccelerator.com",
  "CreatedTime": 1232394847,
  "LastModifiedTime": 1232395654
}
}

```

Para obtener más información, consulte los [aceleradores de enrutamiento personalizados en Global Accelerator en la AWS Guía para desarrolladores de Global Accelerator](#).AWS

- Para API obtener más información, consulte la Referencia de [UpdateCustomRoutingAccelerator](#) comandos AWS CLI .

## update-custom-routing-listener

En el siguiente ejemplo de código, se muestra cómo usar `update-custom-routing-listener`.

### AWS CLI

Para actualizar un detector para un acelerador de enrutamiento personalizado

En el siguiente `update-custom-routing-listener` ejemplo, se actualiza un listener para cambiar el rango de puertos.

```

aws globalaccelerator update-custom-routing-listener \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-  
abcd-1234-abcd-1234abcdefgh/listener/0123vxyz \
  --port-ranges FromPort=10000,ToPort=20000

```

Salida:

```

{
  "Listener": {
    "ListenerArn":
      "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-  
abcd-1234abcdefgh/listener/0123vxyz"
  }
}

```



```

    "PortRanges": [
      {
        "FromPort": 10000,
        "ToPort": 20000
      }
    ],
    "Protocol": "TCP"
  }
}

```

Para obtener más información, consulte [Listeners para obtener aceleradores de enrutamiento personalizados en Global Accelerator en la Guía para desarrolladores de AWS Global Accelerator](#).

- Para obtener API más información, consulte la Referencia de comandos [UpdateCustomRoutingListener](#).AWS CLI

## update-endpoint-group

En el siguiente ejemplo de código, se muestra cómo usar `update-endpoint-group`.

### AWS CLI

Para actualizar un grupo de puntos finales

El siguiente `update-endpoint-group` ejemplo agrega tres puntos de conexión a un grupo de puntos de conexión: una dirección IP elásticaALB, una y unaNLB.

```

aws globalaccelerator update-endpoint-group \
  --endpoint-group-
  arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
  abcd-1234abcdefgh/listener/6789vxyz-vxyz-6789-vxyz-6789lmnopqrs/endpoint-group/
  ab8888example \
  --endpoint-configurations \
    EndpointId=eipalloc-eip01234567890abc,Weight=128 \
    EndpointId=arn:aws:elasticloadbalancing:us-east-1:000123456789:loadbalancer/
  app/ALBTesting/alb01234567890xyz,Weight=128 \
    EndpointId=arn:aws:elasticloadbalancing:us-east-1:000123456789:loadbalancer/
  net/NLBTesting/alb01234567890qrs,Weight=128

```

Salida:

```
{
  "EndpointGroup": {
    "TrafficDialPercentage": 100,
    "EndpointDescriptions": [
      {
        "Weight": 128,
        "EndpointId": "eip01234567890abc"
      },
      {
        "Weight": 128,
        "EndpointId": "arn:aws:elasticloadbalancing:us-
east-1:000123456789:loadbalancer/app/ALBTesting/alb01234567890xyz"
      },
      {
        "Weight": 128,
        "EndpointId": "arn:aws:elasticloadbalancing:us-
east-1:000123456789:loadbalancer/net/NLBTesting/alb01234567890qrs"
      }
    ],
    "EndpointGroupArn":
      "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/6789vxyz-vxyz-6789-vxyz-6789lmnopqrs/endpoint-
group/4321abcd-abcd-4321-abcd-4321abcdefg",
    "EndpointGroupRegion": "us-east-1"
  }
}
```

Para obtener más información, consulte [los grupos de puntos finales de AWS Global Accelerator](#) en la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia [UpdateEndpointGroup](#) de AWS CLI comandos.

## update-listener

En el siguiente ejemplo de código, se muestra cómo usar `update-listener`.

### AWS CLI

Para actualizar un oyente

En el siguiente `update-listener` ejemplo, se actualiza un oyente para cambiar el puerto a 100.

```
aws globalaccelerator update-listener \
  --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/0123vxyz \
  --port-ranges FromPort=100,ToPort=100
```

Salida:

```
{
  "Listener": {
    "ListenerArn":
    "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz
    "PortRanges": [
      {
        "FromPort": 100,
        "ToPort": 100
      }
    ],
    "Protocol": "TCP",
    "ClientAffinity": "NONE"
  }
}
```

Para obtener más información, consulte [Listeners in AWS Global Accelerator en](#) la Guía para desarrolladores de AWS Global Accelerator.

- Para API obtener más información, consulte la Referencia de [UpdateListener](#) comandos AWS CLI .

## withdraw-byoip-cidr

En el siguiente ejemplo de código, se muestra cómo usar `withdraw-byoip-cidr`.

### AWS CLI

Para retirar un rango de direcciones

En el siguiente `withdraw-byoip-cidr` ejemplo, se retira un rango de direcciones de AWS Global Accelerator que anunciaste anteriormente para usarlo con tus recursos. AWS

```
aws globalaccelerator withdraw-byoip-cidr \
```

```
--cidr 192.0.2.250/24
```

Salida:

```
{
  "ByoipCidr": {
    "Cidr": "192.0.2.250/24",
    "State": "PENDING_WITHDRAWING"
  }
}
```

Para obtener más información, consulte Incluya [su propia dirección IP en Global Accelerator en la Guía para desarrolladores de AWS](#) [AWS Global Accelerator](#).

- Para API obtener más información, consulte la Referencia [WithdrawByoipCidr](#) de AWS CLI comandos.

## AWS Glue ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Glue.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

Acciones

### **batch-stop-job-run**

En el siguiente ejemplo de código, se muestra cómo usar `batch-stop-job-run`.

AWS CLI

Para detener la ejecución de tareas

El siguiente `batch-stop-job-run` ejemplo detiene la ejecución de un trabajo.

```
aws glue batch-stop-job-run \  
  --job-name "my-testing-job" \  
  --job-run-id jr_852f1de1f29fb62e0ba4166c33970803935d87f14f96cfdee5089d5274a61d3f
```

Salida:

```
{  
  "SuccessfulSubmissions": [  
    {  
      "JobName": "my-testing-job",  
      "JobRunId":  
"jr_852f1de1f29fb62e0ba4166c33970803935d87f14f96cfdee5089d5274a61d3f"  
    }  
  ],  
  "Errors": [],  
  "ResponseMetadata": {  
    "RequestId": "66bd6b90-01db-44ab-95b9-6aeff0e73d88",  
    "HTTPStatusCode": 200,  
    "HTTPHeaders": {  
      "date": "Fri, 16 Oct 2020 20:54:51 GMT",  
      "content-type": "application/x-amz-json-1.1",  
      "content-length": "148",  
      "connection": "keep-alive",  
      "x-amzn-requestid": "66bd6b90-01db-44ab-95b9-6aeff0e73d88"  
    },  
    "RetryAttempts": 0  
  }  
}
```

Para obtener más información, consulte [Ejecuciones de trabajo](#) en la Guía para desarrolladores de AWS Glue.

- Para API obtener más información, consulte [BatchStopJobRun](#) la Referencia de AWS CLI comandos.

## **create-connection**

En el siguiente ejemplo de código, se muestra cómo usar `create-connection`.

## AWS CLI

Para crear una conexión para los almacenes de datos de AWS Glue

El siguiente `create-connection` ejemplo crea una conexión en el catálogo de datos de AWS Glue que proporciona información de conexión para un banco de datos de Kafka.

```
aws glue create-connection \  
  --connection-input '{ \  
    "Name":"conn-kafka-custom", \  
    "Description":"kafka connection with ssl to custom kafka", \  
    "ConnectionType":"KAFKA", \  
    "ConnectionProperties":{ \  
      "KAFKA_BOOTSTRAP_SERVERS":"<Kafka-broker-server-url>:<SSL-Port>", \  
      "KAFKA_SSL_ENABLED":"true", \  
      "KAFKA_CUSTOM_CERT": "s3://bucket/prefix/cert-file.pem" \  
    }, \  
    "PhysicalConnectionRequirements":{ \  
      "SubnetId":"subnet-1234", \  
      "SecurityGroupIdList":["sg-1234"], \  
      "AvailabilityZone":"us-east-1a"} \  
  }' \  
  --region us-east-1 \  
  --endpoint https://glue.us-east-1.amazonaws.com
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Definición de conexiones en el catálogo de datos de AWS Glue](#) en la Guía del desarrollador de AWS Glue.

- Para API obtener más información, consulte [CreateConnection](#) la Referencia de AWS CLI comandos.

## create-database

En el siguiente ejemplo de código, se muestra cómo usar `create-database`.

## AWS CLI

Para crear una base de datos

En el siguiente `create-database` ejemplo, se crea una base de datos en el catálogo de datos de AWS Glue.

```
aws glue create-database \  
  --database-input "{\"Name\":\"tempdb\"}" \  
  --profile my_profile \  
  --endpoint https://glue.us-east-1.amazonaws.com
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Definición de una base de datos en su Catálogo de datos](#) en la Guía para desarrolladores de AWS Glue.

- Para API obtener más información, consulte [CreateDatabase](#) la Referencia de AWS CLI comandos.

## create-job

En el siguiente ejemplo de código, se muestra cómo usar create-job.

### AWS CLI

Creación de un trabajo a fin de transformar datos

El siguiente ejemplo de create-job crea un trabajo de streaming que ejecuta un script almacenado en S3.

```
aws glue create-job \  
  --name my-testing-job \  
  --role AWSGlueServiceRoleDefault \  
  --command '{ \  
    "Name": "gluestreaming", \  
    "ScriptLocation": "s3://DOC-EXAMPLE-BUCKET/folder/" \  
  }' \  
  --region us-east-1 \  
  --output json \  
  --default-arguments '{ \  
    "--job-language":"scala", \  
    "--class":"GlueApp" \  
  }' \  
  --profile my-profile \  
  --endpoint https://glue.us-east-1.amazonaws.com
```

Contenidos de test\_script.scala:

```
import com.amazonaws.services.glue.ChoiceOption
import com.amazonaws.services.glue.GlueContext
import com.amazonaws.services.glue.MappingSpec
import com.amazonaws.services.glue.ResolveSpec
import com.amazonaws.services.glue.errors.CallSite
import com.amazonaws.services.glue.util.GlueArgParser
import com.amazonaws.services.glue.util.Job
import com.amazonaws.services.glue.util.JsonOptions
import org.apache.spark.SparkContext
import scala.collection.JavaConverters._

object GlueApp {
  def main(sysArgs: Array[String]) {
    val spark: SparkContext = new SparkContext()
    val glueContext: GlueContext = new GlueContext(spark)
    // @params: [JOB_NAME]
    val args = GlueArgParser.getResolvedOptions(sysArgs,
Seq("JOB_NAME").toArray)
    Job.init(args("JOB_NAME"), glueContext, args.asJava)
    // @type: DataSource
    // @args: [database = "tempdb", table_name = "s3-source", transformation_ctx
= "datasource0"]
    // @return: datasource0
    // @inputs: []
    val datasource0 = glueContext.getCatalogSource(database = "tempdb",
tableName = "s3-source", redshiftTmpDir = "", transformationContext =
"datasource0").getDynamicFrame()
    // @type: ApplyMapping
    // @args: [mapping = [("sensorid", "int", "sensorid", "int"),
("currenttemperature", "int", "currenttemperature", "int"), ("status", "string",
"status", "string")], transformation_ctx = "applymapping1"]
    // @return: applymapping1
    // @inputs: [frame = datasource0]
    val applymapping1 = datasource0.applyMapping(mappings = Seq(("sensorid",
"int", "sensorid", "int"), ("currenttemperature", "int", "currenttemperature",
"int"), ("status", "string", "status", "string")), caseSensitive = false,
transformationContext = "applymapping1")
    // @type: SelectFields
    // @args: [paths = ["sensorid", "currenttemperature", "status"],
transformation_ctx = "selectfields2"]
    // @return: selectfields2
    // @inputs: [frame = applymapping1]
```



```

    val selectfields2 = applymapping1.selectFields(paths = Seq("sensorid",
"currenttemperature", "status"), transformationContext = "selectfields2")
    // @type: ResolveChoice
    // @args: [choice = "MATCH_CATALOG", database = "tempdb", table_name = "my-
s3-sink", transformation_ctx = "resolvechoice3"]
    // @return: resolvechoice3
    // @inputs: [frame = selectfields2]
    val resolvechoice3 = selectfields2.resolveChoice(choiceOption =
Some(ChoiceOption("MATCH_CATALOG")), database = Some("tempdb"), tableName =
Some("my-s3-sink"), transformationContext = "resolvechoice3")
    // @type: DataSink
    // @args: [database = "tempdb", table_name = "my-s3-sink",
transformation_ctx = "datasink4"]
    // @return: datasink4
    // @inputs: [frame = resolvechoice3]
    val datasink4 = glueContext.getCatalogSink(database = "tempdb",
tableName = "my-s3-sink", redshiftTmpDir = "", transformationContext =
"datasink4").writeDynamicFrame(resolvechoice3)
    Job.commit()
  }
}

```

Salida:

```

{
  "Name": "my-testing-job"
}

```

Para obtener más información, consulte [Creación de trabajos en AWS Glue en la Guía para desarrolladores de AWS Glue](#).

- Para API obtener más información, consulte [CreateJob](#) la Referencia de AWS CLI comandos.

## create-table

En el siguiente ejemplo de código, se muestra cómo usar create-table.

### AWS CLI

Ejemplo 1: Para crear una tabla para una transmisión de datos de Kinesis

El siguiente create-table ejemplo crea una tabla en el catálogo de datos de AWS Glue que describe una transmisión de datos de Kinesis.

```
aws glue create-table \
  --database-name tempdb \
  --table-input '{"Name":"test-kinesis-input", "StorageDescriptor":{ \
    "Columns":[ \
      {"Name":"sensorid", "Type":"int"}, \
      {"Name":"currenttemperature", "Type":"int"}, \
      {"Name":"status", "Type":"string"} \
    ], \
    "Location":"my-testing-stream", \
    "Parameters":{ \
      "typeOfData":"kinesis","streamName":"my-testing-stream", \
      "kinesisUrl":"https://kinesis.us-east-1.amazonaws.com" \
    }, \
    "SerdeInfo":{ \
      "SerializationLibrary":"org.openx.data.jsonserde.JsonSerDe"} \
  }, \
  "Parameters":{ \
    "classification":"json"} \
}' \
  --profile my-profile \
  --endpoint https://glue.us-east-1.amazonaws.com
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Definición de tablas en el catálogo de datos de AWS Glue](#) en la Guía del desarrollador de AWS Glue.

Ejemplo 2: Para crear una tabla para un banco de datos de Kafka

El siguiente create-table ejemplo crea una tabla en el catálogo de datos de AWS Glue que describe un banco de datos de Kafka.

```
aws glue create-table \
  --database-name tempdb \
  --table-input '{"Name":"test-kafka-input", "StorageDescriptor":{ \
    "Columns":[ \
      {"Name":"sensorid", "Type":"int"}, \
      {"Name":"currenttemperature", "Type":"int"}, \
      {"Name":"status", "Type":"string"} \
    ], \
    "Location":"glue-topic", \
    "Parameters":{ \
```

```

        "typeOfData":"kafka","topicName":"glue-topic", \
        "connectionName":"my-kafka-connection"
    }, \
    "SerdeInfo":{ \
        "SerializationLibrary":"org.apache.hadoop.hive.serde2.OpenCSVSerde"}
\
    }, \
    "Parameters":{ \
        "separatorChar":"," } \
    }' \
--profile my-profile \
--endpoint https://glue.us-east-1.amazonaws.com

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Definición de tablas en el catálogo de datos de AWS Glue](#) en la Guía del desarrollador de AWS Glue.

Ejemplo 3: Para crear una tabla para un banco de datos de AWS S3

En el siguiente `create-table` ejemplo, se crea una tabla en el catálogo de datos de AWS Glue que describe un almacén de datos de AWS Simple Storage Service (AWS S3).

```

aws glue create-table \
  --database-name tempdb \
  --table-input '{"Name":"s3-output", "StorageDescriptor":{ \
    "Columns":[ \
      {"Name":"s1", "Type":"string"}, \
      {"Name":"s2", "Type":"int"}, \
      {"Name":"s3", "Type":"string"}
    ], \
    "Location":"s3://bucket-path/", \
    "SerdeInfo":{ \
      "SerializationLibrary":"org.openx.data.jsonserde.JsonSerDe"} \
  }, \
  "Parameters":{ \
    "classification":"json"} \
}' \
--profile my-profile \
--endpoint https://glue.us-east-1.amazonaws.com

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Definición de tablas en el catálogo de datos de AWS Glue](#) en la Guía del desarrollador de AWS Glue.

- Para API obtener más información, consulte [CreateTable](#) la Referencia de AWS CLI comandos.

## delete-job

En el siguiente ejemplo de código, se muestra cómo usar `delete-job`.

### AWS CLI

Eliminación de un trabajo

En el siguiente ejemplo de `delete-job`, se elimina un trabajo que ya no necesite.

```
aws glue delete-job \  
  --job-name my-testing-job
```

Salida:

```
{  
  "JobName": "my-testing-job"  
}
```

Para obtener más información, consulte Cómo [trabajar con trabajos en la consola de AWS Glue](#) en la Guía para desarrolladores de AWS Glue.

- Para API obtener más información, consulte [DeleteJob](#) la Referencia de AWS CLI comandos.

## get-databases

En el siguiente ejemplo de código, se muestra cómo usar `get-databases`.

### AWS CLI

Para enumerar las definiciones de algunas o todas las bases de datos del catálogo de datos de AWS Glue

El siguiente ejemplo de `get-databases` devuelve información sobre las bases de datos del Catálogo de datos.

```
aws glue get-databases
```

## Salida:

```
{
  "DatabaseList": [
    {
      "Name": "default",
      "Description": "Default Hive database",
      "LocationUri": "file:/spark-warehouse",
      "CreateTime": 1602084052.0,
      "CreateTableDefaultPermissions": [
        {
          "Principal": {
            "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
          },
          "Permissions": [
            "ALL"
          ]
        }
      ],
      "CatalogId": "111122223333"
    },
    {
      "Name": "flights-db",
      "CreateTime": 1587072847.0,
      "CreateTableDefaultPermissions": [
        {
          "Principal": {
            "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
          },
          "Permissions": [
            "ALL"
          ]
        }
      ],
      "CatalogId": "111122223333"
    },
    {
      "Name": "legislators",
      "CreateTime": 1601415625.0,
      "CreateTableDefaultPermissions": [
        {
          "Principal": {
            "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
          },
          "Permissions": [
            "ALL"
          ]
        }
      ],
      "CatalogId": "111122223333"
    }
  ]
}
```

```

        "Permissions": [
            "ALL"
        ]
    },
    ],
    "CatalogId": "111122223333"
},
{
    "Name": "tempdb",
    "CreateTime": 1601498566.0,
    "CreateTableDefaultPermissions": [
        {
            "Principal": {
                "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
            },
            "Permissions": [
                "ALL"
            ]
        }
    ],
    "CatalogId": "111122223333"
}
]
}

```

Para obtener más información, consulte [Definición de una base de datos en su Catálogo de datos](#) en la Guía para desarrolladores de AWS Glue.

- Para API obtener más información, consulte [GetDatabases](#) la Referencia de AWS CLI comandos.

## get-job-run

En el siguiente ejemplo de código, se muestra cómo usar `get-job-run`.

### AWS CLI

Obtención de información sobre una ejecución de trabajo

El siguiente ejemplo de `get-job-run` recupera información sobre una ejecución de trabajo.

```
aws glue get-job-run \
```

```
--job-name "Combine legislators data" \  
--run-id jr_012e176506505074d94d761755e5c62538ee1aad6f17d39f527e9140cf0c9a5e
```

Salida:

```
{  
  "JobRun": {  
    "Id": "jr_012e176506505074d94d761755e5c62538ee1aad6f17d39f527e9140cf0c9a5e",  
    "Attempt": 0,  
    "JobName": "Combine legislators data",  
    "StartedOn": 1602873931.255,  
    "LastModifiedOn": 1602874075.985,  
    "CompletedOn": 1602874075.985,  
    "JobRunState": "SUCCEEDED",  
    "Arguments": {  
      "--enable-continuous-cloudwatch-log": "true",  
      "--enable-metrics": "",  
      "--enable-spark-ui": "true",  
      "--job-bookmark-option": "job-bookmark-enable",  
      "--spark-event-logs-path": "s3://aws-glue-assets-111122223333-us-east-1/  
sparkHistoryLogs/"  
    },  
    "PredecessorRuns": [],  
    "AllocatedCapacity": 10,  
    "ExecutionTime": 117,  
    "Timeout": 2880,  
    "MaxCapacity": 10.0,  
    "WorkerType": "G.1X",  
    "NumberOfWorkers": 10,  
    "LogGroupName": "/aws-glue/jobs",  
    "GlueVersion": "2.0"  
  }  
}
```

Para obtener más información, consulte [Ejecuciones de trabajo](#) en la Guía para desarrolladores de AWS Glue.

- Para API obtener más información, consulte [GetJobRun](#) la Referencia de AWS CLI comandos.

## get-job-runs

En el siguiente ejemplo de código, se muestra cómo usar `get-job-runs`.

## AWS CLI

Obtención de información sobre todas las ejecuciones de trabajo para un trabajo

El siguiente ejemplo de `get-job-runs` recupera información acerca de las ejecuciones de flujo de trabajo para una tarea.

```
aws glue get-job-runs \  
  --job-name "my-testing-job"
```

Salida:

```
{  
  "JobRuns": [  
    {  
      "Id":  
"jr_012e176506505074d94d761755e5c62538ee1aad6f17d39f527e9140cf0c9a5e",  
      "Attempt": 0,  
      "JobName": "my-testing-job",  
      "StartedOn": 1602873931.255,  
      "LastModifiedOn": 1602874075.985,  
      "CompletedOn": 1602874075.985,  
      "JobRunState": "SUCCEEDED",  
      "Arguments": {  
        "--enable-continuous-cloudwatch-log": "true",  
        "--enable-metrics": "",  
        "--enable-spark-ui": "true",  
        "--job-bookmark-option": "job-bookmark-enable",  
        "--spark-event-logs-path": "s3://aws-glue-assets-111122223333-us-  
east-1/sparkHistoryLogs/"  
      },  
      "PredecessorRuns": [],  
      "AllocatedCapacity": 10,  
      "ExecutionTime": 117,  
      "Timeout": 2880,  
      "MaxCapacity": 10.0,  
      "WorkerType": "G.1X",  
      "NumberOfWorkers": 10,  
      "LogGroupName": "/aws-glue/jobs",  
      "GlueVersion": "2.0"  
    },  
  ],  
}
```



```

      "Id":
"jr_03cc19ddb11c4e244d3f735567de74ff93b0b3ef468a713ffe73e53d1aec08f_attempt_2",
      "Attempt": 2,
      "PreviousRunId":
"jr_03cc19ddb11c4e244d3f735567de74ff93b0b3ef468a713ffe73e53d1aec08f_attempt_1",
      "JobName": "my-testing-job",
      "StartedOn": 1602811168.496,
      "LastModifiedOn": 1602811282.39,
      "CompletedOn": 1602811282.39,
      "JobRunState": "FAILED",
      "ErrorMessage": "An error occurred while calling
o122.pyWriteDynamicFrame.
          Access Denied (Service: Amazon S3; Status Code: 403; Error Code:
AccessDenied;
          Request ID: 021AAB703DB20A2D;
          S3 Extended Request ID: teZk24Y09TkXzBvMPG502L5VJBhe9DJuWA9/
TXtuG0qfByajkFL/Tlqt5JBGdEGpigAqzdMDM/U=)",
      "PredecessorRuns": [],
      "AllocatedCapacity": 10,
      "ExecutionTime": 110,
      "Timeout": 2880,
      "MaxCapacity": 10.0,
      "WorkerType": "G.1X",
      "NumberOfWorkers": 10,
      "LogGroupName": "/aws-glue/jobs",
      "GlueVersion": "2.0"
    },
    {
      "Id":
"jr_03cc19ddb11c4e244d3f735567de74ff93b0b3ef468a713ffe73e53d1aec08f_attempt_1",
      "Attempt": 1,
      "PreviousRunId":
"jr_03cc19ddb11c4e244d3f735567de74ff93b0b3ef468a713ffe73e53d1aec08f",
      "JobName": "my-testing-job",
      "StartedOn": 1602811020.518,
      "LastModifiedOn": 1602811138.364,
      "CompletedOn": 1602811138.364,
      "JobRunState": "FAILED",
      "ErrorMessage": "An error occurred while calling
o122.pyWriteDynamicFrame.
          Access Denied (Service: Amazon S3; Status Code: 403; Error Code:
AccessDenied;
          Request ID: 2671D37856AE7ABB;

```

```

S3 Extended Request ID: RLJCJw20brV
+PpC6Gp0RahyF2fp9f1B5SSb2bTGPnUSPVizLXR11PN3QZ1db+v1o9qRVktNYbW8=)",
  "PredecessorRuns": [],
  "AllocatedCapacity": 10,
  "ExecutionTime": 113,
  "Timeout": 2880,
  "MaxCapacity": 10.0,
  "WorkerType": "G.1X",
  "NumberOfWorkers": 10,
  "LogGroupName": "/aws-glue/jobs",
  "GlueVersion": "2.0"
}
]
}

```

Para obtener más información, consulte [Ejecuciones de trabajo](#) en la Guía para desarrolladores de AWS Glue.

- Para API obtener más información, consulte [GetJobRuns](#) la Referencia de AWS CLI comandos.

## get-job

En el siguiente ejemplo de código, se muestra cómo usar `get-job`.

### AWS CLI

Recuperación de información sobre un trabajo

El siguiente ejemplo de `get-job` recupera información sobre un trabajo.

```

aws glue get-job \
  --job-name my-testing-job

```

Salida:

```

{
  "Job": {
    "Name": "my-testing-job",
    "Role": "Glue_DefaultRole",
    "CreatedOn": 1602805698.167,
    "LastModifiedOn": 1602805698.167,
    "ExecutionProperty": {

```

```

        "MaxConcurrentRuns": 1
    },
    "Command": {
        "Name": "gluestreaming",
        "ScriptLocation": "s3://janetst-bucket-01/Scripts/test_script.scala",
        "PythonVersion": "2"
    },
    "DefaultArguments": {
        "--class": "GlueApp",
        "--job-language": "scala"
    },
    "MaxRetries": 0,
    "AllocatedCapacity": 10,
    "MaxCapacity": 10.0,
    "GlueVersion": "1.0"
}
}

```

Para obtener más información, consulte [Trabajos](#) en la Guía para desarrolladores de AWS Glue.

- Para API obtener más información, consulte [GetJob](#) la Referencia de AWS CLI comandos.

## get-plan

En el siguiente ejemplo de código, se muestra cómo usar `get-plan`.

### AWS CLI

Para obtener el código generado para mapear datos de las tablas de origen a las tablas de destino

A continuación, se `get-plan` recupera el código generado para mapear columnas de la fuente de datos al destino de datos.

```

aws glue get-plan --mapping '[ \
  { \
    "SourcePath": "sensorid", \
    "SourceTable": "anything", \
    "SourceType": "int", \
    "TargetPath": "sensorid", \
    "TargetTable": "anything", \
    "TargetType": "int" \
  }, \
]

```

```

{ \
  "SourcePath":"currenttemperature", \
  "SourceTable":"anything", \
  "SourceType":"int", \
  "TargetPath":"currenttemperature", \
  "TargetTable":"anything", \
  "TargetType":"int" \
}, \
{ \
  "SourcePath":"status", \
  "SourceTable":"anything", \
  "SourceType":"string", \
  "TargetPath":"status", \
  "TargetTable":"anything", \
  "TargetType":"string" \
}]' \
--source '{ \
  "DatabaseName":"tempdb", \
  "TableName":"s3-source" \
}' \
--sinks '[' \
  { \
    "DatabaseName":"tempdb", \
    "TableName":"my-s3-sink" \
  } \
]' \
--language "scala"
--endpoint https://glue.us-east-1.amazonaws.com
--output "text"

```

Salida:

```

import com.amazonaws.services.glue.ChoiceOption
import com.amazonaws.services.glue.GlueContext
import com.amazonaws.services.glue.MappingSpec
import com.amazonaws.services.glue.ResolveSpec
import com.amazonaws.services.glue.errors.CallSite
import com.amazonaws.services.glue.util.GlueArgParser
import com.amazonaws.services.glue.util.Job
import com.amazonaws.services.glue.util.JsonOptions
import org.apache.spark.SparkContext
import scala.collection.JavaConverters._

object GlueApp {

```

```

def main(sysArgs: Array[String]) {
  val spark: SparkContext = new SparkContext()
  val glueContext: GlueContext = new GlueContext(spark)
  // @params: [JOB_NAME]
  val args = GlueArgParser.getResolvedOptions(sysArgs, Seq("JOB_NAME").toArray)
  Job.init(args("JOB_NAME"), glueContext, args.asJava)
  // @type: DataSource
  // @args: [database = "tempdb", table_name = "s3-source", transformation_ctx =
"datasource0"]
  // @return: datasource0
  // @inputs: []
  val datasource0 = glueContext.getCatalogSource(database = "tempdb",
tableName = "s3-source", redshiftTmpDir = "", transformationContext =
"datasource0").getDynamicFrame()
  // @type: ApplyMapping
  // @args: [mapping = [("sensorid", "int", "sensorid", "int"),
("currenttemperature", "int", "currenttemperature", "int"), ("status", "string",
"status", "string")], transformation_ctx = "applymapping1"]
  // @return: applymapping1
  // @inputs: [frame = datasource0]
  val applymapping1 = datasource0.applyMapping(mappings = Seq(("sensorid",
"int", "sensorid", "int"), ("currenttemperature", "int", "currenttemperature",
"int"), ("status", "string", "status", "string")), caseSensitive = false,
transformationContext = "applymapping1")
  // @type: SelectFields
  // @args: [paths = ["sensorid", "currenttemperature", "status"],
transformation_ctx = "selectfields2"]
  // @return: selectfields2
  // @inputs: [frame = applymapping1]
  val selectfields2 = applymapping1.selectFields(paths = Seq("sensorid",
"currenttemperature", "status"), transformationContext = "selectfields2")
  // @type: ResolveChoice
  // @args: [choice = "MATCH_CATALOG", database = "tempdb", table_name = "my-s3-
sink", transformation_ctx = "resolvechoice3"]
  // @return: resolvechoice3
  // @inputs: [frame = selectfields2]
  val resolvechoice3 = selectfields2.resolveChoice(choiceOption =
Some(ChoiceOption("MATCH_CATALOG")), database = Some("tempdb"), tableName =
Some("my-s3-sink"), transformationContext = "resolvechoice3")
  // @type: DataSink
  // @args: [database = "tempdb", table_name = "my-s3-sink", transformation_ctx =
"datasink4"]
  // @return: datasink4
  // @inputs: [frame = resolvechoice3]

```

```
    val datasink4 = glueContext.getCatalogSink(database = "tempdb",
tableName = "my-s3-sink", redshiftTmpDir = "", transformationContext =
"datasink4").writeDynamicFrame(resolvechoice3)
    Job.commit()
  }
}
```

Para obtener más información, consulte [Edición de scripts en AWS Glue](#) in the AWS Glue Developer Guide.

- Para API obtener más información, consulte [GetPlan](#) la Referencia de AWS CLI comandos.

## get-tables

En el siguiente ejemplo de código, se muestra cómo usar `get-tables`.

### AWS CLI

Enumeración de las definiciones de algunas o todas las tablas de la base de datos especificada

El siguiente ejemplo de `get-tables` devuelve información sobre las tablas de la base de datos especificada.

```
aws glue get-tables --database-name 'tempdb'
```

Salida:

```
{
  "TableList": [
    {
      "Name": "my-s3-sink",
      "DatabaseName": "tempdb",
      "CreateTime": 1602730539.0,
      "UpdateTime": 1602730539.0,
      "Retention": 0,
      "StorageDescriptor": {
        "Columns": [
          {
            "Name": "sensorid",
            "Type": "int"
          },
          {
            "Name": "currenttemperature",
```

```
        "Type": "int"
      },
      {
        "Name": "status",
        "Type": "string"
      }
    ],
    "Location": "s3://janetst-bucket-01/test-s3-output/",
    "Compressed": false,
    "NumberOfBuckets": 0,
    "SerdeInfo": {
      "SerializationLibrary": "org.openx.data.jsonserde.JsonSerDe"
    },
    "SortColumns": [],
    "StoredAsSubDirectories": false
  },
  "Parameters": {
    "classification": "json"
  },
  "CreatedBy": "arn:aws:iam::007436865787:user/JRSTERN",
  "IsRegisteredWithLakeFormation": false,
  "CatalogId": "007436865787"
},
{
  "Name": "s3-source",
  "DatabaseName": "tempdb",
  "CreateTime": 1602730658.0,
  "UpdateTime": 1602730658.0,
  "Retention": 0,
  "StorageDescriptor": {
    "Columns": [
      {
        "Name": "sensorid",
        "Type": "int"
      },
      {
        "Name": "currenttemperature",
        "Type": "int"
      },
      {
        "Name": "status",
        "Type": "string"
      }
    ]
  }
},
```

```
    "Location": "s3://janetst-bucket-01/",
    "Compressed": false,
    "NumberOfBuckets": 0,
    "SortColumns": [],
    "StoredAsSubDirectories": false
  },
  "Parameters": {
    "classification": "json"
  },
  "CreatedBy": "arn:aws:iam::007436865787:user/JRSTERN",
  "IsRegisteredWithLakeFormation": false,
  "CatalogId": "007436865787"
},
{
  "Name": "test-kinesis-input",
  "DatabaseName": "tempdb",
  "CreateTime": 1601507001.0,
  "UpdateTime": 1601507001.0,
  "Retention": 0,
  "StorageDescriptor": {
    "Columns": [
      {
        "Name": "sensorid",
        "Type": "int"
      },
      {
        "Name": "currenttemperature",
        "Type": "int"
      },
      {
        "Name": "status",
        "Type": "string"
      }
    ]
  },
  "Location": "my-testing-stream",
  "Compressed": false,
  "NumberOfBuckets": 0,
  "SerdeInfo": {
    "SerializationLibrary": "org.openx.data.jsonserde.JsonSerDe"
  },
  "SortColumns": [],
  "Parameters": {
    "kinesisUrl": "https://kinesis.us-east-1.amazonaws.com",
    "streamName": "my-testing-stream",
```



```
        "typeOfData": "kinesis"
      },
      "StoredAsSubDirectories": false
    },
    "Parameters": {
      "classification": "json"
    },
    "CreatedBy": "arn:aws:iam::007436865787:user/JRSTERN",
    "IsRegisteredWithLakeFormation": false,
    "CatalogId": "007436865787"
  }
]
}
```

Para obtener más información, consulte [Definición de tablas en el catálogo de datos de AWS Glue](#) en la Guía del desarrollador de AWS Glue.

- Para API obtener más información, consulte [GetTables](#) la Referencia de AWS CLI comandos.

## start-crawler

En el siguiente ejemplo de código, se muestra cómo usar `start-crawler`.

### AWS CLI

Inicio de un rastreador

El siguiente ejemplo de `start-crawler` inicia un rastreador.

```
aws glue start-crawler --name my-crawler
```

Salida:

```
None
```

Para obtener más información, consulte [Definición de rastreadores](#) en la Guía para desarrolladores de AWS Glue.

- Para API obtener más información, consulte [StartCrawler](#) la Referencia de AWS CLI comandos.

## start-job-run

En el siguiente ejemplo de código, se muestra cómo usar `start-job-run`.

### AWS CLI

Inicio de la ejecución de un trabajo

El siguiente ejemplo de `start-job-run` inicia un trabajo.

```
aws glue start-job-run \  
  --job-name my-job
```

Salida:

```
{  
  "JobRunId":  
  "jr_22208b1f44eb5376a60569d4b21dd20fcb8621e1a366b4e7b2494af764b82ded"  
}
```

Para obtener más información, consulte [Creación de trabajos](#) en la Guía para desarrolladores de AWS Glue.

- Para API obtener más información, consulte [StartJobRun](#) la Referencia de AWS CLI comandos.

## GuardDuty ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with GuardDuty.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **accept-invitation**

En el siguiente ejemplo de código, se muestra cómo usar `accept-invitation`.

#### AWS CLI

Para aceptar una invitación para convertirse en una cuenta de GuardDuty miembro en la región actual

El siguiente `accept-invitation` ejemplo muestra cómo aceptar una invitación para convertirse en una cuenta de GuardDuty miembro en la región actual.

```
aws guardduty accept-invitation \
  --detector-id 12abc34d567e8fa901bc2d34eexample \
  --master-id 123456789111 \
  --invitation-id d6b94fb03a66ff665f7db8764example
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de GuardDuty cuentas mediante invitación](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [AcceptInvitation](#) la Referencia de AWS CLI comandos.

### **archive-findings**

En el siguiente ejemplo de código, se muestra cómo usar `archive-findings`.

#### AWS CLI

Para archivar los hallazgos en la región actual

En este ejemplo, se muestra cómo archivar los hallazgos en la región actual.

```
aws guardduty archive-findings \
  --detector-id 12abc34d567e8fa901bc2d34eexample \
  --finding-ids d6b94fb03a66ff665f7db8764example 3eb970e0de00c16ec14e6910fexample
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de GuardDuty cuentas mediante invitación](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [ArchiveFindings](#) la Referencia de AWS CLI comandos.

## **create-detector**

En el siguiente ejemplo de código, se muestra cómo usar `create-detector`.

### AWS CLI

Para habilitarlo GuardDuty en la región actual

En este ejemplo se muestra cómo crear un nuevo detector, que lo habilite GuardDuty, en la región actual. :

```
aws guardduty create-detector \  
  --enable
```

Salida:

```
{  
  "DetectorId": "b6b992d6d2f48e64bc59180bfexample"  
}
```

Para obtener más información, consulta [Habilitar Amazon GuardDuty](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [CreateDetector](#) la Referencia de AWS CLI comandos.

## **create-filter**

En el siguiente ejemplo de código, se muestra cómo usar `create-filter`.

### AWS CLI

Para crear un filtro nuevo para la región actual

En este ejemplo, se crea un filtro que coincide con todos los resultados del escaneo de puertos, por ejemplo, los creados a partir de una imagen específica. :

```
aws guardduty create-filter \  
  --detector-id b6b992d6d2f48e64bc59180bfexample \  
  --action ARCHIVE \  
  --name myFilter \  
  --finding-criteria '{"Criterion": {"type": {"Eq": ["Recon:EC2/  
Portscan"]}, "resource.instanceDetails.imageId": {"Eq": ["ami-0a7a207083example"]}}}'
```

Salida:

```
{  
  "Name": "myFilter"  
}
```

Para obtener más información, consulte [Filtrar los resultados](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [CreateFilter](#) la Referencia de AWS CLI comandos.

## create-ip-set

En el siguiente ejemplo de código, se muestra cómo usar `create-ip-set`.

### AWS CLI

Para crear un conjunto de direcciones IP de confianza

El siguiente `create-ip-set` ejemplo crea y activa un conjunto de direcciones IP de confianza en la región actual.

```
aws guardduty create-ip-set \  
  --detector-id 12abc34d567e8fa901bc2d34eexample \  
  --name new-ip-set \  
  --format TXT \  
  --location s3://AWSDOC-EXAMPLE-BUCKET/customtrustlist.csv \  
  --activate
```

Salida:

```
{  
  "IpSetId": "d4b94fc952d6912b8f3060768example"  
}
```

Para obtener más información, consulte [Trabajar con listas de IP confiables y listas de amenazas](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [CreateIpSet](#) la Referencia de AWS CLI comandos.

## create-members

En el siguiente ejemplo de código, se muestra cómo usar `create-members`.

### AWS CLI

Para asociar un nuevo miembro a su cuenta GuardDuty maestra en la región actual.

En este ejemplo se muestra cómo asociar las cuentas de los miembros para que las gestione la cuenta corriente como GuardDuty maestra.

```
aws guardduty create-members
  --detector-id b6b992d6d2f48e64bc59180bfexample \
  --account-details AccountId=111122223333,Email=first
+member@example.com AccountId=111111111111 ,Email=another+member@example.com
```

Salida:

```
{
  "UnprocessedAccounts": []
}
```

Para obtener más información, consulte [Administrar varias cuentas](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [CreateMembers](#) la Referencia de AWS CLI comandos.

## create-publishing-destination

En el siguiente ejemplo de código, se muestra cómo usar `create-publishing-destination`.

### AWS CLI

Para crear un destino de publicación al que exportar GuardDuty los resultados de la región actual.

En este ejemplo se muestra cómo crear un destino de publicación para GuardDuty los hallazgos.

```
aws guardduty create-publishing-destination \  
  --detector-id b6b992d6d2f48e64bc59180bfexample \  
  --destination-type S3 \  
  --destination-  
properties DestinationArn=arn:aws:s3:::yourbucket,KmsKeyArn=arn:aws:kms:us-  
west-1:111122223333:key/84cee9c5-dea1-401a-ab6d-e1de7example
```

Salida:

```
{  
  "DestinationId": "46b99823849e1bbc242dfbe3cexample"  
}
```

Para obtener más información, consulte [Exportación de los resultados](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [CreatePublishingDestination](#) la Referencia de AWS CLI comandos.

## create-sample-findings

En el siguiente ejemplo de código, se muestra cómo usar `create-sample-findings`.

AWS CLI

Para crear ejemplos de GuardDuty hallazgos en la región actual.

En este ejemplo se muestra cómo crear un resultado de muestra de los tipos proporcionados.

```
aws guardduty create-sample-findings \  
  --detector-id b6b992d6d2f48e64bc59180bfexample \  
  --finding-types UnauthorizedAccess:EC2/TorClient UnauthorizedAccess:EC2/TorRelay
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Ejemplos de resultados](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [CreateSampleFindings](#) la Referencia de AWS CLI comandos.

## create-threat-intel-set

En el siguiente ejemplo de código, se muestra cómo usar `create-threat-intel-set`.

### AWS CLI

Para crear un nuevo conjunto de información sobre amenazas en la región actual.

En este ejemplo se muestra cómo cargar un conjunto de información sobre amenazas GuardDuty y activarlo inmediatamente.

```
aws guardduty create-threat-intel-set \  
  --detector-id b6b992d6d2f48e64bc59180bfexample \  
  --name myThreatSet \  
  --format TXT \  
  --location s3://EXAMPLEBUCKET/threatlist.csv \  
  --activate
```

Salida:

```
{  
  "ThreatIntelSetId": "20b9a4691aeb33506b808878cexample"  
}
```

Para obtener más información, consulte [IP de confianza y listas de amenazas](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [CreateThreatIntelSet](#) la Referencia de AWS CLI comandos.

## decline-invitations

En el siguiente ejemplo de código, se muestra cómo usar `decline-invitations`.

### AWS CLI

Rechazar una invitación para que Guardduty sea gestionado por otra cuenta de la región actual.

En este ejemplo se muestra cómo rechazar una invitación de membresía.

```
aws guardduty decline-invitations \  
  --account-ids 111122223333
```



Salida:

```
{
  "UnprocessedAccounts": []
}
```

Para obtener más información, consulte [Administrar GuardDuty cuentas mediante invitación](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [DeclineInvitations](#) la Referencia de AWS CLI comandos.

## delete-detector

En el siguiente ejemplo de código, se muestra cómo usar `delete-detector`.

AWS CLI

Para eliminar un detector y GuardDuty desactivarlo en la región actual.

En este ejemplo se muestra cómo eliminar un detector. Si se elimina correctamente, se desactivará GuardDuty en la región asociada a ese detector.

```
aws guardduty delete-detector \
  --detector-id b6b992d6d2f48e64bc59180bfexample
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Suspender o deshabilitar GuardDuty](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte la Referencia [DeleteDetector](#) de AWS CLI comandos.

## delete-filter

En el siguiente ejemplo de código, se muestra cómo usar `delete-filter`.

AWS CLI

Para eliminar un filtro existente en la región actual

En este ejemplo se muestra cómo crear y eliminar un filtro.

```
aws guardduty delete-filter \  
  --detector-id b6b992d6d2f48e64bc59180bfexample \  
  --filter-name byebyeFilter
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Filtrar los resultados](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [DeleteFilter](#) la Referencia de AWS CLI comandos.

### **disable-organization-admin-account**

En el siguiente ejemplo de código, se muestra cómo usar `disable-organization-admin-account`.

AWS CLI

Para eliminar una cuenta como administrador delegado de GuardDuty su organización

En este ejemplo se muestra cómo eliminar una cuenta como administrador delegado de GuardDuty

```
aws guardduty disable-organization-admin-account \  
  --admin-account-id 111122223333
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar cuentas con AWS organizaciones](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [DisableOrganizationAdminAccount](#) la Referencia de AWS CLI comandos.

### **disassociate-from-master-account**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-from-master-account`.

## AWS CLI

Para desasociarse de su cuenta maestra actual en la región actual

El siguiente `disassociate-from-master-account` ejemplo disocia su cuenta de la cuenta GuardDuty maestra actual de la región actual. AWS

```
aws guardduty disassociate-from-master-account \  
  --detector-id d4b040365221be2b54a6264dcexample
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Descripción de la relación entre las cuentas GuardDuty principales y las cuentas de los miembros](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [DisassociateFromMasterAccount](#) la Referencia de AWS CLI comandos.

## get-detector

En el siguiente ejemplo de código, se muestra cómo usar `get-detector`.

## AWS CLI

Para recuperar los detalles de un detector específico

En el siguiente `get-detector` ejemplo, se muestran los detalles de configuración del detector especificado.

```
aws guardduty get-detector \  
  --detector-id 12abc34d567e8fa901bc2d34eexample
```

Salida:

```
{  
  "Status": "ENABLED",  
  "ServiceRole": "arn:aws:iam::111122223333:role/aws-service-role/  
guardduty.amazonaws.com/AWSServiceRoleForAmazonGuardDuty",  
  "Tags": {},  
  "FindingPublishingFrequency": "SIX_HOURS",  
  "UpdatedAt": "2018-11-07T03:24:22.938Z",  
  "CreatedAt": "2017-12-22T22:51:31.940Z"
```

```
}
```

Para obtener más información, consulte [Conceptos y terminología](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [GetDetector](#) la Referencia de AWS CLI comandos.

## get-findings

En el siguiente ejemplo de código, se muestra cómo usar `get-findings`.

### AWS CLI

Ejemplo 1: Para recuperar los detalles de un hallazgo específico

En el siguiente `get-findings` ejemplo, se recuperan todos JSON los detalles de la búsqueda especificada.

```
aws guardduty get-findings \  
  --detector-id 12abc34d567e8fa901bc2d34eexample \  
  --finding-id 1ab92989eaf0e742df4a014d5example
```

Salida:

```
{  
  "Findings": [  
    {  
      "Resource": {  
        "ResourceType": "AccessKey",  
        "AccessKeyDetails": {  
          "UserName": "testuser",  
          "UserType": "IAMUser",  
          "PrincipalId": "AIDACKCEVSQ6C2EXAMPLE",  
          "AccessKeyId": "ASIASZ4SI7REEEXAMPLE"  
        }  
      },  
      "Description": "APIs commonly used to discover the users, groups,  
policies and permissions in an account, was invoked by IAM principal testuser under  
unusual circumstances. Such activity is not typically seen from this principal.",  
      "Service": {  
        "Count": 5,  
        "Archived": false,  
        "ServiceName": "guardduty",  
      }  
    }  
  ]  
}
```

```
"EventFirstSeen": "2020-05-26T22:02:24Z",
"ResourceRole": "TARGET",
"EventLastSeen": "2020-05-26T22:33:55Z",
"DetectorId": "d4b040365221be2b54a6264dcexample",
"Action": {
  "ActionType": "AWS_API_CALL",
  "AwsApiCallAction": {
    "RemoteIpDetails": {
      "GeoLocation": {
        "Lat": 51.5164,
        "Lon": -0.093
      },
      "City": {
        "CityName": "London"
      },
      "IpAddressV4": "52.94.36.7",
      "Organization": {
        "Org": "Amazon.com",
        "Isp": "Amazon.com",
        "Asn": "16509",
        "AsnOrg": "AMAZON-02"
      },
      "Country": {
        "CountryName": "United Kingdom"
      }
    },
    "Api": "ListPolicyVersions",
    "ServiceName": "iam.amazonaws.com",
    "CallerType": "Remote IP"
  }
},
"Title": "Unusual user permission reconnaissance activity by testuser.",
"Type": "Recon:IAMUser/UserPermissions",
"Region": "us-east-1",
"Partition": "aws",
"Arn": "arn:aws:guardduty:us-east-1:111122223333:detector/
d4b040365221be2b54a6264dcexample/finding/1ab92989eaf0e742df4a014d5example",
"UpdatedAt": "2020-05-26T22:55:21.703Z",
"SchemaVersion": "2.0",
"Severity": 5,
"Id": "1ab92989eaf0e742df4a014d5example",
"CreatedAt": "2020-05-26T22:21:48.385Z",
"AccountId": "111122223333"
```

```

    }
  ]
}

```

Para obtener más información, consulte las [conclusiones](#) de la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [GetFindings](#) la Referencia de AWS CLI comandos.

## get-ip-set

En el siguiente ejemplo de código, se muestra cómo usar `get-ip-set`.

### AWS CLI

Para ver una lista, obtenga detalles sobre un conjunto de IP de confianza específico

El siguiente `get-ip-set` ejemplo muestra el estado y los detalles del conjunto de direcciones IP de confianza especificado.

```

aws guardduty get-ip-set \
  --detector-id 12abc34d567e8fa901bc2d34eexample \
  --ip-set-id d4b94fc952d6912b8f3060768example

```

Salida:

```

{
  "Status": "ACTIVE",
  "Location": "s3://AWSDOC-EXAMPLE-BUCKET.s3-us-west-2.amazonaws.com/
customlist.csv",
  "Tags": {},
  "Format": "TXT",
  "Name": "test-ip-set"
}

```

Para obtener más información, consulte [Trabajar con listas de direcciones IP confiables y listas de amenazas](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [GetIpSet](#) la Referencia de AWS CLI comandos.

## get-master-account

En el siguiente ejemplo de código, se muestra cómo usar `get-master-account`.

## AWS CLI

Para recuperar los detalles de su cuenta maestra en la región actual

El siguiente `get-master-account` ejemplo muestra el estado y los detalles de la cuenta maestra asociada al detector en la región actual.

```
aws guardduty get-master-account \  
--detector-id 12abc34d567e8fa901bc2d34eexample
```

Salida:

```
{  
  "Master": {  
    "InvitationId": "04b94d9704854a73f94e061e8example",  
    "InvitedAt": "2020-06-09T22:23:04.970Z",  
    "RelationshipStatus": "Enabled",  
    "AccountId": "123456789111"  
  }  
}
```

Para obtener más información, consulte [Descripción de la relación entre las cuentas GuardDuty principales y las de los miembros](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [GetMasterAccount](#) la Referencia de AWS CLI comandos.

## list-detectors

En el siguiente ejemplo de código, se muestra cómo usar `list-detectors`.

### AWS CLI

Para ver una lista de los detectores disponibles en la región actual

En el siguiente `list-detectors` ejemplo, se enumeran los detectores disponibles en AWS la región actual.

```
aws guardduty list-detectors
```

Salida:

```
{
  "DetectorIds": [
    "12abc34d567e8fa901bc2d34eexample"
  ]
}
```

Para obtener más información, consulte [Conceptos y terminología](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [ListDetectors](#) la Referencia de AWS CLI comandos.

## list-findings

En el siguiente ejemplo de código, se muestra cómo usar `list-findings`.

### AWS CLI

Ejemplo 1: Para enumerar todos los resultados de la región actual

En el siguiente `list-findings` ejemplo, se muestra una lista de todos `findingIds` los datos de la región actual ordenados por gravedad, de mayor a menor.

```
aws guardduty list-findings \
  --detector-id 12abc34d567e8fa901bc2d34eexample \
  --sort-criteria '{"AttributeName": "severity", "OrderBy": "DESC"}
```

Salida:

```
{
  "FindingIds": [
    "04b8ab50fd29c64fc771b232dexample",
    "5ab8ab50fd21373735c826d3aexample",
    "90b93de7aba69107f05bbe60bexample",
    ...
  ]
}
```

Para obtener más información, consulte [las conclusiones](#) de la Guía GuardDuty del usuario.

Ejemplo 2: Listar los hallazgos de la región actual que coincidan con un criterio de hallazgo específico



En el siguiente `list-findings` ejemplo, se muestra una lista de todos los `findingIds` que coinciden con un tipo de búsqueda especificado.

```
aws guardduty list-findings \
  --detector-id 12abc34d567e8fa901bc2d34eexample \
  --finding-criteria '{"Criterion":{"type": {"Eq":["UnauthorizedAccess:EC2/SSHBruteForce"]}}}'
```

Salida:

```
{
  "FindingIds": [
    "90b93de7aba69107f05bbe60bexample",
    "6eb9430d7023d30774d6f05e3example",
    "2eb91a2d060ac9a21963a5848example",
    "44b8ab50fd2b0039a9e48f570example",
    "9eb8ab4cd2b7e5b66ba4f5e96example",
    "e0b8ab3a38e9b0312cc390ceeexample"
  ]
}
```

Para obtener más información, consulte [las conclusiones](#) de la Guía GuardDuty del usuario.

Ejemplo 3: Listar los hallazgos de la región actual que coinciden con un conjunto específico de criterios de búsqueda definidos en un JSON archivo

En el siguiente `list-findings` ejemplo, se muestra una lista de todos los `findingIds` que no están archivados e incluyen al IAM usuario denominado «testuser», tal como se especifica en un JSON archivo.

```
aws guardduty list-findings \
  --detector-id 12abc34d567e8fa901bc2d34eexample \
  --finding-criteria file://myfile.json
```

Contenidos de `myfile.json`:

```
{"Criterion": {
  "resource.accessKeyDetails.userName":{
    "Eq":[
      "testuser"
    ]
  }
}
```

```
    },
    "service.archived": {
      "Eq": [
        "false"
      ]
    }
  }
}
```

Salida:

```
{
  "FindingIds": [
    "1ab92989eaf0e742df4a014d5example"
  ]
}
```

Para obtener más información, consulte las [conclusiones](#) de la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [ListFindings](#) la Referencia de AWS CLI comandos.

## list-invitations

En el siguiente ejemplo de código, se muestra cómo usar `list-invitations`.

### AWS CLI

Para obtener información detallada sobre sus invitaciones para convertirse en miembro de la cuenta en la región actual

En el siguiente `list-invitations` ejemplo, se enumeran los detalles y los estados de sus invitaciones para convertirse en una cuenta de GuardDuty miembro en la región actual.

```
aws guardduty list-invitations
```

Salida:

```
{
  "Invitations": [
    {
      "InvitationId": "d6b94fb03a66ff665f7db8764example",
      "InvitedAt": "2020-06-10T17:56:38.221Z",
```

```
        "RelationshipStatus": "Invited",
        "AccountId": "123456789111"
    }
]
}
```

Para obtener más información, consulte [Administración de GuardDuty cuentas mediante invitación](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [ListInvitations](#) la Referencia de AWS CLI comandos.

## list-ip-sets

En el siguiente ejemplo de código, se muestra cómo usar `list-ip-sets`.

### AWS CLI

Para ver una lista de los conjuntos de IP de confianza de la región actual

En el siguiente `list-ip-sets` ejemplo, se enumeran los conjuntos de IP de confianza de AWS la región actual.

```
aws guardduty list-ip-sets \
  --detector-id 12abc34d567e8fa901bc2d34eexample
```

Salida:

```
{
  "IpSetIds": [
    "d4b94fc952d6912b8f3060768example"
  ]
}
```

Para obtener más información, consulte [Trabajar con listas de IP confiables y listas de amenazas](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [ListIpSets](#) la Referencia de AWS CLI comandos.

## list-members

En el siguiente ejemplo de código, se muestra cómo usar `list-members`.

## AWS CLI

Para ver una lista de todos los miembros de la región actual

En el siguiente `list-members` ejemplo, se enumeran todas las cuentas de los miembros y sus detalles de la región actual.

```
aws guardduty list-members \
  --detector-id 12abc34d567e8fa901bc2d34example
```

Salida:

```
{
  "Members": [
    {
      "RelationshipStatus": "Enabled",
      "InvitedAt": "2020-06-09T22:49:00.910Z",
      "MasterId": "123456789111",
      "DetectorId": "7ab8b2f61b256c87f793f6a86example",
      "UpdatedAt": "2020-06-09T23:08:22.512Z",
      "Email": "your+member@example.com",
      "AccountId": "123456789222"
    }
  ]
}
```

Para obtener más información, consulte [Descripción de la relación entre las cuentas GuardDuty principales y las de los miembros](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [ListMembers](#) la Referencia de AWS CLI comandos.

## update-ip-set

En el siguiente ejemplo de código, se muestra cómo usar `update-ip-set`.

## AWS CLI

Para actualizar un conjunto de direcciones IP de confianza

El siguiente `update-ip-set` ejemplo muestra cómo actualizar los detalles de un conjunto de direcciones IP de confianza.

```
aws guardduty update-ip-set \  
  --detector-id 12abc34d567e8fa901bc2d34eexample \  
  --ip-set-id d4b94fc952d6912b8f3060768example \  
  --location https://AWSDOC-EXAMPLE-BUCKET.s3-us-west-2.amazonaws.com/  
  customtrustList2.csv
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con listas de direcciones IP confiables y listas de amenazas](#) en la Guía del GuardDuty usuario.

- Para API obtener más información, consulte [UpdateIpSet](#) la Referencia de AWS CLI comandos.

## AWS Health ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Health.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **describe-affected-entities**

En el siguiente ejemplo de código, se muestra cómo usar `describe-affected-entities`.

AWS CLI

Para enumerar las entidades que se ven afectadas por un evento de AWS Salud específico

En el siguiente `describe-affected-entities` ejemplo, se enumeran las entidades que se ven afectadas por el evento AWS Health especificado. Este evento es una notificación de facturación para la AWS cuenta.

```
aws health describe-affected-entities \
  --filter "eventArns=arn:aws:health:global::event/BILLING/
AWS_BILLING_NOTIFICATION/AWS_BILLING_NOTIFICATION_6ce1d874-e995-40e2-99cd-
EXAMPLE11145" \
  --region us-east-1
```

Salida:

```
{
  "entities": [
    {
      "entityArn": "arn:aws:health:global:123456789012:entity/
EXAMPLEimSMoULmWHpb",
      "eventArn": "arn:aws:health:global::event/BILLING/
AWS_BILLING_NOTIFICATION/AWS_BILLING_NOTIFICATION_6ce1d874-e995-40e2-99cd-
EXAMPLE11145",
      "entityValue": "AWS_ACCOUNT",
      "awsAccountId": "123456789012",
      "lastUpdatedTime": 1588356454.08
    }
  ]
}
```

Para obtener más información, consulte el [registro de eventos](#) en la Guía del usuario de AWS Health.

- Para API obtener más información, consulte [DescribeAffectedEntities](#) la Referencia de AWS CLI comandos.

## describe-event-details

En el siguiente ejemplo de código, se muestra cómo usar `describe-event-details`.

AWS CLI

Para incluir información sobre un evento AWS de Salud

En el siguiente `describe-event-details` ejemplo, se muestra información sobre el evento AWS Health especificado.

```
aws health describe-event-details \
```

```
--event-arns "arn:aws:health:us-east-1::event/EC2/AWS_EC2_OPERATIONAL_ISSUE/
AWS_EC2_OPERATIONAL_ISSUE_VKTXI_EXAMPLE111" \
--region us-east-1
```

Salida:

```
{
  "successfulSet": [
    {
      "event": {
        "arn": "arn:aws:health:us-east-1::event/EC2/
AWS_EC2_OPERATIONAL_ISSUE/AWS_EC2_OPERATIONAL_ISSUE_VKTXI_EXAMPLE111",
        "service": "EC2",
        "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
        "eventTypeCategory": "issue",
        "region": "us-east-1",
        "startTime": 1587462325.096,
        "endTime": 1587464204.774,
        "lastUpdatedTime": 1587464204.865,
        "statusCode": "closed"
      },
      "eventDescription": {
        "latestDescription": "[RESOLVED] Increased API Error Rates and
Latencies\n\n[02:45 AM PDT] We are investigating increased API error rates and
latencies in the US-EAST-1 Region.\n\n[03:16 AM PDT] Between 2:10 AM and 2:59 AM
PDT we experienced increased API error rates and latencies in the US-EAST-1 Region.
The issue has been resolved and the service is operating normally."
      }
    }
  ],
  "failedSet": []
}
```

Para obtener más información, consulte el [panel de detalles del evento](#) en la Guía del usuario de AWS Health.

- Para API obtener más información, consulte [DescribeEventDetails](#) la Referencia de AWS CLI comandos.

## describe-events

En el siguiente ejemplo de código, se muestra cómo usar describe-events.

## AWS CLI

### Ejemplo 1: Para enumerar eventos AWS de Salud

En el siguiente describe-events ejemplo, se enumeran los eventos AWS de Health recientes.

```
aws health describe-events \  
  --region us-east-1
```

Salida:

```
{  
  "events": [  
    {  
      "arn": "arn:aws:health:us-west-1::event/ECS/AWS_ECS_OPERATIONAL_ISSUE/  
AWS_ECS_OPERATIONAL_ISSUE_KWQPY_EXAMPLE111",  
      "service": "ECS",  
      "eventTypeCode": "AWS_ECS_OPERATIONAL_ISSUE",  
      "eventTypeCategory": "issue",  
      "region": "us-west-1",  
      "startTime": 1589077890.53,  
      "endTime": 1589086345.597,  
      "lastUpdatedTime": 1589086345.905,  
      "statusCode": "closed",  
      "eventScopeCode": "PUBLIC"  
    },  
    {  
      "arn": "arn:aws:health:global::event/BILLING/AWS_BILLING_NOTIFICATION/  
AWS_BILLING_NOTIFICATION_6ce1d874-e995-40e2-99cd-EXAMPLE1118b",  
      "service": "BILLING",  
      "eventTypeCode": "AWS_BILLING_NOTIFICATION",  
      "eventTypeCategory": "accountNotification",  
      "region": "global",  
      "startTime": 1588356000.0,  
      "lastUpdatedTime": 1588356524.358,  
      "statusCode": "open",  
      "eventScopeCode": "ACCOUNT_SPECIFIC"  
    },  
    {  
      "arn": "arn:aws:health:us-west-2::event/  
CLOUDFORMATION/AWS_CLOUDFORMATION_OPERATIONAL_ISSUE/  
AWS_CLOUDFORMATION_OPERATIONAL_ISSUE_OHTWY_EXAMPLE111",  
      "service": "CLOUDFORMATION",
```



```
    "eventTypeCode": "AWS_CLOUDFORMATION_OPERATIONAL_ISSUE",
    "eventTypeCategory": "issue",
    "region": "us-west-2",
    "startTime": 1588279630.761,
    "endTime": 1588284650.0,
    "lastUpdatedTime": 1588284691.941,
    "statusCode": "closed",
    "eventScopeCode": "PUBLIC"
  },
  {
    "arn": "arn:aws:health:ap-northeast-1::event/LAMBDA/
AWS_LAMBDA_OPERATIONAL_ISSUE/AWS_LAMBDA_OPERATIONAL_ISSUE_JZDND_EXAMPLE111",
    "service": "LAMBDA",
    "eventTypeCode": "AWS_LAMBDA_OPERATIONAL_ISSUE",
    "eventTypeCategory": "issue",
    "region": "ap-northeast-1",
    "startTime": 1587379534.08,
    "endTime": 1587391771.0,
    "lastUpdatedTime": 1587395689.316,
    "statusCode": "closed",
    "eventScopeCode": "PUBLIC"
  },
  {
    "arn": "arn:aws:health:us-east-1::event/EC2/AWS_EC2_OPERATIONAL_ISSUE/
AWS_EC2_OPERATIONAL_ISSUE_COBXJ_EXAMPLE111",
    "service": "EC2",
    "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
    "eventTypeCategory": "issue",
    "region": "us-east-1",
    "startTime": 1586473044.284,
    "endTime": 1586479706.091,
    "lastUpdatedTime": 1586479706.153,
    "statusCode": "closed",
    "eventScopeCode": "PUBLIC"
  },
  {
    "arn": "arn:aws:health:global::event/SECURITY/AWS_SECURITY_NOTIFICATION/
AWS_SECURITY_NOTIFICATION_42007387-8129-42da-8c88-EXAMPLE11139",
    "service": "SECURITY",
    "eventTypeCode": "AWS_SECURITY_NOTIFICATION",
    "eventTypeCategory": "accountNotification",
    "region": "global",
    "startTime": 1585674000.0,
    "lastUpdatedTime": 1585674004.132,
```

```
    "statusCode": "open",
    "eventScopeCode": "PUBLIC"
  },
  {
    "arn": "arn:aws:health:global::event/CLOUDFRONT/
AWS_CLOUDFRONT_OPERATIONAL_ISSUE/AWS_CLOUDFRONT_OPERATIONAL_ISSUE_FRQXG_EXAMPLE111",
    "service": "CLOUDFRONT",
    "eventTypeCode": "AWS_CLOUDFRONT_OPERATIONAL_ISSUE",
    "eventTypeCategory": "issue",
    "region": "global",
    "startTime": 1585610898.589,
    "endTime": 1585617671.0,
    "lastUpdatedTime": 1585620638.869,
    "statusCode": "closed",
    "eventScopeCode": "PUBLIC"
  },
  {
    "arn": "arn:aws:health:us-east-1::event/SES/AWS_SES_OPERATIONAL_ISSUE/
AWS_SES_OPERATIONAL_ISSUE_URNDF_EXAMPLE111",
    "service": "SES",
    "eventTypeCode": "AWS_SES_OPERATIONAL_ISSUE",
    "eventTypeCategory": "issue",
    "region": "us-east-1",
    "startTime": 1585342008.46,
    "endTime": 1585344017.0,
    "lastUpdatedTime": 1585344355.989,
    "statusCode": "closed",
    "eventScopeCode": "PUBLIC"
  },
  {
    "arn": "arn:aws:health:global::event/IAM/
AWS_IAM_OPERATIONAL_NOTIFICATION/
AWS_IAM_OPERATIONAL_NOTIFICATION_b6771c34-6ecd-4aea-9d3e-EXAMPLE1117e",
    "service": "IAM",
    "eventTypeCode": "AWS_IAM_OPERATIONAL_NOTIFICATION",
    "eventTypeCategory": "accountNotification",
    "region": "global",
    "startTime": 1584978300.0,
    "lastUpdatedTime": 1584978553.572,
    "statusCode": "open",
    "eventScopeCode": "ACCOUNT_SPECIFIC"
  },
  {
```

```

        "arn": "arn:aws:health:ap-southeast-2::event/EC2/
AWS_EC2_OPERATIONAL_ISSUE/AWS_EC2_OPERATIONAL_ISSUE_HNGHE_EXAMPLE111",
        "service": "EC2",
        "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
        "eventTypeCategory": "issue",
        "region": "ap-southeast-2",
        "startTime": 1583881487.483,
        "endTime": 1583885056.785,
        "lastUpdatedTime": 1583885057.052,
        "statusCode": "closed",
        "eventScopeCode": "PUBLIC"
    }
]
}

```

Para obtener más información, consulte [Cómo empezar a usar el AWS Personal Health Dashboard](#) en la Guía del usuario de AWS Salud.

Ejemplo 2: Para enumerar los eventos de AWS Salud por servicio y código de estado del evento

En el siguiente `describe-events` ejemplo, se enumeran los eventos de AWS Health de Amazon Elastic Compute Cloud (AmazonEC2) en los que el estado del evento es cerrado.

```

aws health describe-events \
  --filter "services=EC2,eventStatusCodes=closed"

```

Salida:

```

{
  "events": [
    {
      "arn": "arn:aws:health:us-east-1::event/EC2/AWS_EC2_OPERATIONAL_ISSUE/
AWS_EC2_OPERATIONAL_ISSUE_VKTXI_EXAMPLE111",
      "service": "EC2",
      "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
      "eventTypeCategory": "issue",
      "region": "us-east-1",
      "startTime": 1587462325.096,
      "endTime": 1587464204.774,
      "lastUpdatedTime": 1587464204.865,
      "statusCode": "closed",
      "eventScopeCode": "PUBLIC"
    }
  ]
}

```

```
    },
    {
      "arn": "arn:aws:health:us-east-1::event/EC2/AWS_EC2_OPERATIONAL_ISSUE/
AWS_EC2_OPERATIONAL_ISSUE_COBJX_EXAMPLE111",
      "service": "EC2",
      "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
      "eventTypeCategory": "issue",
      "region": "us-east-1",
      "startTime": 1586473044.284,
      "endTime": 1586479706.091,
      "lastUpdatedTime": 1586479706.153,
      "statusCode": "closed",
      "eventScopeCode": "PUBLIC"
    },
    {
      "arn": "arn:aws:health:ap-southeast-2::event/EC2/
AWS_EC2_OPERATIONAL_ISSUE/AWS_EC2_OPERATIONAL_ISSUE_HNGHE_EXAMPLE111",
      "service": "EC2",
      "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
      "eventTypeCategory": "issue",
      "region": "ap-southeast-2",
      "startTime": 1583881487.483,
      "endTime": 1583885056.785,
      "lastUpdatedTime": 1583885057.052,
      "statusCode": "closed",
      "eventScopeCode": "PUBLIC"
    }
  ]
}
```

Para obtener más información, consulte [Cómo empezar a usar el AWS Personal Health Dashboard](#) en la Guía del usuario de AWS Salud.

- Para API obtener más información, consulte [DescribeEvents](#) la Referencia de AWS CLI comandos.

## HealthImaging ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with HealthImaging.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

Acciones

### **copy-image-set**

En el siguiente ejemplo de código, se muestra cómo usar `copy-image-set`.

AWS CLI

Ejemplo 1: copia de un conjunto de imágenes sin un destino.

En el siguiente ejemplo de código `copy-image-set` se hace una copia duplicada de un conjunto de imágenes sin un destino.

```
aws medical-imaging copy-image-set \  
  --datastore-id 12345678901234567890123456789012 \  
  --source-image-set-id ea92b0d8838c72a3f25d00d13616f87e \  
  --copy-image-set-information '{"sourceImageSet": {"latestVersionId": "1" } }'
```

Salida:

```
{  
  "destinationImageSetProperties": {  
    "latestVersionId": "2",  
    "imageSetWorkflowStatus": "COPYING",  
    "updatedAt": 1680042357.432,  
    "imageSetId": "b9a06fef182a5f992842f77f8e0868e5",  
    "imageSetState": "LOCKED",  
    "createdAt": 1680042357.432  
  },  
  "sourceImageSetProperties": {  
    "latestVersionId": "1",
```

```

    "imageSetWorkflowStatus": "COPYING_WITH_READ_ONLY_ACCESS",
    "updatedAt": 1680042357.432,
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "LOCKED",
    "createdAt": 1680027126.436
  },
  "datastoreId": "12345678901234567890123456789012"
}

```

Ejemplo 2: copia de un conjunto de imágenes con un destino.

En el siguiente ejemplo de código `copy-image-set` se hace una copia duplicada de un conjunto de imágenes con un destino.

```

aws medical-imaging copy-image-set \
  --datastore-id 12345678901234567890123456789012 \
  --source-image-set-id ea92b0d8838c72a3f25d00d13616f87e \
  --copy-image-set-information '{"sourceImageSet": {"latestVersionId": "1" },
  "destinationImageSet": { "imageSetId": "b9a06fef182a5f992842f77f8e0868e5",
  "latestVersionId": "1"} }'

```

Salida:

```

{
  "destinationImageSetProperties": {
    "latestVersionId": "2",
    "imageSetWorkflowStatus": "COPYING",
    "updatedAt": 1680042505.135,
    "imageSetId": "b9a06fef182a5f992842f77f8e0868e5",
    "imageSetState": "LOCKED",
    "createdAt": 1680042357.432
  },
  "sourceImageSetProperties": {
    "latestVersionId": "1",
    "imageSetWorkflowStatus": "COPYING_WITH_READ_ONLY_ACCESS",
    "updatedAt": 1680042505.135,
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "LOCKED",
    "createdAt": 1680027126.436
  },
  "datastoreId": "12345678901234567890123456789012"
}

```

Para obtener más información, consulta Cómo [copiar un conjunto de imágenes](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [CopyImageSet](#) la Referencia de AWS CLI comandos.

## create-datastore

En el siguiente ejemplo de código, se muestra cómo usar create-datastore.

### AWS CLI

Creación de un almacén de datos

En el siguiente ejemplo de código create-datastore se crea un almacén de datos con el nombre my-datastore.

```
aws medical-imaging create-datastore \  
  --datastore-name "my-datastore"
```

Salida:

```
{  
  "datastoreId": "12345678901234567890123456789012",  
  "datastoreStatus": "CREATING"  
}
```

Para obtener más información, consulte [Creación de un banco de datos](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [CreateDatastore](#) la Referencia de AWS CLI comandos.

## delete-datastore

En el siguiente ejemplo de código, se muestra cómo usar delete-datastore.

### AWS CLI

Eliminación de un almacén de datos

En el siguiente ejemplo de código delete-datastore se elimina un almacén de datos.

```
aws medical-imaging delete-datastore \  
  --datastore-id "12345678901234567890123456789012"
```

Salida:

```
{  
  "datastoreId": "12345678901234567890123456789012",  
  "datastoreStatus": "DELETING"  
}
```

Para obtener más información, consulte [Eliminar un banco de datos](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [DeleteDatastore](#) la Referencia de AWS CLI comandos.

## delete-image-set

En el siguiente ejemplo de código, se muestra cómo usar `delete-image-set`.

AWS CLI

Eliminación de un conjunto de imágenes

En el siguiente ejemplo de código `delete-image-set` se elimina un conjunto de imágenes.

```
aws medical-imaging delete-image-set \  
  --datastore-id 12345678901234567890123456789012 \  
  --image-set-id ea92b0d8838c72a3f25d00d13616f87e
```

Salida:

```
{  
  "imageSetWorkflowStatus": "DELETING",  
  "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",  
  "imageSetState": "LOCKED",  
  "datastoreId": "12345678901234567890123456789012"  
}
```

Para obtener más información, consulte [Eliminar un conjunto de imágenes](#) en la Guía para AWS HealthImaging desarrolladores.



- Para API obtener más información, consulte [DeleteImageSet](#) la Referencia de AWS CLI comandos.

## get-datastore

En el siguiente ejemplo de código, se muestra cómo usar `get-datastore`.

### AWS CLI

Obtención de las propiedades de un almacén de datos

En el siguiente ejemplo de código `get-datastore` se obtienen las propiedades de un almacén de datos.

```
aws medical-imaging get-datastore \
  --datastore-id 12345678901234567890123456789012
```

Salida:

```
{
  "datastoreProperties": {
    "datastoreId": "12345678901234567890123456789012",
    "datastoreName": "TestDatastore123",
    "datastoreStatus": "ACTIVE",
    "datastoreArn": "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012",
    "createdAt": "2022-11-15T23:33:09.643000+00:00",
    "updatedAt": "2022-11-15T23:33:09.643000+00:00"
  }
}
```

Para obtener más información, consulte [Obtener las propiedades del almacén de datos](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [GetDatastore](#) la Referencia de AWS CLI comandos.

## get-dicom-import-job

En el siguiente ejemplo de código, se muestra cómo usar `get-dicom-import-job`.

## AWS CLI

### Obtención de las propiedades de un trabajo de importación DICOM

En el siguiente ejemplo de código `get-dicom-import-job` se obtienen las propiedades de un trabajo de importación DICOM.

```
aws medical-imaging get-dicom-import-job \  
  --datastore-id "12345678901234567890123456789012" \  
  --job-id "09876543210987654321098765432109"
```

Salida:

```
{  
  "jobProperties": {  
    "jobId": "09876543210987654321098765432109",  
    "jobName": "my-job",  
    "jobStatus": "COMPLETED",  
    "datastoreId": "12345678901234567890123456789012",  
    "dataAccessRoleArn": "arn:aws:iam::123456789012:role/  
ImportJobDataAccessRole",  
    "endedAt": "2022-08-12T11:29:42.285000+00:00",  
    "submittedAt": "2022-08-12T11:28:11.152000+00:00",  
    "inputS3Uri": "s3://medical-imaging-dicom-input/dicom_input/",  
    "outputS3Uri": "s3://medical-imaging-output/  
job_output/12345678901234567890123456789012-  
DicomImport-09876543210987654321098765432109/"  
  }  
}
```

Para obtener más información, consulte [Obtener las propiedades de los trabajos de importación](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [GetDICOMImportJob](#) in AWS CLI Command Reference.

### **get-image-frame**

En el siguiente ejemplo de código, se muestra cómo usar `get-image-frame`.

## AWS CLI

Obtención de datos de píxeles de un conjunto de imágenes

En el siguiente ejemplo de código `get-image-frame` se obtiene un marco de una imagen.

```
aws medical-imaging get-image-frame \  
  --datastore-id "12345678901234567890123456789012" \  
  --image-set-id "98765412345612345678907890789012" \  
  --image-frame-information imageFrameId=3abf5d5d7ae72f80a0ec81b2c0de3ef4 \  
  imageframe.jpg
```

Nota: Este ejemplo de código no incluye la salida porque la `GetImageFrame` acción devuelve un flujo de datos de píxeles al archivo `imageframe.jpg`. Para obtener información sobre la decodificación y la visualización de marcos de imágenes, consulte las bibliotecas de decodificación. HTJ2K

Para obtener más información, consulte [Obtener datos de píxeles de conjuntos de imágenes](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [GetImageFrame](#) la Referencia de AWS CLI comandos.

## `get-image-set-metadata`

En el siguiente ejemplo de código, se muestra cómo usar `get-image-set-metadata`.

## AWS CLI

Ejemplo 1: obtención de los metadatos de un conjunto de imágenes sin versión

En el siguiente ejemplo de código `get-image-set-metadata` se obtienen los metadatos de un conjunto de imágenes sin especificar una versión.

Nota: El parámetro `outfile` es obligatorio

```
aws medical-imaging get-image-set-metadata \  
  --datastore-id 12345678901234567890123456789012 \  
  --image-set-id ea92b0d8838c72a3f25d00d13616f87e \  
  studymetadata.json.gz
```

Los metadatos devueltos se comprimen con gzip y se almacenan en el archivo `studymetadata.json.gz`. Para ver el contenido del JSON objeto devuelto, primero debe descomprimirlo.

Salida:

```
{
  "contentType": "application/json",
  "contentEncoding": "gzip"
}
```

Ejemplo 2: obtención de los metadatos de un conjunto de imágenes con versión

En el siguiente ejemplo de código `get-image-set-metadata` se obtienen los metadatos de un conjunto de imágenes con una versión especificada.

Nota: El parámetro `outfile` es obligatorio

```
aws medical-imaging get-image-set-metadata \
  --datastore-id 12345678901234567890123456789012 \
  --image-set-id ea92b0d8838c72a3f25d00d13616f87e \
  --version-id 1 \
  studymetadata.json.gz
```

Los metadatos devueltos se comprimen con gzip y se almacenan en el archivo `studymetadata.json.gz`. Para ver el contenido del JSON objeto devuelto, primero debe descomprimirlo.

Salida:

```
{
  "contentType": "application/json",
  "contentEncoding": "gzip"
}
```

Para obtener más información, consulta [Cómo obtener los metadatos del conjunto de imágenes](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [GetImageSetMetadata](#) la Referencia de AWS CLI comandos.

## get-image-set

En el siguiente ejemplo de código, se muestra cómo usar `get-image-set`.

### AWS CLI

Obtención de las propiedades de un conjunto de imágenes

En el siguiente ejemplo de código `get-image-set` se obtienen las propiedades de un conjunto de imágenes.

```
aws medical-imaging get-image-set \  
  --datastore-id 12345678901234567890123456789012 \  
  --image-set-id 18f88ac7870584f58d56256646b4d92b \  
  --version-id 1
```

Salida:

```
{  
  "versionId": "1",  
  "imageSetWorkflowStatus": "COPIED",  
  "updatedAt": 1680027253.471,  
  "imageSetId": "18f88ac7870584f58d56256646b4d92b",  
  "imageSetState": "ACTIVE",  
  "createdAt": 1679592510.753,  
  "datastoreId": "12345678901234567890123456789012"  
}
```

Para obtener más información, consulte [Obtener las propiedades de un conjunto de imágenes](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [GetImageSet](#) la Referencia de AWS CLI comandos.

## list-datastores

En el siguiente ejemplo de código, se muestra cómo usar `list-datastores`.

### AWS CLI

Enumeración de almacenes de datos

En el siguiente ejemplo de código `list-datastores` se enumeran los almacenes de datos disponibles.

```
aws medical-imaging list-datastores
```

Salida:

```
{
  "datastoreSummaries": [
    {
      "datastoreId": "12345678901234567890123456789012",
      "datastoreName": "TestDatastore123",
      "datastoreStatus": "ACTIVE",
      "datastoreArn": "arn:aws:medical-imaging:us-east-1:123456789012:datastore/12345678901234567890123456789012",
      "createdAt": "2022-11-15T23:33:09.643000+00:00",
      "updatedAt": "2022-11-15T23:33:09.643000+00:00"
    }
  ]
}
```

Para obtener más información, consulte la [lista de almacenes de datos](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [ListDatastores](#) la Referencia de AWS CLI comandos.

## **list-dicom-import-jobs**

En el siguiente ejemplo de código, se muestra cómo usar `list-dicom-import-jobs`.

AWS CLI

Enumeración de los trabajos de importación DICOM

En el siguiente ejemplo de código `list-dicom-import-jobs` se enumeran los trabajos de importación DICOM.

```
aws medical-imaging list-dicom-import-jobs \
  --datastore-id "12345678901234567890123456789012"
```

Salida:

```
{
  "jobSummaries": [
    {
      "jobId": "09876543210987654321098765432109",
      "jobName": "my-job",
      "jobStatus": "COMPLETED",
      "datastoreId": "12345678901234567890123456789012",
      "dataAccessRoleArn": "arn:aws:iam::123456789012:role/
ImportJobDataAccessRole",
      "endedAt": "2022-08-12T11:21:56.504000+00:00",
      "submittedAt": "2022-08-12T11:20:21.734000+00:00"
    }
  ]
}
```

Para obtener más información, consulte [Listar los trabajos de importación](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [ListDICOMImport Jobs](#) en AWS CLI Command Reference.

## list-image-set-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-image-set-versions`.

### AWS CLI

Enumeración de las versiones de un conjunto de imágenes

En el siguiente ejemplo de código `list-image-set-versions` se enumera el historial de versiones de un conjunto de imágenes.

```
aws medical-imaging list-image-set-versions \
  --datastore-id 12345678901234567890123456789012 \
  --image-set-id ea92b0d8838c72a3f25d00d13616f87e
```

Salida:

```
{
  "imageSetPropertiesList": [
    {
      "ImageSetWorkflowStatus": "UPDATED",
```

```

    "versionId": "4",
    "updatedAt": 1680029436.304,
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "ACTIVE",
    "createdAt": 1680027126.436
  },
  {
    "ImageSetWorkflowStatus": "UPDATED",
    "versionId": "3",
    "updatedAt": 1680029163.325,
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "ACTIVE",
    "createdAt": 1680027126.436
  },
  {
    "ImageSetWorkflowStatus": "COPY_FAILED",
    "versionId": "2",
    "updatedAt": 1680027455.944,
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "ACTIVE",
    "message": "INVALID_REQUEST: Series of SourceImageSet and
DestinationImageSet don't match.",
    "createdAt": 1680027126.436
  },
  {
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "ACTIVE",
    "versionId": "1",
    "ImageSetWorkflowStatus": "COPIED",
    "createdAt": 1680027126.436
  }
]
}

```

Para obtener más información, consulta la sección sobre [la lista de versiones de conjuntos de imágenes](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [ListImageSetVersions](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.



## AWS CLI

Ejemplo 1: enumeración de las etiquetas de recursos de un almacén de datos

En el siguiente ejemplo de código `list-tags-for-resource` se enumeran las etiquetas de un almacén de datos.

```
aws medical-imaging list-tags-for-resource \  
  --resource-arn "arn:aws:medical-imaging:us-  
east-1:123456789012:datastore/12345678901234567890123456789012"
```

Salida:

```
{  
  "tags":{  
    "Deployment":"Development"  
  }  
}
```

Ejemplo 2: enumeración de las etiquetas de recursos de un conjunto de imágenes

En el siguiente ejemplo de código `list-tags-for-resource` se enumeran las etiquetas de un conjunto de imágenes.

```
aws medical-imaging list-tags-for-resource \  
  --resource-arn "arn:aws:medical-imaging:us-  
east-1:123456789012:datastore/12345678901234567890123456789012/  
imageset/18f88ac7870584f58d56256646b4d92b"
```

Salida:

```
{  
  "tags":{  
    "Deployment":"Development"  
  }  
}
```

Para obtener más información, consulte [Etiquetar recursos AWS HealthImaging](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## search-image-sets

En el siguiente ejemplo de código, se muestra cómo usar `search-image-sets`.

### AWS CLI

Ejemplo 1: Para buscar conjuntos de imágenes con un EQUAL operador

El siguiente ejemplo `search-image-sets` de código utiliza el EQUAL operador para buscar conjuntos de imágenes en función de un valor específico.

```
aws medical-imaging search-image-sets \  
  --datastore-id 12345678901234567890123456789012 \  
  --search-criteria file://search-criteria.json
```

Contenido de `search-criteria.json`

```
{  
  "filters": [{  
    "values": [{"DICOMPatientId" : "SUBJECT08701"}],  
    "operator": "EQUAL"  
  }]  
}
```

Salida:

```
{  
  "imageSetsMetadataSummaries": [{  
    "imageSetId": "09876543210987654321098765432109",  
    "createdAt": "2022-12-06T21:40:59.429000+00:00",  
    "version": 1,  
    "DICOMTags": {  
      "DICOMStudyId": "2011201407",  
      "DICOMStudyDate": "19991122",  
      "DICOMPatientSex": "F",  
      "DICOMStudyInstanceUID": "1.2.840.99999999.84710745.943275268089",  
      "DICOMPatientBirthDate": "19201120",  
      "DICOMStudyDescription": "UNKNOWN",  
      "DICOMPatientId": "SUBJECT08701",  
      "DICOMPatientName": "Melissa844 Huel628",  
      "DICOMNumberOfStudyRelatedInstances": 1,  
      "DICOMStudyTime": "140728",  
      "DICOMNumberOfStudyRelatedSeries": 1  
    }  
  }]  
}
```

```

    },
    "updatedAt": "2022-12-06T21:40:59.429000+00:00"
  ]}
}

```

Ejemplo 2: Para buscar conjuntos de imágenes con un BETWEEN operador, utilice DICOMStudyDate y DICOMStudyTime

El siguiente ejemplo de `search-image-sets` código busca conjuntos de imágenes con DICOM estudios generados entre el 1 de enero de 1990 (12:00 a. m.) y el 1 de enero de 2023 (12:00 a. m.).

Nota: DICOMStudyTime es opcional. Si no está presente, el valor de hora de las fechas indicado para el filtrado es a las 00:00 h (inicio del día).

```

aws medical-imaging search-image-sets \
  --datastore-id 12345678901234567890123456789012 \
  --search-criteria file://search-criteria.json

```

Contenido de `search-criteria.json`

```

{
  "filters": [{
    "values": [{
      "DICOMStudyDateAndTime": {
        "DICOMStudyDate": "19900101",
        "DICOMStudyTime": "000000"
      }
    },
    {
      "DICOMStudyDateAndTime": {
        "DICOMStudyDate": "20230101",
        "DICOMStudyTime": "000000"
      }
    }
  ]},
  "operator": "BETWEEN"
}]
}

```

Salida:

```
{
```

```

"imageSetsMetadataSummaries": [{
  "imageSetId": "09876543210987654321098765432109",
  "createdAt": "2022-12-06T21:40:59.429000+00:00",
  "version": 1,
  "DICOMTags": {
    "DICOMStudyId": "2011201407",
    "DICOMStudyDate": "19991122",
    "DICOMPatientSex": "F",
    "DICOMStudyInstanceUID": "1.2.840.99999999.84710745.943275268089",
    "DICOMPatientBirthDate": "19201120",
    "DICOMStudyDescription": "UNKNOWN",
    "DICOMPatientId": "SUBJECT08701",
    "DICOMPatientName": "Melissa844 Huel628",
    "DICOMNumberOfStudyRelatedInstances": 1,
    "DICOMStudyTime": "140728",
    "DICOMNumberOfStudyRelatedSeries": 1
  },
  "updatedAt": "2022-12-06T21:40:59.429000+00:00"
}]
}

```

Ejemplo 3: Para buscar conjuntos de imágenes con un BETWEEN operador utilizando createdAt (previamente se conservaban los estudios de tiempo)

El siguiente ejemplo de search-image-sets código busca conjuntos de imágenes cuyos DICOM estudios persistan HealthImaging entre los rangos de tiempo de la zona horariaUTC.

Nota: Introdúzcalo createdAt en un formato de ejemplo («1985-04-12T 23:20:50.52 Z»).

```

aws medical-imaging search-image-sets \
  --datastore-id 12345678901234567890123456789012 \
  --search-criteria file://search-criteria.json

```

Contenido de search-criteria.json

```

{
  "filters": [{
    "values": [{
      "createdAt": "1985-04-12T23:20:50.52Z"
    },
    {
      "createdAt": "2022-04-12T23:20:50.52Z"
    }
  ]
}

```

```

    ]],
    "operator": "BETWEEN"
  ]]
}

```

Salida:

```

{
  "imageSetsMetadataSummaries": [{
    "imageSetId": "09876543210987654321098765432109",
    "createdAt": "2022-12-06T21:40:59.429000+00:00",
    "version": 1,
    "DICOMTags": {
      "DICOMStudyId": "2011201407",
      "DICOMStudyDate": "19991122",
      "DICOMPatientSex": "F",
      "DICOMStudyInstanceUID": "1.2.840.99999999.84710745.943275268089",
      "DICOMPatientBirthDate": "19201120",
      "DICOMStudyDescription": "UNKNOWN",
      "DICOMPatientId": "SUBJECT08701",
      "DICOMPatientName": "Melissa844 Huel628",
      "DICOMNumberOfStudyRelatedInstances": 1,
      "DICOMStudyTime": "140728",
      "DICOMNumberOfStudyRelatedSeries": 1
    },
    "lastUpdatedAt": "2022-12-06T21:40:59.429000+00:00"
  ]}
}

```

Ejemplo 4: Para buscar conjuntos de imágenes con un EQUAL operador activado y activado DICOMSeriesInstanceUID y ordenar las respuestas por orden en el campo BETWEEN updatedAt ASC updatedAt

En el siguiente ejemplo de `search-image-sets` código se buscan conjuntos de imágenes con un EQUAL operador activado DICOMSeriesInstanceUID y BETWEEN activado updatedAt y se ordenan las ASC respuestas por updatedAt campo.

Nota: Escríbalo updatedAt en un formato de ejemplo («1985-04-12T 23:20:50 .52 Z»).

```

aws medical-imaging search-image-sets \
  --datastore-id 12345678901234567890123456789012 \
  --search-criteria file://search-criteria.json

```

## Contenido de search-criteria.json

```
{
  "filters": [{
    "values": [{
      "updatedAt": "2024-03-11T15:00:05.074000-07:00"
    }, {
      "updatedAt": "2024-03-11T16:00:05.074000-07:00"
    }],
    "operator": "BETWEEN"
  }, {
    "values": [{
      "DICOMSeriesInstanceUID": "1.2.840.99999999.84710745.943275268089"
    }],
    "operator": "EQUAL"
  }],
  "sort": {
    "sortField": "updatedAt",
    "sortOrder": "ASC"
  }
}
```

## Salida:

```
{
  "imageSetsMetadataSummaries": [{
    "imageSetId": "09876543210987654321098765432109",
    "createdAt": "2022-12-06T21:40:59.429000+00:00",
    "version": 1,
    "DICOMTags": {
      "DICOMStudyId": "2011201407",
      "DICOMStudyDate": "19991122",
      "DICOMPatientSex": "F",
      "DICOMStudyInstanceUID": "1.2.840.99999999.84710745.943275268089",
      "DICOMPatientBirthDate": "19201120",
      "DICOMStudyDescription": "UNKNOWN",
      "DICOMPatientId": "SUBJECT08701",
      "DICOMPatientName": "Melissa844 Huel628",
      "DICOMNumberOfStudyRelatedInstances": 1,
      "DICOMStudyTime": "140728",
      "DICOMNumberOfStudyRelatedSeries": 1
    },
    "lastUpdatedAt": "2022-12-06T21:40:59.429000+00:00"
  }]
```

```
}]
}
```

Para obtener más información, consulte [Búsqueda de conjuntos de imágenes en la Guía para desarrolladores.AWS HealthImaging](#)

- Para API obtener más información, consulte [SearchImageSets](#) la Referencia de AWS CLI comandos.

## start-dicom-import-job

En el siguiente ejemplo de código, se muestra cómo usar `start-dicom-import-job`.

### AWS CLI

Inicio de un trabajo de importación DICOM

En el siguiente ejemplo de código `start-dicom-import-job` se inicia un trabajo de importación DICOM.

```
aws medical-imaging start-dicom-import-job \
  --job-name "my-job" \
  --datastore-id "12345678901234567890123456789012" \
  --input-s3-uri "s3://medical-imaging-dicom-input/dicom_input/" \
  --output-s3-uri "s3://medical-imaging-output/job_output/" \
  --data-access-role-arn "arn:aws:iam::123456789012:role/ImportJobDataAccessRole"
```

Salida:

```
{
  "datastoreId": "12345678901234567890123456789012",
  "jobId": "09876543210987654321098765432109",
  "jobStatus": "SUBMITTED",
  "submittedAt": "2022-08-12T11:28:11.152000+00:00"
}
```

Para obtener más información, consulte Cómo [iniciar un trabajo de importación](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [StartDICOMImport Job](#) in AWS CLI Command Reference.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Ejemplo 1: etiquetado de un almacén de datos

En los siguientes ejemplos de código `tag-resource` se etiqueta un almacén de datos.

```
aws medical-imaging tag-resource \  
  --resource-arn "arn:aws:medical-imaging:us-  
east-1:123456789012:datastore/12345678901234567890123456789012" \  
  --tags '{"Deployment":"Development"}'
```

Este comando no genera ninguna salida.

Ejemplo 2: etiquetado de un conjunto de imágenes

En los siguientes ejemplos de código `tag-resource` se etiqueta un conjunto de imágenes.

```
aws medical-imaging tag-resource \  
  --resource-arn "arn:aws:medical-imaging:us-  
east-1:123456789012:datastore/12345678901234567890123456789012/  
imageset/18f88ac7870584f58d56256646b4d92b" \  
  --tags '{"Deployment":"Development"}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar recursos AWS HealthImaging](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Ejemplo 1: eliminación de las etiquetas de un almacén de datos



En el siguiente ejemplo de código `untag-resource` se eliminan las etiquetas de un almacén de datos.

```
aws medical-imaging untag-resource \  
  --resource-arn "arn:aws:medical-imaging:us-  
east-1:123456789012:datastore/12345678901234567890123456789012" \  
  --tag-keys ["Deployment"]
```

Este comando no genera ninguna salida.

Ejemplo 2: eliminación de las etiquetas de un conjunto de imágenes

En el siguiente ejemplo de código `untag-resource` se eliminan las etiquetas de un conjunto de imágenes.

```
aws medical-imaging untag-resource \  
  --resource-arn "arn:aws:medical-imaging:us-  
east-1:123456789012:datastore/12345678901234567890123456789012/  
imageset/18f88ac7870584f58d56256646b4d92b" \  
  --tag-keys ["Deployment"]
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar recursos AWS HealthImaging](#) en la Guía para AWS HealthImaging desarrolladores.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-image-set-metadata

En el siguiente ejemplo de código, se muestra cómo usar `update-image-set-metadata`.

### AWS CLI

Para insertar o actualizar un atributo en los metadatos del conjunto de imágenes

El siguiente ejemplo `update-image-set-metadata` de código inserta o actualiza un atributo en los metadatos del conjunto de imágenes.

```
aws medical-imaging update-image-set-metadata \  
  --datastore-id 12345678901234567890123456789012 \  
  --tag-key Deployment
```

```
--image-set-id ea92b0d8838c72a3f25d00d13616f87e \  
--latest-version-id 1 \  
--update-image-set-metadata-updates file://metadata-updates.json
```

Contenido de metadata-updates.json

```
{  
  "DICOMUpdates": {  
    "updatableAttributes":  
    "eyJTY2h1bWFWZXJzaW9uIjoxLjEsIlBhdGllbnQiOnsiRElDT00iOnsiUGF0aWVudE5hbWUiOiJNWF5NWCJ9fX0=" }  
  }  
}
```

Nota: updatableAttributes es una JSON cadena codificada en Base64. Esta es la cadena no codificadaJSON.

```
{» SchemaVersion «:1.1, "Paciente»: {» DICOM «: {» PatientName «:"MX^MX"}}}
```

Salida:

```
{  
  "latestVersionId": "2",  
  "imageSetWorkflowStatus": "UPDATING",  
  "updatedAt": 1680042257.908,  
  "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",  
  "imageSetState": "LOCKED",  
  "createdAt": 1680027126.436,  
  "datastoreId": "12345678901234567890123456789012"  
}
```

Para eliminar un atributo de los metadatos del conjunto de imágenes

El siguiente ejemplo update-image-set-metadata de código elimina un atributo de los metadatos del conjunto de imágenes.

```
aws medical-imaging update-image-set-metadata \  
--datastore-id 12345678901234567890123456789012 \  
--image-set-id ea92b0d8838c72a3f25d00d13616f87e \  
--latest-version-id 1 \  
--update-image-set-metadata-updates file://metadata-updates.json
```

Contenido de metadata-updates.json

```
{
  "DICOMUpdates": {
    "removableAttributes":
    "e1NjaGVtYVZlcnNpb246MS4xLFN0dWR50ntESUNPTTp7U3R1ZH1EZXNjcmlwdGlvbjpgDSEVTVH19fQo="
  }
}
```

Nota: `removableAttributes` es una JSON cadena codificada en Base64. Esta es la cadena no codificadaJSON. La clave y el valor deben coincidir con el atributo que se va a eliminar.

```
{» SchemaVersion «:1.1, "Estudiar»: {» DICOM «:» StudyDescription CHEST «}}
```

Salida:

```
{
  "latestVersionId": "2",
  "imageSetWorkflowStatus": "UPDATING",
  "updatedAt": 1680042257.908,
  "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
  "imageSetState": "LOCKED",
  "createdAt": 1680027126.436,
  "datastoreId": "12345678901234567890123456789012"
}
```

Para eliminar una instancia de los metadatos del conjunto de imágenes

En el siguiente ejemplo `update-image-set-metadata` de código, se elimina una instancia de los metadatos del conjunto de imágenes.

```
aws medical-imaging update-image-set-metadata \
  --datastore-id 12345678901234567890123456789012 \
  --image-set-id ea92b0d8838c72a3f25d00d13616f87e \
  --latest-version-id 1 \
  --update-image-set-metadata-updates file://metadata-updates.json
```

Contenido de `metadata-updates.json`

```
{
  "DICOMUpdates": {
    "removableAttributes":
    "eezEuMS4xLjEuMS4xLjEyMzQ1LjEyMzQ1Njc40TAXMi4xMjMuMTIzNDU2Nzg5MDEyMzQuMTp7SW5zdGFuY2Vz0nsxL"
```

```
}  
}
```

Nota: `removableAttributes` es una JSON cadena codificada en Base64. Esta es la cadena no codificadaJSON.

```
{"1.1.1.1.1.12345.123456789012.123.123456789012341": {"Instancias»:  
{"1.1.1.1.1.1.12345.123456789012.123.12345678901234.1": {}}}}
```

Salida:

```
{  
  "latestVersionId": "2",  
  "imageSetWorkflowStatus": "UPDATING",  
  "updatedAt": 1680042257.908,  
  "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",  
  "imageSetState": "LOCKED",  
  "createdAt": 1680027126.436,  
  "datastoreId": "12345678901234567890123456789012"  
}
```

Para obtener más información, consulte [Actualización de los metadatos del conjunto de imágenes en la Guía AWS HealthImaging para desarrolladores](#).

- Para API obtener más información, consulte [UpdateImageSetMetadata](#) la Referencia de AWS CLI comandos.

## HealthLake ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with HealthLake.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **create-fhir-datastore**

En el siguiente ejemplo de código, se muestra cómo usar `create-fhir-datastore`.

#### AWS CLI

Para crear un almacén FHIR de datos.

El siguiente `create-fhir-datastore` ejemplo muestra cómo crear un nuevo almacén de datos en Amazon HealthLake.

```
aws healthlake create-fhir-datastore \  
  --region us-east-1 \  
  --datastore-type-version R4 \  
  --datastore-type-version R4 \  
  --datastore-name "FhirTestDatastore"
```

Salida:

```
{  
  "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/  
(Datastore ID)/r4/",  
  "DatastoreArn": "arn:aws:healthlake:us-east-1:(AWS Account ID):datastore/  
(Datastore ID)",  
  "DatastoreStatus": "CREATING",  
  "DatastoreId": "(Datastore ID)"  
}
```

Para obtener más información, consulte [Creación y supervisión de un almacén de FHIR datos](#) en la Guía para HealthLake desarrolladores de Amazon.

- Para API obtener más información, consulte [CreateFhirDatastore](#) la Referencia de AWS CLI comandos.

### **delete-fhir-datastore**

En el siguiente ejemplo de código, se muestra cómo usar `delete-fhir-datastore`.

#### AWS CLI

Para eliminar un almacén FHIR de datos

El siguiente `delete-fhir-datastore` ejemplo muestra cómo eliminar un almacén de datos y todo su contenido en Amazon HealthLake.

```
aws healthlake delete-fhir-datastore \  
  --datastore-id (Data Store ID) \  
  --region us-east-1
```

Salida:

```
{  
  "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/  
(Datastore ID)/r4/",  
  "DatastoreArn": "arn:aws:healthlake:us-east-1:(AWS Account ID):datastore/  
(Datastore ID)",  
  "DatastoreStatus": "DELETING",  
  "DatastoreId": "(Datastore ID)"  
}
```

Para obtener más información, consulte [Creación y supervisión de un almacén de FHIR datos < https://docs.aws.amazon.com/healthlake/latest/devguide/working-with-FHIR-healthlake.html >](https://docs.aws.amazon.com/healthlake/latest/devguide/working-with-FHIR-healthlake.html) en la Guía para desarrolladores de Amazon HealthLake

- Para obtener API más información, [DeleteFhirDatastore](#) consulte AWS CLI la Referencia de comandos.

## **describe-fhir-datastore**

En el siguiente ejemplo de código, se muestra cómo usar `describe-fhir-datastore`.

AWS CLI

Para describir un almacén FHIR de datos

El siguiente `describe-fhir-datastore` ejemplo muestra cómo encontrar las propiedades de un almacén de datos en Amazon HealthLake.

```
aws healthlake describe-fhir-datastore \  
  --datastore-id "1f2f459836ac6c513ce899f9e4f66a59" \  
  --region us-east-1
```

Salida:

```
{
  "DatastoreProperties": {
    "PreloadDataConfig": {
      "PreloadDataType": "SYNTHEA"
    },
    "DatastoreName": "FhirTestDatastore",
    "DatastoreArn": "arn:aws:healthlake:us-east-1:(AWS Account ID):datastore/
(Datastore ID)",
    "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/
(Datastore ID)/r4/",
    "DatastoreStatus": "CREATING",
    "DatastoreTypeVersion": "R4",
    "DatastoreId": "(Datastore ID)"
  }
}
```

Para obtener más información, consulte [Creación y supervisión de almacenes de FHIR datos](#) en la Guía para HealthLake desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeFhirDatastore](#) la Referencia de AWS CLI comandos.

## describe-fhir-export-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-fhir-export-job`.

### AWS CLI

Para describir un trabajo FHIR de exportación

El siguiente `describe-fhir-export-job` ejemplo muestra cómo encontrar las propiedades de un trabajo de FHIR exportación en Amazon HealthLake.

```
aws healthlake describe-fhir-export-job \
  --datastore-id (Datastore ID) \
  --job-id 9b9a51943afaedd0a8c0c26c49135a31
```

Salida:

```
{
  "ExportJobProperties": {
    "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)",
```

```

    "JobStatus": "IN_PROGRESS",
    "JobId": "9009813e9d69ba7cf79bcb3468780f16",
    "SubmitTime": 1609175692.715,
    "OutputDataConfig": {
      "S3Uri": "s3://(Bucket Name)/(Prefix
Name)/59593b2d0367ce252b5e66bf5fd6b574-
FHIR_EXPORT-9009813e9d69ba7cf79bcb3468780f16/"
    },
    "DatastoreId": "(Datastore ID)"
  }
}

```

Para obtener más información, consulte [Exportación de archivos desde un almacén de FHIR datos](#) en la Guía para HealthLake desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeFhirExportJob](#) la Referencia de AWS CLI comandos.

## describe-fhir-import-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-fhir-import-job`.

### AWS CLI

Para describir un trabajo FHIR de importación

El siguiente `describe-fhir-import-job` ejemplo muestra cómo aprender las propiedades de un trabajo de FHIR importación con Amazon HealthLake.

```

aws healthlake describe-fhir-import-job \
  --datastore-id (Datastore ID) \
  --job-id c145fbb27b192af392f8ce6e7838e34f \
  --region us-east-1

```

Salida:

```

{
  "ImportJobProperties": {
    "InputDataConfig": {
      "S3Uri": "s3://(Bucket Name)/(Prefix Name)/"
      { "arrayitem2": 2 }
    },

```



```

    "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)",
    "JobStatus": "COMPLETED",
    "JobId": "c145fbb27b192af392f8ce6e7838e34f",
    "SubmitTime": 1606272542.161,
    "EndTime": 1606272609.497,
    "DatastoreId": "(Datastore ID)"
  }
}

```

Para obtener más información, consulta [Importación de archivos a un almacén de FHIR datos](#) en la Guía para HealthLake desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeFhirImportJob](#) la Referencia de AWS CLI comandos.

## list-fhir-datastores

En el siguiente ejemplo de código, se muestra cómo usar `list-fhir-datastores`.

### AWS CLI

Para enumerar FHIR los almacenes de datos

El siguiente `list-fhir-datastores` ejemplo muestra cómo utilizar el comando y cómo los usuarios pueden filtrar los resultados en función del estado del almacén de datos en Amazon HealthLake.

```

aws healthlake list-fhir-datastores \
  --region us-east-1 \
  --filter DatastoreStatus=ACTIVE

```

Salida:

```

{
  "DatastorePropertiesList": [
    {
      "PreloadDataConfig": {
        "PreloadDataType": "SYNTHEA"
      },
      "DatastoreName": "FhirTestDatastore",
      "DatastoreArn": "arn:aws:healthlake:us-east-1:<AWS Account ID>:datastore/
<Datastore ID>",

```

```

    "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/
<Datastore ID>/r4/",
    "DatastoreStatus": "ACTIVE",
    "DatastoreTypeVersion": "R4",
    "CreatedAt": 1605574003.209,
    "DatastoreId": "<Datastore ID>"
  },
  {
    "DatastoreName": "Demo",
    "DatastoreArn": "arn:aws:healthlake:us-east-1:<AWS Account ID>:datastore/
<Datastore ID>",
    "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/
<Datastore ID>/r4/",
    "DatastoreStatus": "ACTIVE",
    "DatastoreTypeVersion": "R4",
    "CreatedAt": 1603761064.881,
    "DatastoreId": "<Datastore ID>"
  }
]
}

```

Para obtener más información, consulte [Creación y supervisión de un almacén de FHIR datos](#) en la Guía para HealthLake desarrolladores de Amazon.

- Para API obtener más información, consulte [ListFhirDatastores](#) la Referencia de AWS CLI comandos.

## list-fhir-export-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-fhir-export-jobs`.

### AWS CLI

Para ver una lista de todos los trabajos de FHIR exportación

El siguiente `list-fhir-export-jobs` ejemplo muestra cómo utilizar el comando para ver una lista de los trabajos de exportación asociados a una cuenta.

```

aws healthlake list-fhir-export-jobs \
  --datastore-id (Datastore ID) \
  --submitted-before (DATE Like 2024-10-13T19:00:00Z) \
  --submitted-after (DATE Like 2020-10-13T19:00:00Z) \
  --job-name "FHIR-EXPORT" \

```

```
--job-status SUBMITTED \
--max-results (Integer between 1 and 500)
```

Salida:

```
{
  "ExportJobProperties": {
    "OutputDataConfig": {
      "S3Uri": "s3://(Bucket Name)/(Prefix Name)/"
      "S3Configuration": {
        "S3Uri": "s3://(Bucket Name)/(Prefix Name)/",
        "KmsKeyId" : "(KmsKey Id)"
      },
    },
    "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)",
    "JobStatus": "COMPLETED",
    "JobId": "c145fbb27b192af392f8ce6e7838e34f",
    "JobName": "FHIR-EXPORT",
    "SubmitTime": 1606272542.161,
    "EndTime": 1606272609.497,
    "DatastoreId": "(Datastore ID)"
  }
}
"NextToken": String
```

Para obtener más información, consulte [Exportación de archivos desde un almacén de FHIR datos](#) en la Guía para HealthLake desarrolladores de Amazon.

- Para API obtener más información, consulte [ListFhirExportJobs](#) la Referencia de AWS CLI comandos.

## list-fhir-import-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-fhir-import-jobs`.

### AWS CLI

Para ver una lista de todos los trabajos de FHIR importación

El siguiente `list-fhir-import-jobs` ejemplo muestra cómo utilizar el comando para ver una lista de todos los trabajos de importación asociados a una cuenta.

```
aws healthlake list-fhir-import-jobs \
```

```

--datastore-id (Datastore ID) \
--submitted-before (DATE Like 2024-10-13T19:00:00Z) \
--submitted-after (DATE Like 2020-10-13T19:00:00Z ) \
--job-name "FHIR-IMPORT" \
--job-status SUBMITTED \
-max-results (Integer between 1 and 500)

```

Salida:

```

{
  "ImportJobProperties": {
    "OutputDataConfig": {
      "S3Uri": "s3://(Bucket Name)/(Prefix Name)/",
      "S3Configuration": {
        "S3Uri": "s3://(Bucket Name)/(Prefix Name)/",
        "KmsKeyId" : "(KmsKey Id)"
      },
    },
    "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)",
    "JobStatus": "COMPLETED",
    "JobId": "c145fbb27b192af392f8ce6e7838e34f",
    "JobName": "FHIR-IMPORT",
    "SubmitTime": 1606272542.161,
    "EndTime": 1606272609.497,
    "DatastoreId": "(Datastore ID)"
  }
}
"NextToken": String

```

Para obtener más información, consulte [Importación de archivos a FHIR Data Store](#) en la Guía para HealthLake desarrolladores de Amazon.

- Para API obtener más información, consulte [ListFhirImportJobs](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un almacén de datos

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas asociadas al almacén de datos especificado. :

```
aws healthlake list-tags-for-resource \  
  --resource-arn "arn:aws:healthLake:us-east-1:674914422125:datastore/  
fhir/0725c83f4307f263e16fd56b6d8ebdbe" \  
  --region us-east-1
```

Salida:

```
{  
  "tags": {  
    "key": "value",  
    "key1": "value1"  
  }  
}
```

Para obtener más información, consulta Cómo [etiquetar recursos en Amazon HealthLake en la Guía para HealthLake desarrolladores de Amazon](#).

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## start-fhir-export-job

En el siguiente ejemplo de código, se muestra cómo usar `start-fhir-export-job`.

### AWS CLI

Para iniciar un trabajo FHIR de exportación

El siguiente `start-fhir-export-job` ejemplo muestra cómo iniciar un trabajo de FHIR exportación con Amazon HealthLake.

```
aws healthlake start-fhir-export-job \  
  --output-data-config S3Uri="s3://(Bucket Name)/(Prefix Name)/" \  
  --datastore-id (Datastore ID) \  
  --data-access-role-arn arn:aws:iam::(AWS Account ID):role/(Role Name)
```

Salida:

```
{
```

```

    "DatastoreId": "(Datastore ID)",
    "JobStatus": "SUBMITTED",
    "JobId": "9b9a51943afaedd0a8c0c26c49135a31"
  }

```

Para obtener más información, consulte [Exportación de archivos desde un almacén de FHIR datos](#) en la Guía para HealthLake desarrolladores de Amazon.

- Para API obtener más información, consulte [StartFhirExportJob](#) la Referencia de AWS CLI comandos.

## start-fhir-import-job

En el siguiente ejemplo de código, se muestra cómo usar `start-fhir-import-job`.

### AWS CLI

Para iniciar un trabajo FHIR de importación

El siguiente `start-fhir-import-job` ejemplo muestra cómo iniciar un trabajo de FHIR importación con Amazon HealthLake.

```

aws healthlake start-fhir-import-job \
  --input-data-config S3Uri="s3://(Bucket Name)/(Prefix Name)/" \
  --datastore-id (Datastore ID) \
  --data-access-role-arn "arn:aws:iam::(AWS Account ID):role/(Role Name)" \
  --region us-east-1

```

Salida:

```

{
  "DatastoreId": "(Datastore ID)",
  "JobStatus": "SUBMITTED",
  "JobId": "c145fbb27b192af392f8ce6e7838e34f"
}

```

Para obtener más información, consulte [Importación de archivos a un almacén de FHIR datos](https://docs.aws.amazon.com/healthlake/latest/devguide/import-datastore.html) en la Guía para HealthLake desarrolladores de Amazon.

- Para API obtener más información, consulte [StartFhirImportJob](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir una etiqueta al almacén de datos

En el siguiente `tag-resource` ejemplo, se muestra cómo añadir una etiqueta a un almacén de datos.

```
aws healthlake tag-resource \  
  --resource-arn "arn:aws:healthlake:us-east-1:691207106566:datastore/  
fhir/0725c83f4307f263e16fd56b6d8ebdbe" \  
  --tags '[{"Key": "key1", "Value": "value1"}]' \  
  --region us-east-1
```

Este comando no genera ninguna salida.

Para obtener más información, consulta «Añadir una etiqueta a un almacén de datos < <https://docs.aws.amazon.com/healthlake/latest/devguide/add-a-tag.html>>'\_\_ en la Guía para desarrolladores de Amazon. HealthLake .

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar etiquetas de un almacén de datos.

En el siguiente `untag-resource` ejemplo, se muestra cómo eliminar etiquetas de un almacén de datos.

```
aws healthlake untag-resource \  
  --resource-arn "arn:aws:healthlake:us-east-1:674914422125:datastore/fhir/  
b91723d65c6fdeb1d26543a49d2ed1fa" \  
  --tag-keys '["key1"]' \  
  --region us-east-1
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Eliminar etiquetas de un almacén de datos](#) en la Guía para HealthLake desarrolladores de Amazon.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## HealthOmics ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with HealthOmics.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **abort-multipart-read-set-upload**

En el siguiente ejemplo de código, se muestra cómo usar `abort-multipart-read-set-upload`.

AWS CLI

Para detener la carga de un conjunto de lectura de varias partes

El siguiente `abort-multipart-read-set-upload` ejemplo detiene la carga de un conjunto de lectura de varias partes en el almacén de HealthOmics secuencias.

```
aws omics abort-multipart-read-set-upload \  
  --sequence-store-id 0123456789 \  
  --upload-id 1122334455
```

Este comando no genera ninguna salida.



Para obtener más información, consulte [Carga directa a un almacén de secuencias](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [AbortMultipartReadSetUpload](#) la Referencia de AWS CLI comandos.

## accept-share

En el siguiente ejemplo de código, se muestra cómo usar `accept-share`.

### AWS CLI

Para aceptar una parte de los datos del almacén de análisis

El siguiente `accept-share` ejemplo acepta una parte de los datos del almacén de HealthOmics análisis.

```
aws omics accept-share \  
----share-id "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a"
```

Salida:

```
{  
  "status": "ACTIVATING"  
}
```

Para obtener más información, consulte [Uso compartido entre cuentas](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [AcceptShare](#) la Referencia de AWS CLI comandos.

## batch-delete-read-set

En el siguiente ejemplo de código, se muestra cómo usar `batch-delete-read-set`.

### AWS CLI

Para eliminar varios conjuntos de lectura

En el siguiente `batch-delete-read-set` ejemplo, se eliminan dos conjuntos de lectura.

```
aws omics batch-delete-read-set \  
-----
```

```
--sequence-store-id 1234567890 \  
--ids 1234567890 0123456789
```

Si se produce un error al eliminar alguno de los conjuntos de lectura especificados, el servicio devuelve una lista de errores.

```
{  
  "errors": [  
    {  
      "code": "",  
      "id": "0123456789",  
      "message": "The specified readset does not exist."  
    }  
  ]  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [BatchDeleteReadSet](#) la Referencia de AWS CLI comandos.

## cancel-annotation-import-job

En el siguiente ejemplo de código, se muestra cómo usar `cancel-annotation-import-job`.

### AWS CLI

Para cancelar un trabajo de importación de anotaciones

En el siguiente `cancel-annotation-import-job` ejemplo, se cancela un trabajo de importación de anotaciones con un ID. `04f57618-xmpl-4fd0-9349-e5a85aefb997`

```
aws omics cancel-annotation-import-job \  
--job-id 04f57618-xmpl-4fd0-9349-e5a85aefb997
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [CancelAnnotationImportJob](#) la Referencia de AWS CLI comandos.

## cancel-run

En el siguiente ejemplo de código, se muestra cómo usar `cancel-run`.

### AWS CLI

Para cancelar una ejecución

En el siguiente `cancel-run` ejemplo, se cancela una ejecución con un ID. 1234567

```
aws omics cancel-run \  
  --id 1234567
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [CancelRun](#) la Referencia de AWS CLI comandos.

## cancel-variant-import-job

En el siguiente ejemplo de código, se muestra cómo usar `cancel-variant-import-job`.

### AWS CLI

Para cancelar un trabajo de importación de variantes

En el siguiente `cancel-variant-import-job` ejemplo, se cancela un trabajo de importación de variantes con un ID `69cb65d6-xmpl-4a4a-9025-4565794b684e`.

```
aws omics cancel-variant-import-job \  
  --job-id 69cb65d6-xmpl-4a4a-9025-4565794b684e
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [CancelVariantImportJob](#) la Referencia de AWS CLI comandos.

## complete-multipart-read-set-upload

En el siguiente ejemplo de código, se muestra cómo usar `complete-multipart-read-set-upload`.

## AWS CLI

Para finalizar una carga de varias partes una vez que haya cargado todos los componentes.

En el siguiente `complete-multipart-read-set-upload` ejemplo, se concluye una carga multiparte en un almacén de secuencias una vez que se han cargado todos los componentes.

```
aws omics complete-multipart-read-set-upload \  
  --sequence-store-id 0123456789 \  
  --upload-id 1122334455 \  
  --parts ' [{"checksum": "gaCBQMe+rpCFZxLpoP6gydBoXaKKDA/  
Vobh5zBDb4W4=", "partNumber": 1, "partSource": "SOURCE1"} ]'
```

Salida:

```
{  
  "readSetId": "00000000001"  
  "readSetId": "00000000002"  
  "readSetId": "00000000003"  
}
```

Para obtener más información, consulte [Carga directa a un almacén de secuencias](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [CompleteMultipartReadSetUpload](#) la Referencia de AWS CLI comandos.

## `create-annotation-store-version`

En el siguiente ejemplo de código, se muestra cómo usar `create-annotation-store-version`.

## AWS CLI

Para crear una nueva versión de un almacén de anotaciones

En el siguiente `create-annotation-store-version` ejemplo, se crea una nueva versión de un almacén de anotaciones.

```
aws omics create-annotation-store-version \  
  --name my_annotation_store \  
  --version-name my_version
```

Salida:

```
{
  "creationTime": "2023-07-21T17:15:49.251040+00:00",
  "id": "3b93cdef69d2",
  "name": "my_annotation_store",
  "reference": {
    "referenceArn": "arn:aws:omics:us-
west-2:555555555555:referenceStore/6505293348/reference/5987565360"
  },
  "status": "CREATING",
  "versionName": "my_version"
}
```

Para obtener más información, consulte [Creación de nuevas versiones de almacenes de anotaciones](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [CreateAnnotationStoreVersion](#) la Referencia de AWS CLI comandos.

## create-annotation-store

En el siguiente ejemplo de código, se muestra cómo usar create-annotation-store.

### AWS CLI

Ejemplo 1: Para crear un almacén VCF de anotaciones

En el siguiente create-annotation-store ejemplo, se crea un almacén de anotaciones de VCF formato.

```
aws omics create-annotation-store \
  --name my_ann_store \
  --store-format VCF \
  --reference referenceArn=arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890
```

Salida:

```
{
  "creationTime": "2022-11-23T22:48:39.226492Z",
```

```

    "id": "0a91xmplc71f",
    "name": "my_ann_store",
    "reference": {
      "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
    },
    "status": "CREATING",
    "storeFormat": "VCF"
  }

```

Ejemplo 2: Para crear un almacén de TSV anotaciones

En el siguiente `create-annotation-store` ejemplo, se crea un almacén de anotaciones de TSV formato.

```

aws omics create-annotation-store \
  --name tsv_ann_store \
  --store-format TSV \
  --reference referenceArn=arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890 \
  --store-options file://tsv-store-options.json

```

`tsv-store-options.json` configura las opciones de formato para las anotaciones.

```

{
  "tsvStoreOptions": {
    "annotationType": "CHR_START_END_ZERO_BASE",
    "formatToHeader": {
      "CHR": "chromosome",
      "START": "start",
      "END": "end"
    },
    "schema": [
      {
        "chromosome": "STRING"
      },
      {
        "start": "LONG"
      },
      {
        "end": "LONG"
      }
    ]
  }
}

```

```

        "name": "STRING"
      }
    ]
  }
}

```

**Salida:**

```

{
  "creationTime": "2022-11-30T01:28:08.525586Z",
  "id": "861cxmpl96b0",
  "name": "tsv_ann_store",
  "reference": {
    "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
  },
  "status": "CREATING",
  "storeFormat": "TSV",
  "storeOptions": {
    "tsvStoreOptions": {
      "annotationType": "CHR_START_END_ZERO_BASE",
      "formatToHeader": {
        "CHR": "chromosome",
        "END": "end",
        "START": "start"
      },
      "schema": [
        {
          "chromosome": "STRING"
        },
        {
          "start": "LONG"
        },
        {
          "end": "LONG"
        },
        {
          "name": "STRING"
        }
      ]
    }
  }
}

```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [CreateAnnotationStore](#) la Referencia de AWS CLI comandos.

## create-multipart-read-set-upload

En el siguiente ejemplo de código, se muestra cómo usar `create-multipart-read-set-upload`.

### AWS CLI

Para iniciar la carga de un conjunto de lectura de varias partes.

En el siguiente `create-multipart-read-set-upload` ejemplo, se inicia la carga de un conjunto de lectura de varias partes.

```
aws omics create-multipart-read-set-upload \  
  --sequence-store-id 0123456789 \  
  --name HG00146 \  
  --source-file-type FASTQ \  
  --subject-id mySubject\  
  --sample-id mySample\  
  --description "FASTQ for HG00146"\  
  --generated-from "1000 Genomes"
```

Salida:

```
{  
  "creationTime": "2022-07-13T23:25:20Z",  
  "description": "FASTQ for HG00146",  
  "generatedFrom": "1000 Genomes",  
  "name": "HG00146",  
  "sampleId": "mySample",  
  "sequenceStoreId": "0123456789",  
  "sourceFileType": "FASTQ",  
  "subjectId": "mySubject",  
  "uploadId": "1122334455"  
}
```



Para obtener más información, consulte [Carga directa a un almacén de secuencias](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [CreateMultipartReadSetUpload](#) la Referencia de AWS CLI comandos.

## **create-reference-store**

En el siguiente ejemplo de código, se muestra cómo usar `create-reference-store`.

### AWS CLI

Para crear un almacén de referencia

En el siguiente `create-reference-store` ejemplo, se crea un almacén de referencias `my-ref-store`.

```
aws omics create-reference-store \  
  --name my-ref-store
```

Salida:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890",  
  "creationTime": "2022-11-22T22:13:25.947Z",  
  "id": "1234567890",  
  "name": "my-ref-store"  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [CreateReferenceStore](#) la Referencia de AWS CLI comandos.

## **create-run-group**

En el siguiente ejemplo de código, se muestra cómo usar `create-run-group`.

### AWS CLI

Para crear un grupo de carreras

En el siguiente `create-run-group` ejemplo, se crea un grupo de carreras denominado `cram-converter`.

```
aws omics create-run-group \  
  --name cram-converter \  
  --max-cpus 20 \  
  --max-duration 600
```

Salida:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:runGroup/1234567",  
  "id": "1234567",  
  "tags": {}  
}
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [CreateRunGroup](#) la Referencia de AWS CLI comandos.

## **create-sequence-store**

En el siguiente ejemplo de código, se muestra cómo usar `create-sequence-store`.

AWS CLI

Para crear un almacén de secuencias

En el siguiente `create-sequence-store` ejemplo, se crea un almacén de secuencias.

```
aws omics create-sequence-store \  
  --name my-seq-store
```

Salida:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890",  
  "creationTime": "2022-11-23T01:24:33.629Z",  
  "id": "1234567890",
```

```
"name": "my-seq-store"  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [CreateSequenceStore](#) la Referencia de AWS CLI comandos.

## create-share

En el siguiente ejemplo de código, se muestra cómo usar `create-share`.

### AWS CLI

Para crear un recurso compartido de un almacén de HealthOmics análisis

En el siguiente `create-share` ejemplo, se muestra cómo crear una parte de un almacén de HealthOmics análisis que pueda ser aceptada por un suscriptor ajeno a la cuenta.

```
aws omics create-share \  
  --resource-arn "arn:aws:omics:us-west-2:555555555555:variantStore/  
omics_dev_var_store" \  
  --principal-subscriber "123456789012" \  
  --name "my_Share-123"
```

Salida:

```
{  
  "shareId": "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a",  
  "name": "my_Share-123",  
  "status": "PENDING"  
}
```

Para obtener más información, consulte [Uso compartido entre cuentas](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte la Referencia [CreateShare](#) de AWS CLI comandos.

## create-variant-store

En el siguiente ejemplo de código, se muestra cómo usar `create-variant-store`.

## AWS CLI

Para crear un almacén de variantes

En el siguiente `create-variant-store` ejemplo, se crea un almacén de variantes denominadomy\_var\_store.

```
aws omics create-variant-store \  
  --name my_var_store \  
  --reference referenceArn=arn:aws:omics:us-  
west-2:123456789012:referenceStore/1234567890/reference/1234567890
```

Salida:

```
{  
  "creationTime": "2022-11-23T22:09:07.534499Z",  
  "id": "02dexmplcfdd",  
  "name": "my_var_store",  
  "reference": {  
    "referenceArn": "arn:aws:omics:us-  
west-2:123456789012:referenceStore/1234567890/reference/1234567890"  
  },  
  "status": "CREATING"  
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [CreateVariantStore](#) la Referencia de AWS CLI comandos.

## create-workflow

En el siguiente ejemplo de código, se muestra cómo usar `create-workflow`.

## AWS CLI

Para crear un flujo de trabajo

En el siguiente `create-workflow` ejemplo, se crea un WDL flujo de trabajo.

```
aws omics create-workflow \  
  --name my_workflow \  
  --reference referenceArn=arn:aws:omics:us-  
west-2:123456789012:referenceStore/1234567890/reference/1234567890
```

```
--name cram-converter \  
--engine WDL \  
--definition-zip fileb://workflow-crambam.zip \  
--parameter-template file://workflow-params.json
```

`workflow-crambam.zip` es un ZIP archivo que contiene una definición de flujo de trabajo. `workflow-params.json` define los parámetros de tiempo de ejecución del flujo de trabajo.

```
{  
  "ref_fasta" : {  
    "description": "Reference genome fasta file",  
    "optional": false  
  },  
  "ref_fasta_index" : {  
    "description": "Index of the reference genome fasta file",  
    "optional": false  
  },  
  "ref_dict" : {  
    "description": "dictionary file for 'ref_fasta'",  
    "optional": false  
  },  
  "input_cram" : {  
    "description": "The Cram file to convert to BAM",  
    "optional": false  
  },  
  "sample_name" : {  
    "description": "The name of the input sample, used to name the output BAM",  
    "optional": false  
  }  
}
```

Salida:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567",  
  "id": "1234567",  
  "status": "CREATING",  
  "tags": {}  
}
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [CreateWorkflow](#) la Referencia de AWS CLI comandos.

## **delete-annotation-store-versions**

En el siguiente ejemplo de código, se muestra cómo usar `delete-annotation-store-versions`.

### AWS CLI

Para eliminar una versión del almacén de anotaciones

En el siguiente `delete-annotation-store-versions` ejemplo, se elimina una versión del almacén de anotaciones.

```
aws omics delete-annotation-store-versions \  
  --name my_annotation_store \  
  --versions my_version
```

Salida:

```
{  
  "errors": []  
}
```

Para obtener más información, consulte [Creación de nuevas versiones de almacenes de anotaciones](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [DeleteAnnotationStoreVersions](#) la Referencia de AWS CLI comandos.

## **delete-annotation-store**

En el siguiente ejemplo de código, se muestra cómo usar `delete-annotation-store`.

### AWS CLI

Para eliminar un almacén de anotaciones

En el siguiente `delete-annotation-store` ejemplo, se elimina un almacén de anotaciones denominado `my_vcf_store`

```
aws omics delete-annotation-store \  
  --name my_vcf_store
```

Salida:

```
{  
  "status": "DELETING"  
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [DeleteAnnotationStore](#) la Referencia de AWS CLI comandos.

## delete-reference-store

En el siguiente ejemplo de código, se muestra cómo usar `delete-reference-store`.

AWS CLI

Para eliminar un almacén de referencia

En el siguiente `delete-reference-store` ejemplo, se elimina un almacén de referencias con un ID. `1234567890`

```
aws omics delete-reference-store \  
  --id 1234567890
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [DeleteReferenceStore](#) la Referencia de AWS CLI comandos.

## delete-reference

En el siguiente ejemplo de código, se muestra cómo usar `delete-reference`.

## AWS CLI

Para eliminar una referencia

En el siguiente `delete-reference` ejemplo, se elimina una referencia.

```
aws omics delete-reference \  
  --reference-store-id 1234567890 \  
  --id 1234567890
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [DeleteReference](#) la Referencia de AWS CLI comandos.

## `delete-run-group`

En el siguiente ejemplo de código, se muestra cómo usar `delete-run-group`.

## AWS CLI

Para eliminar un grupo de carreras

En el siguiente `delete-run-group` ejemplo, se elimina un grupo de carreras con un ID. 1234567

```
aws omics delete-run-group \  
  --id 1234567
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [DeleteRunGroup](#) la Referencia de AWS CLI comandos.

## `delete-run`

En el siguiente ejemplo de código, se muestra cómo usar `delete-run`.



## AWS CLI

Para eliminar un flujo de trabajo, ejecute

En el siguiente `delete-run` ejemplo, se elimina una ejecución con un ID. 1234567

```
aws omics delete-run \  
  --id 1234567
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [DeleteRun](#) la Referencia de AWS CLI comandos.

## `delete-sequence-store`

En el siguiente ejemplo de código, se muestra cómo usar `delete-sequence-store`.

## AWS CLI

Para eliminar un almacén de secuencias

En el siguiente `delete-sequence-store` ejemplo, se elimina un almacén de secuencias con un ID. 1234567890

```
aws omics delete-sequence-store \  
  --id 1234567890
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [DeleteSequenceStore](#) la Referencia de AWS CLI comandos.

## `delete-share`

En el siguiente ejemplo de código, se muestra cómo usar `delete-share`.

## AWS CLI

Para eliminar una parte de los datos de HealthOmics análisis

En el siguiente `delete-share` ejemplo, se elimina un intercambio de datos de análisis entre cuentas.

```
aws omics delete-share \  
  --share-id "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a"
```

Salida:

```
{  
  "status": "DELETING"  
}
```

Para obtener más información, consulte [Uso compartido entre cuentas](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [DeleteShare](#) la Referencia de AWS CLI comandos.

## **delete-variant-store**

En el siguiente ejemplo de código, se muestra cómo usar `delete-variant-store`.

### AWS CLI

Para eliminar un almacén de variantes

En el siguiente `delete-variant-store` ejemplo, se elimina un almacén de variantes denominado `my_var_store`.

```
aws omics delete-variant-store \  
  --name my_var_store
```

Salida:

```
{  
  "status": "DELETING"  
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [DeleteVariantStore](#) la Referencia de AWS CLI comandos.

## delete-workflow

En el siguiente ejemplo de código, se muestra cómo usar `delete-workflow`.

### AWS CLI

Para eliminar un flujo de trabajo

En el siguiente `delete-workflow` ejemplo, se elimina un flujo de trabajo con un ID. 1234567

```
aws omics delete-workflow \  
  --id 1234567
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [DeleteWorkflow](#) la Referencia de AWS CLI comandos.

## get-annotation-import-job

En el siguiente ejemplo de código, se muestra cómo usar `get-annotation-import-job`.

### AWS CLI

Para ver un trabajo de importación de anotaciones

En el siguiente `get-annotation-import-job` ejemplo, se obtienen detalles sobre un trabajo de importación de anotaciones.

```
aws omics get-annotation-import-job \  
  --job-id 984162c7-xmpl-4d23-ab47-286f7950bfbf
```

Salida:

```
{  
  "creationTime": "2022-11-30T01:40:11.017746Z",  
  "destinationName": "tsv_ann_store",  
  "id": "984162c7-xmpl-4d23-ab47-286f7950bfbf",  
  "items": [  
    {  
      "jobStatus": "COMPLETED",  
      "source": "s3://omics-artifacts-01d6xmpl4e72dd32/targetedregions.bed.gz"    }  
  ]  
}
```

```

    }
  ],
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ",
  "runLeftNormalization": false,
  "status": "COMPLETED",
  "updateTime": "2022-11-30T01:42:39.134009Z"
}

```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetAnnotationImportJob](#) la Referencia de AWS CLI comandos.

## get-annotation-store-version

En el siguiente ejemplo de código, se muestra cómo usar `get-annotation-store-version`.

### AWS CLI

Para recuperar los metadatos de una versión del almacén de anotaciones

En el siguiente `get-annotation-store-version` ejemplo, se recuperan los metadatos de la versión del almacén de anotaciones solicitada.

```

aws omics get-annotation-store-version \
  --name my_annotation_store \
  --version-name my_version

```

Salida:

```

{
  "storeId": "4934045d1c6d",
  "id": "2a3f4a44aa7b",
  "status": "ACTIVE",
  "versionArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/
my_annotation_store/version/my_version",
  "name": "my_annotation_store",
  "versionName": "my_version",
  "creationTime": "2023-07-21T17:15:49.251040+00:00",
  "updateTime": "2023-07-21T17:15:56.434223+00:00",
  "statusMessage": ""
}

```

```
"versionSizeBytes": 0
}
```

Para obtener más información, consulte [Creación de nuevas versiones de almacenes de anotaciones](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [GetAnnotationStoreVersion](#) la Referencia de AWS CLI comandos.

## get-annotation-store

En el siguiente ejemplo de código, se muestra cómo usar `get-annotation-store`.

### AWS CLI

Para ver un almacén de anotaciones

En el siguiente `get-annotation-store` ejemplo se obtienen detalles sobre un almacén de anotaciones denominado `my_ann_store`

```
aws omics get-annotation-store \
  --name my_ann_store
```

Salida:

```
{
  "creationTime": "2022-11-23T22:48:39.226492Z",
  "id": "0a91xmplc71f",
  "name": "my_ann_store",
  "reference": {
    "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
  },
  "status": "CREATING",
  "storeArn": "arn:aws:omics:us-west-2:123456789012:annotationStore/my_ann_store",
  "storeFormat": "VCF",
  "storeSizeBytes": 0,
  "tags": {}
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetAnnotationStore](#) la Referencia de AWS CLI comandos.

## get-read-set-activation-job

En el siguiente ejemplo de código, se muestra cómo usar `get-read-set-activation-job`.

### AWS CLI

Para ver un trabajo de activación de un conjunto de lectura

En el siguiente `get-read-set-activation-job` ejemplo, se obtienen detalles sobre un trabajo de activación de conjuntos de lectura.

```
aws omics get-read-set-activation-job \  
  --sequence-store-id 1234567890 \  
  --id 1234567890
```

Salida:

```
{  
  "completionTime": "2022-12-06T22:33:42.828Z",  
  "creationTime": "2022-12-06T22:32:45.213Z",  
  "id": "1234567890",  
  "sequenceStoreId": "1234567890",  
  "sources": [  
    {  
      "readSetId": "1234567890",  
      "status": "FINISHED",  
      "statusMessage": "No activation needed as read set is already in  
ACTIVATING or ACTIVE state."  
    }  
  ],  
  "status": "COMPLETED",  
  "statusMessage": "The job completed successfully."  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetReadSetActivationJob](#) la Referencia de AWS CLI comandos.

## get-read-set-export-job

En el siguiente ejemplo de código, se muestra cómo usar `get-read-set-export-job`.

### AWS CLI

Para ver un trabajo de exportación de un conjunto de lectura

En el siguiente `get-read-set-export-job` ejemplo, se obtienen detalles sobre un trabajo de exportación de conjuntos de lecturas.

```
aws omics get-read-set-export-job \  
  --sequence-store-id 1234567890 \  
  --id 1234567890
```

Salida:

```
{  
  "completionTime": "2022-12-06T22:39:14.491Z",  
  "creationTime": "2022-12-06T22:37:18.612Z",  
  "destination": "s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/",  
  "id": "1234567890",  
  "sequenceStoreId": "1234567890",  
  "status": "COMPLETED",  
  "statusMessage": "The job is submitted and will start soon."  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetReadSetExportJob](#) la Referencia de AWS CLI comandos.

## get-read-set-import-job

En el siguiente ejemplo de código, se muestra cómo usar `get-read-set-import-job`.

### AWS CLI

Para ver un trabajo de importación de conjuntos de lectura

En el siguiente `get-read-set-import-job` ejemplo, se obtienen detalles sobre un trabajo de importación de conjuntos de lecturas.

```
aws omics get-read-set-import-job \
  --sequence-store-id 1234567890 \
  --id 1234567890
```

Salida:

```
{
  "creationTime": "2022-11-23T01:36:38.158Z",
  "id": "1234567890",
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ",
  "sequenceStoreId": "1234567890",
  "sources": [
    {
      "name": "HG00100",
      "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890",
      "sampleId": "bam-sample",
      "sourceFileType": "BAM",
      "sourceFiles": {
        "source1": "s3://omics-artifacts-01d6xmpl4e72dd32/
HG00100.chrom20.ILLUMINA.bwa.GBR.low_coverage.20101123.bam",
        "source2": ""
      },
      "status": "IN_PROGRESS",
      "statusMessage": "The source job is currently in progress.",
      "subjectId": "bam-subject",
      "tags": {
        "aws:omics:sampleId": "bam-sample",
        "aws:omics:subjectId": "bam-subject"
      }
    },
    {
      "name": "HG00146",
      "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890",
      "sampleId": "fastq-sample",
      "sourceFileType": "FASTQ",
      "sourceFiles": {
        "source1": "s3://omics-artifacts-01d6xmpl4e72dd32/
SRR233106_1.filt.fastq.gz",
        "source2": "s3://omics-artifacts-01d6xmpl4e72dd32/
SRR233106_2.filt.fastq.gz"
      }
    }
  ]
}
```



```

    },
    "status": "IN_PROGRESS",
    "statusMessage": "The source job is currently in progress.",
    "subjectId": "fastq-subject",
    "tags": {
      "aws:omics:sampleId": "fastq-sample",
      "aws:omics:subjectId": "fastq-subject"
    }
  },
  {
    "name": "HG00096",
    "referenceArn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/reference/1234567890",
    "sampleId": "cram-sample",
    "sourceFileType": "CRAM",
    "sourceFiles": {
      "source1": "s3://omics-artifacts-01d6xmpl14e72dd32/HG00096.alt_bwamem_GRCh38DH.20150718.GBR.low_coverage.cram",
      "source2": ""
    },
    "status": "IN_PROGRESS",
    "statusMessage": "The source job is currently in progress.",
    "subjectId": "cram-subject",
    "tags": {
      "aws:omics:sampleId": "cram-sample",
      "aws:omics:subjectId": "cram-subject"
    }
  }
],
"status": "IN_PROGRESS",
"statusMessage": "The job is currently in progress."
}

```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetReadSetImportJob](#) la Referencia de AWS CLI comandos.

## get-read-set-metadata

En el siguiente ejemplo de código, se muestra cómo usar `get-read-set-metadata`.

## AWS CLI

Para ver un conjunto de lecturas

En el siguiente `get-read-set-metadata` ejemplo, se obtienen detalles sobre los archivos de un conjunto de lecturas.

```
aws omics get-read-set-metadata \  
  --sequence-store-id 1234567890 \  
  --id 1234567890
```

Salida:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890/  
readSet/1234567890",  
  "creationTime": "2022-11-23T21:55:00.515Z",  
  "fileType": "FASTQ",  
  "files": {  
    "source1": {  
      "contentLength": 310054739,  
      "partSize": 104857600,  
      "totalParts": 3  
    },  
    "source2": {  
      "contentLength": 307846621,  
      "partSize": 104857600,  
      "totalParts": 3  
    }  
  },  
  "id": "1234567890",  
  "name": "HG00146",  
  "referenceArn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/  
reference/1234567890",  
  "sampleId": "fastq-sample",  
  "sequenceInformation": {  
    "alignment": "UNALIGNED",  
    "totalBaseCount": 677717384,  
    "totalReadCount": 8917334  
  },  
  "sequenceStoreId": "1234567890",  
  "status": "ACTIVE",  
  "subjectId": "fastq-subject"
```

```
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetReadSetMetadata](#) la Referencia de AWS CLI comandos.

## get-read-set

En el siguiente ejemplo de código, se muestra cómo usar `get-read-set`.

### AWS CLI

Para descargar un conjunto de lectura

El siguiente `get-read-set` ejemplo descarga la parte 3 de un conjunto de lecturas como `1234567890.3.bam`.

```
aws omics get-read-set \  
  --sequence-store-id 1234567890 \  
  --id 1234567890 \  
  --part-number 3 1234567890.3.bam
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetReadSet](#) la Referencia de AWS CLI comandos.

## get-reference-import-job

En el siguiente ejemplo de código, se muestra cómo usar `get-reference-import-job`.

### AWS CLI

Para ver un trabajo de importación de referencia

En el siguiente `get-reference-import-job` ejemplo, se obtienen detalles sobre un trabajo de importación de referencia.

```
aws omics get-reference-import-job \  
  --reference-store-id 1234567890 \  
  --job-id 1234567890
```

```
--reference-store-id 1234567890 \  
--id 1234567890
```

Salida:

```
{  
  "creationTime": "2022-11-22T22:25:41.124Z",  
  "id": "1234567890",  
  "referenceStoreId": "1234567890",  
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-  
W801XMPL7QZ",  
  "sources": [  
    {  
      "name": "assembly-38",  
      "sourceFile": "s3://omics-artifacts-01d6xmpl4e72dd32/  
Homo_sapiens_assembly38.fasta",  
      "status": "IN_PROGRESS",  
      "statusMessage": "The source job is currently in progress."  
    }  
  ],  
  "status": "IN_PROGRESS",  
  "statusMessage": "The job is currently in progress."  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetReferenceImportJob](#) la Referencia de AWS CLI comandos.

## get-reference-metadata

En el siguiente ejemplo de código, se muestra cómo usar `get-reference-metadata`.

AWS CLI

Para ver una referencia

En el siguiente `get-reference-metadata` ejemplo, se obtienen detalles sobre una referencia.

```
aws omics get-reference-metadata \  

```

```
--reference-store-id 1234567890 \  
--id 1234567890
```

Salida:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/  
reference/1234567890",  
  "creationTime": "2022-11-22T22:27:09.033Z",  
  "files": {  
    "index": {  
      "contentLength": 160928,  
      "partSize": 104857600,  
      "totalParts": 1  
    },  
    "source": {  
      "contentLength": 3249912778,  
      "partSize": 104857600,  
      "totalParts": 31  
    }  
  },  
  "id": "1234567890",  
  "md5": "7ff134953dcca8c8997453bbb80b6b5e",  
  "name": "assembly-38",  
  "referenceStoreId": "1234567890",  
  "status": "ACTIVE",  
  "updateTime": "2022-11-22T22:27:09.033Z"  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetReferenceMetadata](#) la Referencia de AWS CLI comandos.

## get-reference-store

En el siguiente ejemplo de código, se muestra cómo usar `get-reference-store`.

### AWS CLI

Para ver un almacén de referencia

En el siguiente `get-reference-store` ejemplo, se obtienen detalles sobre un almacén de referencia.

```
aws omics get-reference-store \  
--id 1234567890
```

Salida:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890",  
  "creationTime": "2022-09-23T23:27:20.364Z",  
  "id": "1234567890",  
  "name": "my-rstore-0"  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetReferenceStore](#) la Referencia de AWS CLI comandos.

## get-reference

En el siguiente ejemplo de código, se muestra cómo usar `get-reference`.

AWS CLI

Para descargar una referencia sobre el genoma

El siguiente `get-reference` ejemplo descarga la parte 1 de un genoma como `hg38.1.fa`.

```
aws omics get-reference \  
--reference-store-id 1234567890 \  
--id 1234567890 \  
--part-number 1 hg38.1.fa
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetReference](#) la Referencia de AWS CLI comandos.

## get-run-group

En el siguiente ejemplo de código, se muestra cómo usar `get-run-group`.

### AWS CLI

Para ver un grupo de carreras

En el siguiente `get-run-group` ejemplo, se obtienen detalles sobre un grupo de carreras.

```
aws omics get-run-group \  
  --id 1234567
```

Salida:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:runGroup/1234567",  
  "creationTime": "2022-12-01T00:58:42.915219Z",  
  "id": "1234567",  
  "maxCpus": 20,  
  "maxDuration": 600,  
  "name": "cram-convert",  
  "tags": {}  
}
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetRunGroup](#) la Referencia de AWS CLI comandos.

## get-run-task

En el siguiente ejemplo de código, se muestra cómo usar `get-run-task`.

### AWS CLI

Para ver una tarea

En el siguiente `get-run-task` ejemplo, se obtienen detalles sobre una tarea de flujo de trabajo.

```
aws omics get-run-task \  
  --id 1234567
```

```
--id 1234567 \  
--task-id 1234567
```

Salida:

```
{  
  "cpus": 1,  
  "creationTime": "2022-11-30T23:13:00.718651Z",  
  "logStream": "arn:aws:logs:us-west-2:123456789012:log-group:/aws/omics/  
WorkflowLog:log-stream:run/1234567/task/1234567",  
  "memory": 15,  
  "name": "CramToBamTask",  
  "startTime": "2022-11-30T23:17:47.016Z",  
  "status": "COMPLETED",  
  "stopTime": "2022-11-30T23:18:21.503Z",  
  "taskId": "1234567"  
}
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetRunTask](#) la Referencia de AWS CLI comandos.

## get-run

En el siguiente ejemplo de código, se muestra cómo usar `get-run`.

### AWS CLI

Para ver un flujo de trabajo, ejecute

En el siguiente `get-run` ejemplo, se obtienen detalles sobre la ejecución de un flujo de trabajo.

```
aws omics get-run \  
--id 1234567
```

Salida:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",  
  "creationTime": "2022-11-30T22:58:22.615865Z",
```



```

    "digest":
      "sha256:c54bxmpl1742dcc26f7fa1f10e37550ddd8f251f418277c0a58e895b801ed28cf",
      "id": "1234567",
      "name": "cram-to-bam",
      "outputUri": "s3://omics-artifacts-01d6xmpl14e72dd32/workflow-output/",
      "parameters": {
        "ref_dict": "s3://omics-artifacts-01d6xmpl14e72dd32/
Homo_sapiens_assembly38.dict",
        "ref_fasta_index": "s3://omics-artifacts-01d6xmpl14e72dd32/
Homo_sapiens_assembly38.fasta.fai",
        "ref_fasta": "s3://omics-artifacts-01d6xmpl14e72dd32/
Homo_sapiens_assembly38.fasta",
        "sample_name": "NA12878",
        "input_cram": "s3://omics-artifacts-01d6xmpl14e72dd32/NA12878.cram"
      },
      "resourceDigests": {
        "s3://omics-artifacts-01d6xmpl14e72dd32/Homo_sapiens_assembly38.fasta.fai":
"etag:f76371b113734a56cde236bc0372de0a",
        "s3://omics-artifacts-01d6xmpl14e72dd32/Homo_sapiens_assembly38.dict":
"etag:3884c62eb0e53fa92459ed9bfff133ae6",
        "s3://omics-artifacts-01d6xmpl14e72dd32/Homo_sapiens_assembly38.fasta":
"etag:e307d81c605fb91b7720a08f00276842-388",
        "s3://omics-artifacts-01d6xmpl14e72dd32/NA12878.cram":
"etag:a9f52976381286c6143b5cc681671ec6"
      },
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ",
      "startedBy": "arn:aws:iam::123456789012:user/laptop-2020",
      "status": "STARTING",
      "tags": {},
      "workflowId": "1234567",
      "workflowType": "PRIVATE"
  }
}

```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetRun](#) la Referencia de AWS CLI comandos.

## get-sequence-store

En el siguiente ejemplo de código, se muestra cómo usar `get-sequence-store`.

## AWS CLI

Para ver un almacén de secuencias

En el siguiente `get-sequence-store` ejemplo, se obtienen detalles sobre un almacén de secuencias con un ID1234567890.

```
aws omics get-sequence-store \  
  --id 1234567890
```

Salida:

```
{  
  "arn": "arn:aws:omics:us-east-1:123456789012:sequenceStore/1234567890",  
  "creationTime": "2022-11-23T19:55:48.376Z",  
  "id": "1234567890",  
  "name": "my-seq-store"  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetSequenceStore](#) la Referencia de AWS CLI comandos.

## get-share

En el siguiente ejemplo de código, se muestra cómo usar `get-share`.

## AWS CLI

Para recuperar los metadatos sobre una parte de un dato HealthOmics analítico

En el siguiente `get-share` ejemplo, se recuperan los metadatos de un intercambio de datos de análisis entre cuentas.

```
aws omics get-share \  
  --share-id "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a"
```

Salida:

```
{
  "share": {
    "shareId": "495c21bedc889d07d0ab69d710a6841e-
dd75ab7a1a9c384fa848b5bd8e5a7e0a",
    "name": "my_Share-123",
    "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
omics_dev_var_store",
    "principalSubscriber": "123456789012",
    "ownerId": "555555555555",
    "status": "PENDING"
  }
}
```

Para obtener más información, consulte [Uso compartido entre cuentas](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [GetShare](#) la Referencia de AWS CLI comandos.

## get-variant-import-job

En el siguiente ejemplo de código, se muestra cómo usar `get-variant-import-job`.

### AWS CLI

Para ver un trabajo de importación de variantes

En el siguiente `get-variant-import-job` ejemplo, se obtienen detalles sobre un trabajo de importación de variantes.

```
aws omics get-variant-import-job \
  --job-id edd7b8ce-xmpl-47e2-bc99-258cac95a508
```

Salida:

```
{
  "creationTime": "2022-11-23T22:42:50.037812Z",
  "destinationName": "my_var_store",
  "id": "edd7b8ce-xmpl-47e2-bc99-258cac95a508",
  "items": [
    {
      "jobStatus": "IN_PROGRESS",
```

```

      "source": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.known_indels.vcf.gz"
    }
  ],
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ",
  "runLeftNormalization": false,
  "status": "IN_PROGRESS",
  "updateTime": "2022-11-23T22:43:05.898309Z"
}

```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetVariantImportJob](#) la Referencia de AWS CLI comandos.

## get-variant-store

En el siguiente ejemplo de código, se muestra cómo usar `get-variant-store`.

### AWS CLI

Para ver una tienda de variantes

En el siguiente `get-variant-store` ejemplo, se obtienen detalles sobre una tienda de variantes.

```

aws omics get-variant-store \
  --name my_var_store

```

Salida:

```

{
  "creationTime": "2022-11-23T22:09:07.534499Z",
  "id": "02dexmplcfdd",
  "name": "my_var_store",
  "reference": {
    "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
  },
  "status": "CREATING",
}

```

```
"storeArn": "arn:aws:omics:us-west-2:123456789012:variantStore/my_var_store",
"storeSizeBytes": 0,
"tags": {},
"updateTime": "2022-11-23T22:09:24.931711Z"
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetVariantStore](#) la Referencia de AWS CLI comandos.

## get-workflow

En el siguiente ejemplo de código, se muestra cómo usar `get-workflow`.

### AWS CLI

Para ver un flujo de trabajo

En el siguiente `get-workflow` ejemplo, se obtienen detalles sobre un flujo de trabajo con un ID1234567.

```
aws omics get-workflow \
  --id 1234567
```

Salida:

```
{
  "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567",
  "creationTime": "2022-11-30T22:33:16.225368Z",
  "digest":
    "sha256:c54bxmpl742dcc26f7fa1f10e37550ddd8f251f418277c0a58e895b801ed28cf",
  "engine": "WDL",
  "id": "1234567",
  "main": "workflow-crambam.wdl",
  "name": "cram-converter",
  "parameterTemplate": {
    "ref_dict": {
      "description": "dictionary file for 'ref_fasta'"
    },
    "ref_fasta_index": {
```

```

        "description": "Index of the reference genome fasta file"
    },
    "ref_fasta": {
        "description": "Reference genome fasta file"
    },
    "input_cram": {
        "description": "The Cram file to convert to BAM"
    },
    "sample_name": {
        "description": "The name of the input sample, used to name the output
BAM"
    }
},
"status": "ACTIVE",
"statusMessage": "workflow-crambam.wdl\n    workflow CramToBamFlow\n
call CramToBamTask\n        call ValidateSamFile\n    task CramToBamTask\n    task
ValidateSamFile\n",
"tags": {},
"type": "PRIVATE"
}

```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [GetWorkflow](#) la Referencia de AWS CLI comandos.

## list-annotation-import-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-annotation-import-jobs`.

### AWS CLI

Para obtener una lista de los trabajos de importación de anotaciones

A continuación `list-annotation-import-jobs` se obtiene una lista de los trabajos de importación de anotaciones.

```
aws omics list-annotation-import-jobs
```

Salida:

```
{
```

```
"annotationImportJobs": [  
  {  
    "creationTime": "2022-11-30T01:39:41.478294Z",  
    "destinationName": "gff_ann_store",  
    "id": "18a9e792-xmpl-4869-a105-e5b602900444",  
    "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-  
serviceRole-W801XMPL7QZ",  
    "runLeftNormalization": false,  
    "status": "COMPLETED",  
    "updateTime": "2022-11-30T01:47:09.145178Z"  
  },  
  {  
    "creationTime": "2022-11-30T00:45:58.007838Z",  
    "destinationName": "my_ann_store",  
    "id": "4e9eafc8-xmpl-431e-a0b2-3bda27cb600a",  
    "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-  
serviceRole-W801XMPL7QZ",  
    "runLeftNormalization": false,  
    "status": "FAILED",  
    "updateTime": "2022-11-30T00:47:01.706325Z"  
  }  
]  
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListAnnotationImportJobs](#) la Referencia de AWS CLI comandos.

## **list-annotation-store-versions**

En el siguiente ejemplo de código, se muestra cómo usar `list-annotation-store-versions`.

### AWS CLI

Para enumerar todas las versiones de un almacén de anotaciones.

En el siguiente `list-annotation-store-versions` ejemplo, se enumeran todas las versiones que existen de un almacén de anotaciones.

```
aws omics list-annotation-store-versions \
```

```
--name my_annotation_store
```

Salida:

```
{
  "annotationStoreVersions": [
    {
      "storeId": "4934045d1c6d",
      "id": "2a3f4a44aa7b",
      "status": "CREATING",
      "versionArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/
my_annotation_store/version/my_version_2",
      "name": "my_annotation_store",
      "versionName": "my_version_2",
      "creationTime": "2023-07-21T17:20:59.380043+00:00",
      "versionSizeBytes": 0
    },
    {
      "storeId": "4934045d1c6d",
      "id": "4934045d1c6d",
      "status": "ACTIVE",
      "versionArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/
my_annotation_store/version/my_version_1",
      "name": "my_annotation_store",
      "versionName": "my_version_1",
      "creationTime": "2023-07-21T17:15:49.251040+00:00",
      "updateTime": "2023-07-21T17:15:56.434223+00:00",
      "statusMessage": "",
      "versionSizeBytes": 0
    }
  ]
}
```

Para obtener más información, consulte [Creación de nuevas versiones de almacenes de anotaciones](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [ListAnnotationStoreVersions](#) la Referencia de AWS CLI comandos.

## list-annotation-stores

En el siguiente ejemplo de código, se muestra cómo usar `list-annotation-stores`.



## AWS CLI

Para obtener una lista de almacenes de anotaciones

En el siguiente `list-annotation-stores` ejemplo, se obtiene una lista de almacenes de anotaciones.

```
aws omics list-annotation-stores
```

Salida:

```
{
  "annotationStores": [
    {
      "creationTime": "2022-11-23T22:48:39.226492Z",
      "id": "0a91xmplc71f",
      "name": "my_ann_store",
      "reference": {
        "referenceArn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/reference/1234567890"
      },
      "status": "ACTIVE",
      "statusMessage": "",
      "storeArn": "arn:aws:omics:us-west-2:123456789012:annotationStore/my_ann_store",
      "storeFormat": "VCF",
      "storeSizeBytes": 0,
      "updateTime": "2022-11-23T22:53:27.372840Z"
    }
  ]
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListAnnotationStores](#) la Referencia de AWS CLI comandos.

## **list-multipart-read-set-uploads**

En el siguiente ejemplo de código, se muestra cómo usar `list-multipart-read-set-uploads`.

## AWS CLI

Para ver una lista de todas las cargas de conjuntos de lectura multiparte y sus estados.

En el siguiente `list-multipart-read-set-uploads` ejemplo, se enumeran todas las cargas de conjuntos de lectura multiparte y sus estados.

```
aws omics list-multipart-read-set-uploads \  
--sequence-store-id 0123456789
```

Salida:

```
{  
  "uploads":  
    [  
      {  
        "sequenceStoreId": "0123456789",  
        "uploadId": "8749584421",  
        "sourceFileType": "FASTQ",  
        "subjectId": "mySubject",  
        "sampleId": "mySample",  
        "generatedFrom": "1000 Genomes",  
        "name": "HG00146",  
        "description": "FASTQ for HG00146",  
        "creationTime": "2023-11-29T19:22:51.349298+00:00"  
      },  
      {  
        "sequenceStoreId": "0123456789",  
        "uploadId": "5290538638",  
        "sourceFileType": "BAM",  
        "subjectId": "mySubject",  
        "sampleId": "mySample",  
        "generatedFrom": "1000 Genomes",  
        "referenceArn": "arn:aws:omics:us-  
west-2:845448930428:referenceStore/8168613728/reference/2190697383",  
        "name": "HG00146",  
        "description": "BAM for HG00146",  
        "creationTime": "2023-11-29T19:23:33.116516+00:00"  
      },  
      {  
        "sequenceStoreId": "0123456789",  
        "uploadId": "4174220862",  
        "sourceFileType": "BAM",
```

```
    "subjectId": "mySubject",
    "sampleId": "mySample",
    "generatedFrom": "1000 Genomes",
    "referenceArn": "arn:aws:omics:us-
west-2:845448930428:referenceStore/8168613728/reference/2190697383",
    "name": "HG00147",
    "description": "BAM for HG00147",
    "creationTime": "2023-11-29T19:23:47.007866+00:00"
  }
]
}
```

Para obtener más información, consulte [Carga directa a un almacén de secuencias](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [ListMultipartReadSetUploads](#) la Referencia de AWS CLI comandos.

## list-read-set-activation-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-read-set-activation-jobs`.

### AWS CLI

Para obtener una lista de los trabajos de activación de conjuntos de lectura

En el siguiente `list-read-set-activation-jobs` ejemplo, se obtiene una lista de los trabajos de activación de un almacén de secuencias con un identificador1234567890.

```
aws omics list-read-set-activation-jobs \
  --sequence-store-id 1234567890
```

Salida:

```
{
  "activationJobs": [
    {
      "completionTime": "2022-12-06T22:33:42.828Z",
      "creationTime": "2022-12-06T22:32:45.213Z",
      "id": "1234567890",
      "sequenceStoreId": "1234567890",
      "status": "COMPLETED"
    }
  ]
}
```

```
    },
    {
      "creationTime": "2022-12-06T22:35:10.100Z",
      "id": "1234567890",
      "sequenceStoreId": "1234567890",
      "status": "IN_PROGRESS"
    }
  ]
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListReadSetActivationJobs](#) la Referencia de AWS CLI comandos.

## list-read-set-export-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-read-set-export-jobs`.

### AWS CLI

Para obtener una lista de los trabajos de exportación de conjuntos de lectura

En el siguiente `list-read-set-export-jobs` ejemplo, se obtiene una lista de trabajos de exportación para un almacén de secuencias con un identificador `1234567890`.

```
aws omics list-read-set-export-jobs \
  --sequence-store-id 1234567890
```

Salida:

```
{
  "exportJobs": [
    {
      "completionTime": "2022-12-06T22:39:14.491Z",
      "creationTime": "2022-12-06T22:37:18.612Z",
      "destination": "s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/",
      "id": "1234567890",
      "sequenceStoreId": "1234567890",
      "status": "COMPLETED"
    }
  ],
}
```

```
{
  "creationTime": "2022-12-06T22:38:04.871Z",
  "destination": "s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/",
  "id": "1234567890",
  "sequenceStoreId": "1234567890",
  "status": "IN_PROGRESS"
}
]
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListReadSetExportJobs](#) la Referencia de AWS CLI comandos.

## list-read-set-import-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-read-set-import-jobs`.

### AWS CLI

Para obtener una lista de los trabajos de importación de conjuntos de lectura

En el siguiente `list-read-set-import-jobs` ejemplo, se obtiene una lista de trabajos de importación para un almacén de secuencias con un identificador `1234567890`.

```
aws omics list-read-set-import-jobs \
  --sequence-store-id 1234567890
```

Salida:

```
{
  "importJobs": [
    {
      "completionTime": "2022-11-29T18:17:49.244Z",
      "creationTime": "2022-11-29T17:32:47.700Z",
      "id": "1234567890",
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W801XMPL7QZ",
      "sequenceStoreId": "1234567890",
      "status": "COMPLETED"
    }
  ]
}
```

```

    },
    {
      "completionTime": "2022-11-23T22:01:34.090Z",
      "creationTime": "2022-11-23T21:52:43.289Z",
      "id": "1234567890",
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W801XMPL7QZ",
      "sequenceStoreId": "1234567890",
      "status": "COMPLETED_WITH_FAILURES"
    }
  ]
}

```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListReadSetImportJobs](#) la Referencia de AWS CLI comandos.

## list-read-set-upload-parts

En el siguiente ejemplo de código, se muestra cómo usar `list-read-set-upload-parts`.

### AWS CLI

Para enumerar todas las partes de una carga multiparte solicitada para un almacén de secuencias.

En el siguiente `list-read-set-upload-parts` ejemplo, se enumeran todas las partes de una carga multiparte solicitada para un almacén de secuencias.

```

aws omics list-read-set-upload-parts \
  --sequence-store-id 0123456789 \
  --upload-id 1122334455 \
  --part-source SOURCE1

```

Salida:

```

{
  "parts": [
    {

```

```
    "partNumber": 1,
    "partSize": 94371840,
    "file": "SOURCE1",
    "checksum":
"984979b9928ae8d8622286c4a9cd8e99d964a22d59ed0f5722e1733eb280e635",
    "lastUpdatedTime": "2023-02-02T20:14:47.533000+00:00"
  }
  {
    "partNumber": 2,
    "partSize": 10471840,
    "file": "SOURCE1",
    "checksum":
"984979b9928ae8d8622286c4a9cd8e99d964a22d59ed0f5722e1733eb280e635",
    "lastUpdatedTime": "2023-02-02T20:14:47.533000+00:00"
  }
]
}
```

Para obtener más información, consulte [Carga directa a un almacén de secuencias](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [ListReadSetUploadParts](#) la Referencia de AWS CLI comandos.

## list-read-sets

En el siguiente ejemplo de código, se muestra cómo usar `list-read-sets`.

### AWS CLI

Para obtener una lista de conjuntos de lectura

En el siguiente `list-read-sets` ejemplo, se obtiene una lista de conjuntos de lectura para un almacén de secuencias con un identificador `1234567890`.

```
aws omics list-read-sets \
  --sequence-store-id 1234567890
```

Salida:

```
{
```

```
"readSets": [  
  {  
    "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890/  
readSet/1234567890",  
    "creationTime": "2022-11-23T21:55:00.515Z",  
    "fileType": "FASTQ",  
    "id": "1234567890",  
    "name": "HG00146",  
    "referenceArn": "arn:aws:omics:us-  
west-2:123456789012:referenceStore/1234567890/reference/1234567890",  
    "sampleId": "fastq-sample",  
    "sequenceStoreId": "1234567890",  
    "status": "ACTIVE",  
    "subjectId": "fastq-subject"  
  }  
]  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListReadSets](#) la Referencia de AWS CLI comandos.

## list-reference-import-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-reference-import-jobs`.

### AWS CLI

Para obtener una lista de trabajos de importación de referencia

En el siguiente `list-reference-import-jobs` ejemplo, se obtiene una lista de trabajos de importación de referencia para un almacén de referencias con un identificador `1234567890`.

```
aws omics list-reference-import-jobs \  
--reference-store-id 1234567890
```

Salida:

```
{  
  "importJobs": [  
    {  
      "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/importJob/1234567890",  
      "creationTime": "2022-11-23T21:55:00.515Z",  
      "fileType": "FASTQ",  
      "id": "1234567890",  
      "name": "HG00146",  
      "referenceArn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/reference/1234567890",  
      "sampleId": "fastq-sample",  
      "sequenceStoreId": "1234567890",  
      "status": "ACTIVE",  
      "subjectId": "fastq-subject"  
    }  
  ]  
}
```



```
{
  "completionTime": "2022-11-23T19:54:58.204Z",
  "creationTime": "2022-11-23T19:53:20.729Z",
  "id": "1234567890",
  "referenceStoreId": "1234567890",
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W801XMPL7QZ",
  "status": "COMPLETED"
},
{
  "creationTime": "2022-11-23T20:34:03.250Z",
  "id": "1234567890",
  "referenceStoreId": "1234567890",
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W801XMPL7QZ",
  "status": "IN_PROGRESS"
}
]
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListReferenceImportJobs](#) la Referencia de AWS CLI comandos.

## list-reference-stores

En el siguiente ejemplo de código, se muestra cómo usar `list-reference-stores`.

### AWS CLI

Para obtener una lista de tiendas de referencia

En el siguiente `list-reference-stores` ejemplo, se obtiene una lista de tiendas de referencia.

```
aws omics list-reference-stores
```

Salida:

```
{
```

```
"referenceStores": [  
  {  
    "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890",  
    "creationTime": "2022-11-22T22:13:25.947Z",  
    "id": "1234567890",  
    "name": "my-ref-store"  
  }  
]
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListReferenceStores](#) la Referencia de AWS CLI comandos.

## list-references

En el siguiente ejemplo de código, se muestra cómo usar `list-references`.

### AWS CLI

Para obtener una lista de referencias

En el siguiente `list-references` ejemplo, se obtiene una lista de referencias genómicas para un almacén de referencias con un identificador `1234567890`.

```
aws omics list-references \  
  --reference-store-id 1234567890
```

Salida:

```
{  
  "references": [  
    {  
      "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/  
reference/1234567890",  
      "creationTime": "2022-11-22T22:27:09.033Z",  
      "id": "1234567890",  
      "md5": "7ff134953dcca8c8997453bbb80b6b5e",  
      "name": "assembly-38",  
      "referenceStoreId": "1234567890",
```

```
        "status": "ACTIVE",
        "updateTime": "2022-11-22T22:27:09.033Z"
      }
    ]
  }
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListReferences](#) la Referencia de AWS CLI comandos.

## list-run-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-run-groups`.

AWS CLI

Para obtener una lista de los grupos de ejecución

En el siguiente `list-run-groups` ejemplo, se obtiene una lista de grupos de carreras.

```
aws omics list-run-groups
```

Salida:

```
{
  "items": [
    {
      "arn": "arn:aws:omics:us-west-2:123456789012:runGroup/1234567",
      "creationTime": "2022-12-01T00:58:42.915219Z",
      "id": "1234567",
      "maxCpus": 20,
      "maxDuration": 600,
      "name": "cram-convert"
    }
  ]
}
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListRunGroups](#) la Referencia de AWS CLI comandos.

## list-run-tasks

En el siguiente ejemplo de código, se muestra cómo usar `list-run-tasks`.

### AWS CLI

Para obtener una lista de tareas

En el siguiente `list-run-tasks` ejemplo, se obtiene una lista de tareas para la ejecución de un flujo de trabajo.

```
aws omics list-run-tasks \  
  --id 1234567
```

Salida:

```
{  
  "items": [  
    {  
      "cpus": 1,  
      "creationTime": "2022-11-30T23:13:00.718651Z",  
      "memory": 15,  
      "name": "CramToBamTask",  
      "startTime": "2022-11-30T23:17:47.016Z",  
      "status": "COMPLETED",  
      "stopTime": "2022-11-30T23:18:21.503Z",  
      "taskId": "1234567"  
    },  
    {  
      "cpus": 1,  
      "creationTime": "2022-11-30T23:18:32.315606Z",  
      "memory": 4,  
      "name": "ValidateSamFile",  
      "startTime": "2022-11-30T23:23:40.165Z",  
      "status": "COMPLETED",  
      "stopTime": "2022-11-30T23:24:14.766Z",  
      "taskId": "1234567"  
    }  
  ]  
}
```

```
}
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListRunTasks](#) la Referencia de AWS CLI comandos.

## list-runs

En el siguiente ejemplo de código, se muestra cómo usar `list-runs`.

### AWS CLI

Para obtener una lista de las ejecuciones del flujo de trabajo

En el siguiente `list-runs` ejemplo, se obtiene una lista de las ejecuciones del flujo de trabajo.

```
aws omics list-runs
```

Salida:

```
{
  "items": [
    {
      "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",
      "creationTime": "2022-12-02T23:20:01.202074Z",
      "id": "1234567",
      "name": "cram-to-bam",
      "priority": 1,
      "startTime": "2022-12-02T23:29:18.115Z",
      "status": "COMPLETED",
      "stopTime": "2022-12-02T23:57:54.428812Z",
      "storageCapacity": 10,
      "workflowId": "1234567"
    },
    {
      "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",
      "creationTime": "2022-12-03T00:16:57.180066Z",
      "id": "1234567",
      "name": "cram-to-bam",
      "priority": 1,
```

```
    "startTime": "2022-12-03T00:26:50.233Z",
    "status": "FAILED",
    "stopTime": "2022-12-03T00:37:21.451340Z",
    "storageCapacity": 10,
    "workflowId": "1234567"
  },
  {
    "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",
    "creationTime": "2022-12-05T17:57:08.444817Z",
    "id": "1234567",
    "name": "cram-to-bam",
    "status": "STARTING",
    "workflowId": "1234567"
  }
]
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListRuns](#) la Referencia de AWS CLI comandos.

## list-sequence-stores

En el siguiente ejemplo de código, se muestra cómo usar `list-sequence-stores`.

### AWS CLI

Para obtener una lista de almacenes de secuencias

En el siguiente `list-sequence-stores` ejemplo se obtiene una lista de almacenes de secuencias.

```
aws omics list-sequence-stores
```

Salida:

```
{
  "sequenceStores": [
    {
      "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890",
      "creationTime": "2022-11-23T01:24:33.629Z",
```

```
        "id": "1234567890",
        "name": "my-seq-store"
    }
]
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListSequenceStores](#) la Referencia de AWS CLI comandos.

## list-shares

En el siguiente ejemplo de código, se muestra cómo usar `list-shares`.

### AWS CLI

Para enumerar las participaciones disponibles de un dato HealthOmics analítico

En el siguiente `list-shares` ejemplo, se enumeran todos los recursos compartidos que se han creado para el propietario de un recurso.

```
aws omics list-shares \
  --resource-owner SELF
```

Salida:

```
{
  "shares": [
    {
      "shareId": "595c1cbd-a008-4eca-a887-954d30c91c6e",
      "name": "myShare",
      "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
store_1",
      "principalSubscriber": "123456789012",
      "ownerId": "555555555555",
      "status": "PENDING"
    }
    {
      "shareId": "39b65d0d-4368-4a19-9814-b0e31d73c10a",
      "name": "myShare3456",

```

```

    "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
store_2",
    "principalSubscriber": "123456789012",
    "ownerId": "555555555555",
    "status": "ACTIVE"
  },
  {
    "shareId": "203152f5-eef9-459d-a4e0-a691668d44ef",
    "name": "myShare4",
    "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
store_3",
    "principalSubscriber": "123456789012",
    "ownerId": "555555555555",
    "status": "ACTIVE"
  }
]
}

```

Para obtener más información, consulte [Uso compartido entre cuentas](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [ListShares](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para obtener una lista de etiquetas

En el siguiente `list-tags-for-resource` ejemplo, se obtiene una lista de etiquetas para un flujo de trabajo con un identificador1234567.

```

aws omics list-tags-for-resource \
  --resource-arn arn:aws:omics:us-west-2:123456789012:workflow/1234567

```

Salida:

```

{
  "tags": {
    "department": "analytics"
  }
}

```



```
}  
}
```

Para obtener más información, consulte [los recursos de etiquetado en Amazon Omics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## list-variant-import-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-variant-import-jobs`.

### AWS CLI

Para obtener una lista de los trabajos de importación de variantes

En el siguiente `list-variant-import-jobs` ejemplo, se obtiene una lista de trabajos de importación de variantes.

```
aws omics list-variant-import-jobs
```

Salida:

```
{  
  "variantImportJobs": [  
    {  
      "creationTime": "2022-11-23T22:47:02.514002Z",  
      "destinationName": "my_var_store",  
      "id": "69cb65d6-xmpl-4a4a-9025-4565794b684e",  
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-  
serviceRole-W801XMPL7QZ",  
      "runLeftNormalization": false,  
      "status": "COMPLETED",  
      "updateTime": "2022-11-23T22:49:17.976597Z"  
    },  
    {  
      "creationTime": "2022-11-23T22:42:50.037812Z",  
      "destinationName": "my_var_store",  
      "id": "edd7b8ce-xmpl-47e2-bc99-258cac95a508",  
      "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-  
serviceRole-W801XMPL7QZ",  
    }  
  ]  
}
```

```
        "runLeftNormalization": false,  
        "status": "COMPLETED",  
        "updateTime": "2022-11-23T22:45:26.009880Z"  
    }  
]  
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListVariantImportJobs](#) la Referencia de AWS CLI comandos.

## list-variant-stores

En el siguiente ejemplo de código, se muestra cómo usar `list-variant-stores`.

### AWS CLI

Para obtener una lista de tiendas de variantes

En el siguiente `list-variant-stores` ejemplo, se obtiene una lista de tiendas de variantes.

```
aws omics list-variant-stores
```

Salida:

```
{  
  "variantStores": [  
    {  
      "creationTime": "2022-11-23T22:09:07.534499Z",  
      "id": "02dexmplcfdd",  
      "name": "my_var_store",  
      "reference": {  
        "referenceArn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/reference/1234567890"  
      },  
      "status": "CREATING",  
      "storeArn": "arn:aws:omics:us-west-2:123456789012:variantStore/my_var_store",  
      "storeSizeBytes": 0,  
      "updateTime": "2022-11-23T22:09:24.931711Z"  
    }  
  ]  
}
```

```
    },
    {
      "creationTime": "2022-09-23T23:00:09.140265Z",
      "id": "8777xmpl1a24",
      "name": "myvstore0",
      "status": "ACTIVE",
      "storeArn": "arn:aws:omics:us-west-2:123456789012:variantStore/
myvstore0",
      "storeSizeBytes": 0,
      "updateTime": "2022-09-23T23:03:26.013220Z"
    }
  ]
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListVariantStores](#) la Referencia de AWS CLI comandos.

## list-workflows

En el siguiente ejemplo de código, se muestra cómo usar `list-workflows`.

### AWS CLI

Para obtener una lista de flujos de trabajo

En el siguiente `list-workflows` ejemplo se obtiene una lista de flujos de trabajo.

```
aws omics list-workflows
```

Salida:

```
{
  "items": [
    {
      "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567",
      "creationTime": "2022-09-23T23:08:22.041227Z",
      "digest": "nSCNo/qMWFxmplXpUdokXJnwgne0axyyc2Y0xVxrJTE=",
      "id": "1234567",
      "name": "my-wkflow-0",
```

```

        "status": "ACTIVE",
        "type": "PRIVATE"
    },
    {
        "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567",
        "creationTime": "2022-11-30T22:33:16.225368Z",
        "digest":
"sha256:c54bxmpl742dcc26f7fa1f10e37550ddd8f251f418277c0a58e895b801ed28cf",
        "id": "1234567",
        "name": "cram-converter",
        "status": "ACTIVE",
        "type": "PRIVATE"
    }
]
}

```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [ListWorkflows](#) la Referencia de AWS CLI comandos.

## start-annotation-import-job

En el siguiente ejemplo de código, se muestra cómo usar `start-annotation-import-job`.

### AWS CLI

Para importar anotaciones

El siguiente `start-annotation-import-job` ejemplo importa anotaciones de Amazon S3.

```

aws omics start-annotation-import-job \
  --destination-name tsv_ann_store \
  --no-run-left-normalization \
  --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ \
  --items source=s3://omics-artifacts-01d6xmpl4e72dd32/targetedregions.bed.gz

```

Salida:

```

{
  "jobId": "984162c7-xmpl-4d23-ab47-286f7950bfbf"
}

```

```
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [StartAnnotationImportJob](#) la Referencia de AWS CLI comandos.

## **start-read-set-activation-job**

En el siguiente ejemplo de código, se muestra cómo usar `start-read-set-activation-job`.

### AWS CLI

Para activar un conjunto de lectura archivado

En el siguiente `start-read-set-activation-job` ejemplo, se activan dos conjuntos de lectura.

```
aws omics start-read-set-activation-job \  
  --sequence-store-id 1234567890 \  
  --sources readSetId=1234567890 readSetId=1234567890
```

Salida:

```
{  
  "creationTime": "2022-12-06T22:35:10.100Z",  
  "id": "1234567890",  
  "sequenceStoreId": "1234567890",  
  "status": "SUBMITTED"  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [StartReadSetActivationJob](#) la Referencia de AWS CLI comandos.

## **start-read-set-export-job**

En el siguiente ejemplo de código, se muestra cómo usar `start-read-set-export-job`.

## AWS CLI

Para exportar un conjunto de lectura

El siguiente `start-read-set-export-job` ejemplo exporta dos conjuntos de lectura a Amazon S3.

```
aws omics start-read-set-export-job \  
  --sequence-store-id 1234567890 \  
  --sources readSetId=1234567890 readSetId=1234567890 \  
  --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-  
W801XMPL7QZ  
\  
  --destination s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/
```

Salida:

```
{  
  "creationTime": "2022-12-06T22:37:18.612Z",  
  "destination": "s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/",  
  "id": "1234567890",  
  "sequenceStoreId": "1234567890",  
  "status": "SUBMITTED"  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [StartReadSetExportJob](#) la Referencia de AWS CLI comandos.

## **start-read-set-import-job**

En el siguiente ejemplo de código, se muestra cómo usar `start-read-set-import-job`.

## AWS CLI

Para importar un conjunto de lectura

En el siguiente `start-read-set-import-job` ejemplo, se importa un conjunto de lecturas.

```
aws omics start-read-set-import-job \  
  --sequence-store-id 1234567890 \  
  --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-  
W801XMPL7QZ
```

```
--role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ \
--sources file://readset-sources.json
```

readset-sources.json es un JSON documento con el siguiente contenido.

```
[
  {
    "sourceFiles":
    {
      "source1": "s3://omics-artifacts-01d6xmpl4e72dd32/
HG00100.chrom20.ILLUMINA.bwa.GBR.low_coverage.20101123.bam"
    },
    "sourceFileType": "BAM",
    "subjectId": "bam-subject",
    "sampleId": "bam-sample",
    "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890",
    "name": "HG00100"
  }
]
```

Salida:

```
{
  "creationTime": "2022-11-23T01:36:38.158Z",
  "id": "1234567890",
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ",
  "sequenceStoreId": "1234567890",
  "status": "SUBMITTED"
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [StartReadSetImportJob](#) la Referencia de AWS CLI comandos.

## start-reference-import-job

En el siguiente ejemplo de código, se muestra cómo usar `start-reference-import-job`.

## AWS CLI

Para importar un genoma de referencia

El siguiente `start-reference-import-job` ejemplo importa un genoma de referencia de Amazon S3.

```
aws omics start-reference-import-job \  
  --reference-store-id 1234567890 \  
  --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-  
W801XMPL7QZ \  
  --sources sourceFile=s3://omics-artifacts-01d6xmpl4e72dd32/  
Homo_sapiens_assembly38.fasta,name=assembly-38
```

Salida:

```
{  
  "creationTime": "2022-11-22T22:25:41.124Z",  
  "id": "1234567890",  
  "referenceStoreId": "1234567890",  
  "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-  
W801XMPL7QZ",  
  "status": "SUBMITTED"  
}
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [StartReferenceImportJob](#) la Referencia de AWS CLI comandos.

## start-run

En el siguiente ejemplo de código, se muestra cómo usar `start-run`.

## AWS CLI

Para ejecutar un flujo de trabajo

En el siguiente `start-run` ejemplo, se ejecuta un flujo de trabajo con un ID1234567.

```
aws omics start-run \  

```



```
--workflow-id 1234567 \  
--role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-  
WS01XMPL7QZ \  
--name 'cram-to-bam' \  
--output-uri s3://omics-artifacts-01d6xmpl4e72dd32/workflow-output/ \  
--run-group-id 1234567 \  
--priority 1 \  
--storage-capacity 10 \  
--log-level ALL \  
--parameters file://workflow-inputs.json
```

workflow-inputs.json es un documento con el siguiente contenido. JSON

```
{  
  "sample_name": "NA12878",  
  "input_cram": "s3://omics-artifacts-01d6xmpl4e72dd32/NA12878.cram",  
  "ref_dict": "s3://omics-artifacts-01d6xmpl4e72dd32/  
Homo_sapiens_assembly38.dict",  
  "ref_fasta": "s3://omics-artifacts-01d6xmpl4e72dd32/  
Homo_sapiens_assembly38.fasta",  
  "ref_fasta_index": "omics-artifacts-01d6xmpl4e72dd32/  
Homo_sapiens_assembly38.fasta.fai"  
}
```

Salida:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",  
  "id": "1234567",  
  "status": "PENDING",  
  "tags": {}  
}
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

Para cargar archivos fuente de Amazon Omics

También puede cargar archivos fuente desde el almacenamiento de Amazon Omics mediante un servicio específico URIs. En el siguiente ejemplo, el archivo workflow-inputs.json utiliza Amazon URIs Omics para leer fuentes genómicas de conjuntos y de referencia.

```
{
  "sample_name": "NA12878",
  "input_cram": "omics://123456789012.storage.us-west-2.amazonaws.com/1234567890/
readSet/1234567890/source1",
  "ref_dict": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.dict",
  "ref_fasta": "omics://123456789012.storage.us-west-2.amazonaws.com/1234567890/
reference/1234567890",
  "ref_fasta_index": "omics://123456789012.storage.us-
west-2.amazonaws.com/1234567890/reference/1234567890/index"
}
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [StartRun](#) la Referencia de AWS CLI comandos.

## start-variant-import-job

En el siguiente ejemplo de código, se muestra cómo usar `start-variant-import-job`.

### AWS CLI

Para importar un archivo de variantes

En el siguiente `start-variant-import-job` ejemplo, se importa un archivo de variantes de VCF formato.

```
aws omics start-variant-import-job \
  --destination-name my_var_store \
  --no-run-left-normalization \
  --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ \
  --items source=s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.known_indels.vcf.gz
```

Salida:

```
{
  "jobId": "edd7b8ce-xmpl-47e2-bc99-258cac95a508"
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [StartVariantImportJob](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar un recurso

En el siguiente `tag-resource` ejemplo, se agrega una `department` etiqueta a un flujo de trabajo con un identificador `1234567`.

```
aws omics tag-resource \  
  --resource-arn arn:aws:omics:us-west-2:123456789012:workflow/1234567 \  
  --tags department=analytics
```

Para obtener más información, consulte [los recursos de etiquetado en Amazon Omics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar una etiqueta de un recurso

En el siguiente `untag-resource` ejemplo, se elimina la `department` etiqueta de un flujo de trabajo.

```
aws omics untag-resource \  
  --resource-arn arn:aws:omics:us-west-2:123456789012:workflow/1234567 \  
  --tag-keys department
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-annotation-store

En el siguiente ejemplo de código, se muestra cómo usar `update-annotation-store`.

### AWS CLI

Para actualizar un almacén de anotaciones

En el siguiente `update-annotation-store` ejemplo se actualiza la descripción de un almacén de anotaciones denominado `my_vcf_store`

```
aws omics update-annotation-store \  
  --name my_vcf_store \  
  --description "VCF annotation store"
```

Salida:

```
{  
  "creationTime": "2022-12-05T18:00:56.101860Z",  
  "description": "VCF annotation store",  
  "id": "bd6axmpl2444",  
  "name": "my_vcf_store",  
  "reference": {  
    "referenceArn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/reference/1234567890"  
  },  
  "status": "ACTIVE",  
  "storeFormat": "VCF",  
  "updateTime": "2022-12-05T18:13:16.100051Z"  
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [UpdateAnnotationStore](#) la Referencia de AWS CLI comandos.

## update-run-group

En el siguiente ejemplo de código, se muestra cómo usar `update-run-group`.

### AWS CLI

Para actualizar un grupo de carreras

En el siguiente `update-run-group` ejemplo, se actualiza la configuración de un grupo de carreras con un identificador1234567.

```
aws omics update-run-group \  
  --id 1234567 \  
  --max-cpus 10
```

Salida:

```
{  
  "arn": "arn:aws:omics:us-west-2:123456789012:runGroup/1234567",  
  "creationTime": "2022-12-01T00:58:42.915219Z",  
  "id": "1234567",  
  "maxCpus": 10,  
  "maxDuration": 600,  
  "name": "cram-convert",  
  "tags": {}  
}
```

Para obtener más información, consulte [Omics Workflows](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [UpdateRunGroup](#) la Referencia de AWS CLI comandos.

## update-variant-store

En el siguiente ejemplo de código, se muestra cómo usar `update-variant-store`.

### AWS CLI

Para actualizar un almacén de variantes

En el siguiente `update-variant-store` ejemplo, se actualiza la descripción de un almacén de variantes denominadomy\_var\_store.

```
aws omics update-variant-store \  
  --name my_var_store \  
  --description "variant store"
```

Salida:

```
{  
  "creationTime": "2022-11-23T22:09:07.534499Z",  
  "description": "variant store",  
  "id": "02dexmplcfdd",  
  "name": "my_var_store",  
  "reference": {  
    "referenceArn": "arn:aws:omics:us-  
west-2:123456789012:referenceStore/1234567890/reference/1234567890"  
  },  
  "status": "ACTIVE",  
  "updateTime": "2022-12-05T18:23:37.686402Z"  
}
```

Para obtener más información, consulte [Omics Analytics](#) en la Guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [UpdateVariantStore](#) la Referencia de AWS CLI comandos.

## update-workflow

En el siguiente ejemplo de código, se muestra cómo usar `update-workflow`.

### AWS CLI

Para actualizar un flujo de trabajo

En el siguiente `update-workflow` ejemplo, se actualiza la descripción de un flujo de trabajo con un ID1234567.

```
aws omics update-workflow \  
  --id 1234567 \  
  --description "copy workflow"
```

Para obtener más información, consulte [Omics Storage](#) en la guía para desarrolladores de Amazon Omics.

- Para API obtener más información, consulte [UpdateWorkflow](#) la Referencia de AWS CLI comandos.

## upload-read-set-part

En el siguiente ejemplo de código, se muestra cómo usar `upload-read-set-part`.

### AWS CLI

Para cargar una parte del conjunto de lectura.

En el siguiente `upload-read-set-part` ejemplo, se carga una parte específica de un conjunto de lecturas.

```
aws omics upload-read-set-part \  
  --sequence-store-id 0123456789 \  
  --upload-id 1122334455 \  
  --part-source SOURCE1 \  
  --part-number 1 \  
  --payload /path/to/file/read_1_part_1.fastq.gz
```

Salida:

```
{  
  "checksum": "984979b9928ae8d8622286c4a9cd8e99d964a22d59ed0f5722e1733eb280e635"  
}
```

Para obtener más información, consulte [Carga directa a un almacén de secuencias](#) en la Guía del AWS HealthOmics usuario.

- Para API obtener más información, consulte [UploadReadSetPart](#) la Referencia de AWS CLI comandos.

## IA Mejores ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with IAM.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

Acciones

### **add-client-id-to-open-id-connect-provider**

En el siguiente ejemplo de código, se muestra cómo usar `add-client-id-to-open-id-connect-provider`.

AWS CLI

Para añadir un ID de cliente (audiencia) a un proveedor de Open-ID Connect (OIDC)

El siguiente `add-client-id-to-open-id-connect-provider` comando agrega el ID de cliente `my-application-ID` al OIDC proveedor nombrado `server.example.com`.

```
aws iam add-client-id-to-open-id-connect-provider \  
  --client-id my-application-ID \  
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/  
server.example.com
```

Este comando no genera ninguna salida.

Para crear un OIDC proveedor, utilice el `create-open-id-connect-provider` comando.

Para obtener más información, consulte [Creación de proveedores de identidad de OpenID Connect \(OIDC\)](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [AddClientIdToOpenIdConnectProvider](#) la Referencia de AWS CLI comandos.



## add-role-to-instance-profile

En el siguiente ejemplo de código, se muestra cómo usar `add-role-to-instance-profile`.

### AWS CLI

Cómo añadir un rol a un perfil de instancia

El siguiente comando `add-role-to-instance-profile` agrega el rol nombrado `S3Access` al perfil de instancia llamado `Webserver`.

```
aws iam add-role-to-instance-profile \  
  --role-name S3Access \  
  --instance-profile-name Webserver
```

Este comando no genera ninguna salida.

Para crear un perfil de instancia, utilice el comando `create-instance-profile`.

Para obtener más información, consulte [Uso de un IAM rol para conceder permisos a aplicaciones que se ejecutan en EC2 instancias de Amazon](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [AddRoleToInstanceProfile](#) la Referencia de AWS CLI comandos.

## add-user-to-group

En el siguiente ejemplo de código, se muestra cómo usar `add-user-to-group`.

### AWS CLI

Para añadir un usuario a un IAM grupo

El siguiente `add-user-to-group` comando agrega un IAM usuario nombrado `Bob` al IAM grupo denominado `Admins`.

```
aws iam add-user-to-group \  
  --user-name Bob \  
  --group-name Admins
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir y eliminar usuarios de un grupo IAM de usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [AddUserToGroup](#) la Referencia de AWS CLI comandos.

## attach-group-policy

En el siguiente ejemplo de código, se muestra cómo usar `attach-group-policy`.

### AWS CLI

Para adjuntar una política gestionada a un IAM grupo

El siguiente `attach-group-policy` comando adjunta la política AWS administrada nombrada `ReadOnlyAccess` al IAM grupo denominado `Finance`.

```
aws iam attach-group-policy \  
  --policy-arn arn:aws:iam::aws:policy/ReadOnlyAccess \  
  --group-name Finance
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Políticas administradas y políticas integradas en la Guía del AWS IAM usuario](#).

- Para API obtener más información, consulte [AttachGroupPolicy](#) la Referencia de AWS CLI comandos.

## attach-role-policy

En el siguiente ejemplo de código, se muestra cómo usar `attach-role-policy`.

### AWS CLI

Para adjuntar una política gestionada a un IAM rol

El siguiente `attach-role-policy` comando adjunta la política AWS administrada nombrada `ReadOnlyAccess` al IAM rol nombrado `ReadOnlyRole`.

```
aws iam attach-role-policy \  
  --policy-arn arn:aws:iam::aws:policy/ReadOnlyAccess \  
  --role-name ReadOnlyRole
```

```
--role-name ReadOnlyRole
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Políticas administradas y políticas integradas en la Guía del AWS IAM](#) usuario.

- Para API obtener más información, consulte [AttachRolePolicy](#) la Referencia de AWS CLI comandos.

## attach-user-policy

En el siguiente ejemplo de código, se muestra cómo usar `attach-user-policy`.

### AWS CLI

Para adjuntar una política gestionada a un IAM usuario

El siguiente `attach-user-policy` comando adjunta la política AWS administrada nombrada `AdministratorAccess` al IAM usuario nombrado `Alice`.

```
aws iam attach-user-policy \  
  --policy-arn arn:aws:iam::aws:policy/AdministratorAccess \  
  --user-name Alice
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Políticas administradas y políticas integradas en la Guía del AWS IAM](#) usuario.

- Para API obtener más información, consulte [AttachUserPolicy](#) la Referencia de AWS CLI comandos.

## change-password

En el siguiente ejemplo de código, se muestra cómo usar `change-password`.

### AWS CLI

Para cambiar la contraseña de su IAM usuario

Para cambiar la contraseña de su IAM usuario, le recomendamos que utilice el `--cli-input-json` parámetro para pasar un JSON archivo que contenga las contraseñas antiguas y nuevas.

Con este método, puede utilizar contraseñas seguras con caracteres no alfanuméricos. Puede resultar difícil utilizar contraseñas con caracteres no alfanuméricos al pasarlas como parámetros de la línea de comandos. Para usar el parámetro `--cli-input-json`, comience por usar el comando `change-password` con el parámetro `--generate-cli-skeleton`, como en el siguiente ejemplo.

```
aws iam change-password \  
  --generate-cli-skeleton > change-password.json
```

El comando anterior crea un JSON archivo llamado `change-password.json` que puedes usar para rellenar tus contraseñas antiguas y nuevas. Por ejemplo, el rol podría tener el siguiente aspecto.

```
{  
  "OldPassword": "3s0K_;xh4~8XXI",  
  "NewPassword": "]35d/{pB9Fo9wJ"  
}
```

A continuación, para cambiar la contraseña, vuelva a utilizar el `change-password` comando, esta vez pasando el `--cli-input-json` parámetro para especificar el archivo. JSON El `change-password` comando siguiente usa el `--cli-input-json` parámetro con un JSON archivo llamado `change-password.json`.

```
aws iam change-password \  
  --cli-input-json file://change-password.json
```

Este comando no genera ninguna salida.

Solo los usuarios pueden llamar a este comando. IAM Si se llama a este comando con las credenciales de la AWS cuenta (root), el comando devuelve un `InvalidUserType` error.

Para obtener más información, consulte [Cómo un IAM usuario cambia su propia contraseña](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ChangePassword](#) la Referencia de AWS CLI comandos.

## create-access-key

En el siguiente ejemplo de código, se muestra cómo usar `create-access-key`.

## AWS CLI

Para crear una clave de acceso para un IAM usuario

El siguiente `create-access-key` comando crea una clave de acceso (identificador de clave de acceso y clave de acceso secreta) para el IAM usuario nombrado Bob.

```
aws iam create-access-key \  
  --user-name Bob
```

Salida:

```
{  
  "AccessKey": {  
    "UserName": "Bob",  
    "Status": "Active",  
    "CreateDate": "2015-03-09T18:39:23.411Z",  
    "SecretAccessKey": "wJa1rXUtnFEMI/K7MDENG/bPxRfiCYzEXAMPLEKEY",  
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE"  
  }  
}
```

Almacene la clave de acceso secreta en un lugar seguro. Si se pierde, no se puede recuperar y debe crear una nueva clave de acceso.

Para obtener más información, consulte [Administrar las claves de acceso de IAM los usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [CreateAccessKey](#) la Referencia de AWS CLI comandos.

## **create-account-alias**

En el siguiente ejemplo de código, se muestra cómo usar `create-account-alias`.

## AWS CLI

Cómo crear un alias de una cuenta

El siguiente `create-account-alias` comando crea el alias `examplecorp` de su AWS cuenta.

```
aws iam create-account-alias \  
  --account-alias examplecorp
```

Este comando no genera ninguna salida.

Para obtener más información, consulta el [identificador de AWS cuenta y su alias](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [CreateAccountAlias](#) la Referencia de AWS CLI comandos.

## create-group

En el siguiente ejemplo de código, se muestra cómo usar create-group.

### AWS CLI

Para crear un IAM grupo

El siguiente create-group comando crea un IAM grupo denominado Admins.

```
aws iam create-group \  
  --group-name Admins
```

Salida:

```
{  
  "Group": {  
    "Path": "/",  
    "CreateDate": "2015-03-09T20:30:24.940Z",  
    "GroupId": "AIDGPMS9R04H3FEXAMPLE",  
    "Arn": "arn:aws:iam::123456789012:group/Admins",  
    "GroupName": "Admins"  
  }  
}
```

Para obtener más información, consulte [Creación IAM de grupos de usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [CreateGroup](#) la Referencia de AWS CLI comandos.

## create-instance-profile

En el siguiente ejemplo de código, se muestra cómo usar `create-instance-profile`.

### AWS CLI

Cómo crear un perfil de instancia

El siguiente comando `create-instance-profile` crea un perfil de instancia denominado `Webserver`.

```
aws iam create-instance-profile \  
  --instance-profile-name Webserver
```

Salida:

```
{  
  "InstanceProfile": {  
    "InstanceProfileId": "AIPAJMBC7DLSPEXAMPLE",  
    "Roles": [],  
    "CreateDate": "2015-03-09T20:33:19.626Z",  
    "InstanceProfileName": "Webserver",  
    "Path": "/",  
    "Arn": "arn:aws:iam::123456789012:instance-profile/Webserver"  
  }  
}
```

Para añadir un rol a un perfil de instancia, utilice el comando `add-role-to-instance-profile`.

Para obtener más información, consulte [Uso de un IAM rol para conceder permisos a aplicaciones que se ejecutan en EC2 instancias de Amazon](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [CreateInstanceProfile](#) la Referencia de AWS CLI comandos.

## create-login-profile

En el siguiente ejemplo de código, se muestra cómo usar `create-login-profile`.

### AWS CLI

Para crear una contraseña para un IAM usuario

Para crear una contraseña para un IAM usuario, se recomienda utilizar el `--cli-input-json` parámetro para pasar un JSON archivo que contenga la contraseña. Con este método, puede crear una contraseña segura con caracteres no alfanuméricos. Puede resultar difícil crear una contraseña con caracteres no alfanuméricos si la pasa como parámetro de la línea de comandos.

Para usar el parámetro `--cli-input-json`, comience por usar el comando `create-login-profile` con el parámetro `--generate-cli-skeleton`, como en el siguiente ejemplo.

```
aws iam create-login-profile \  
  --generate-cli-skeleton > create-login-profile.json
```

El comando anterior crea un JSON archivo denominado `create-login-profile.json` que puede utilizar para rellenar la información de un `create-login-profile` comando posterior. Por ejemplo:

```
{  
  "UserName": "Bob",  
  "Password": "&1-3a6u:RA0djs",  
  "PasswordResetRequired": true  
}
```

A continuación, para crear una contraseña para un IAM usuario, vuelva a utilizar el `create-login-profile` comando, esta vez pasando el `--cli-input-json` parámetro para especificar el JSON archivo. El `create-login-profile` comando siguiente usa el `--cli-input-json` parámetro con un JSON archivo denominado `create-login-profile.json`.

```
aws iam create-login-profile \  
  --cli-input-json file://create-login-profile.json
```

Salida:

```
{  
  "LoginProfile": {  
    "UserName": "Bob",  
    "CreateDate": "2015-03-10T20:55:40.274Z",  
    "PasswordResetRequired": true  
  }  
}
```



Si la nueva contraseña infringe la política de contraseñas de la cuenta, el comando devuelve el error `PasswordPolicyViolation`.

Para cambiar la contraseña de un usuario que ya posee una, utilice `update-login-profile`. Para establecer una política de contraseñas para la cuenta, utilice el comando `update-account-password-policy`.

Si la política de contraseñas de la cuenta lo permite, IAM los usuarios pueden cambiar sus propias contraseñas mediante el `change-password` comando.

Para obtener más información, consulte [Administrar las contraseñas de IAM los usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [CreateLoginProfile](#) la Referencia de AWS CLI comandos.

## **create-open-id-connect-provider**

En el siguiente ejemplo de código, se muestra cómo usar `create-open-id-connect-provider`.

### AWS CLI

Para crear un proveedor de OpenID Connect (OIDC)

Para crear un proveedor de OpenID Connect (OIDC), se recomienda utilizar el `--cli-input-json` parámetro para pasar un JSON archivo que contenga los parámetros necesarios. Al crear un OIDC proveedor, debe pasar el URL del proveedor y el URL debe empezar `https://` por. Puede resultar difícil pasarlo URL como parámetro de línea de comandos, ya que los dos puntos (:) y la barra inclinada (/) tienen un significado especial en algunos entornos de línea de comandos. El uso del parámetro `--cli-input-json` evita esta limitación.

Para usar el parámetro `--cli-input-json`, comience por usar el comando `create-open-id-connect-provider` con el parámetro `--generate-cli-skeleton`, como en el siguiente ejemplo.

```
aws iam create-open-id-connect-provider \  
  --generate-cli-skeleton > create-open-id-connect-provider.json
```

El comando anterior crea un JSON archivo denominado `create-open-id-connect-provider.json` que puede utilizar para rellenar la información de un comando posterior. `create-open-id-connect-provider` Por ejemplo:

```
{
  "Url": "https://server.example.com",
  "ClientIDList": [
    "example-application-ID"
  ],
  "ThumbprintList": [
    "c3768084dfb3d2b68b7897bf5f565da8eEXAMPLE"
  ]
}
```

A continuación, para crear el proveedor OpenID Connect (OIDC), vuelva a utilizar el `create-open-id-connect-provider` comando, esta vez pasando el `--cli-input-json` parámetro para especificar el archivoJSON. El `create-open-id-connect-provider` comando siguiente usa el `--cli-input-json` parámetro con un JSON archivo llamado `create-open-id-connect-provider.json`.

```
aws iam create-open-id-connect-provider \
  --cli-input-json file://create-open-id-connect-provider.json
```

Salida:

```
{
  "OpenIDConnectProviderArn": "arn:aws:iam::123456789012:oidc-provider/
server.example.com"
}
```

Para obtener más información sobre OIDC los proveedores, consulte [Creación de proveedores de identidad de OpenID Connect \(OIDC\)](#) en la Guía del AWS IAM usuario.

Para obtener más información sobre la obtención de huellas digitales para un OIDC proveedor, consulte [Obtención de la huella digital para un proveedor de identidades de OpenID Connect en la Guía del usuario](#).AWS IAM

- Para API obtener más información, consulte la Referencia de [CreateOpenIdConnectProvider](#) comandos AWS CLI .

## **create-policy-version**

En el siguiente ejemplo de código, se muestra cómo usar `create-policy-version`.

## AWS CLI

Cómo crear una nueva versión de la política administrada

Este ejemplo crea una nueva v2 versión de la IAM política cuyo ARN es `arn:aws:iam::123456789012:policy/MyPolicy` y la convierte en la versión predeterminada.

```
aws iam create-policy-version \  
  --policy-arn arn:aws:iam::123456789012:policy/MyPolicy \  
  --policy-document file://NewPolicyVersion.json \  
  --set-as-default
```

Salida:

```
{  
  "PolicyVersion": {  
    "CreateDate": "2015-06-16T18:56:03.721Z",  
    "VersionId": "v2",  
    "IsDefaultVersion": true  
  }  
}
```

Para obtener más información, consulte [IAM las políticas de control de versiones](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [CreatePolicyVersion](#) la Referencia de AWS CLI comandos.

## create-policy

En el siguiente ejemplo de código, se muestra cómo usar `create-policy`.

### AWS CLI

Ejemplo 1: cómo crear una política administrada por el cliente

El siguiente comando crea una política administrada por el cliente denominada `my-policy`.

```
aws iam create-policy \  
  --policy-name my-policy \  
  --policy-document file://my-policy.json
```

```
--policy-document file://policy
```

El archivo `policy` es un JSON documento de la carpeta actual que concede acceso de solo lectura a la `shared` carpeta de un bucket de Amazon S3 denominado `my-bucket`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:Get*",
        "s3:List*"
      ],
      "Resource": [
        "arn:aws:s3:::my-bucket/shared/*"
      ]
    }
  ]
}
```

Salida:

```
{
  "Policy": {
    "PolicyName": "my-policy",
    "CreateDate": "2015-06-01T19:31:18.620Z",
    "AttachmentCount": 0,
    "IsAttachable": true,
    "PolicyId": "ZXR6A36LTYANPAI7NJ5UV",
    "DefaultVersionId": "v1",
    "Path": "/",
    "Arn": "arn:aws:iam::0123456789012:policy/my-policy",
    "UpdateDate": "2015-06-01T19:31:18.620Z"
  }
}
```

Para obtener más información sobre el uso de archivos como entrada para los parámetros de cadena, consulte [Especificar los valores de los parámetros AWS CLI](#) en la Guía del AWS CLI usuario.

Ejemplo 2: cómo crear una política administrada por el cliente con una descripción

El siguiente comando crea una política administrada por el cliente denominada `my-policy` con una descripción inmutable:

```
aws iam create-policy \  
  --policy-name my-policy \  
  --policy-document file://policy.json \  
  --description "This policy grants access to all Put, Get, and List actions for my-bucket"
```

El archivo `policy.json` es un JSON documento de la carpeta actual que permite el acceso a todas las acciones Put, List y Get de un bucket de Amazon S3 denominado `my-bucket`.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "s3:ListBucket*",  
        "s3:PutBucket*",  
        "s3:GetBucket*"  
      ],  
      "Resource": [  
        "arn:aws:s3:::my-bucket"  
      ]  
    }  
  ]  
}
```

Salida:

```
{  
  "Policy": {  
    "PolicyName": "my-policy",  
    "PolicyId": "ANPAWGSUGIDPEXAMPLE",  
    "Arn": "arn:aws:iam::123456789012:policy/my-policy",  
    "Path": "/",  
    "DefaultVersionId": "v1",  
    "AttachmentCount": 0,  
    "PermissionsBoundaryUsageCount": 0,  
    "IsAttachable": true,  
    "CreateDate": "2023-05-24T22:38:47+00:00",
```

```

    "UpdateDate": "2023-05-24T22:38:47+00:00"
  }
}

```

Para obtener más información sobre las políticas basadas en la identidad, consulte las políticas basadas en la [identidad y las políticas basadas en recursos en la Guía del usuario](#).AWS IAM

### Ejemplo 3: cómo crear una política administrada por el cliente con etiquetas

El siguiente comando crea una política administrada por el cliente denominada `my-policy` con etiquetas. En este ejemplo se utiliza el indicador de `--tags` parámetros con las siguientes etiquetas con formato: JSON `'{"Key": "Department", "Value": "Accounting"}'` `'{"Key": "Location", "Value": "Seattle"}'` Alternativamente, el indicador `--tags` se puede usar con etiquetas en formato abreviado: `'Key=Department,Value=Accounting Key=Location,Value=Seattle'`.

```

aws iam create-policy \
  --policy-name my-policy \
  --policy-document file://policy.json \
  --tags '{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location",
"Value": "Seattle"}'

```

El archivo `policy.json` es un JSON documento de la carpeta actual que permite el acceso a todas las acciones Put, List y Get de un bucket de Amazon S3 denominado `my-bucket`.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket*",
        "s3:PutBucket*",
        "s3:GetBucket*"
      ],
      "Resource": [
        "arn:aws:s3:::my-bucket"
      ]
    }
  ]
}

```

**Salida:**

```
{
  "Policy": {
    "PolicyName": "my-policy",
    "PolicyId": "ANPAWGSUGIDPEXAMPLE",
    "Arn": "arn:aws:iam::12345678012:policy/my-policy",
    "Path": "/",
    "DefaultVersionId": "v1",
    "AttachmentCount": 0,
    "PermissionsBoundaryUsageCount": 0,
    "IsAttachable": true,
    "CreateDate": "2023-05-24T23:16:39+00:00",
    "UpdateDate": "2023-05-24T23:16:39+00:00",
    "Tags": [
      {
        "Key": "Department",
        "Value": "Accounting"
      },
      {
        "Key": "Location",
        "Value": "Seattle"
      }
    ]
  }
}
```

Para obtener más información sobre las políticas de etiquetado, consulte [Etiquetar las políticas administradas por los clientes](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte la Referencia [CreatePolicy](#) de AWS CLI comandos.

**create-role**

En el siguiente ejemplo de código, se muestra cómo usar `create-role`.

**AWS CLI**

Ejemplo 1: Para crear un IAM rol

El siguiente comando `create-role` crea un rol denominado `Test-Role` y le asocia una política de confianza.

```
aws iam create-role \  
  --role-name Test-Role \  
  --assume-role-policy-document file://Test-Role-Trust-Policy.json
```

Salida:

```
{  
  "Role": {  
    "AssumeRolePolicyDocument": "<URL-encoded-JSON>",  
    "RoleId": "AKIAIOSFODNN7EXAMPLE",  
    "CreateDate": "2013-06-07T20:43:32.821Z",  
    "RoleName": "Test-Role",  
    "Path": "/",  
    "Arn": "arn:aws:iam::123456789012:role/Test-Role"  
  }  
}
```

La política de confianza se define como un JSON documento del archivo test-role-trust-policy.json. (El nombre y la extensión del archivo no son significativos). La política de confianza debe especificar una entidad principal.

Utilice el comando `put-role-policy` para asociar una política de permisos a un rol.

[Para obtener más información, consulte Creación de funciones en la Guía del usuario. IAM AWS IAM](#)

Ejemplo 2: Para crear un IAM rol con una duración de sesión máxima especificada

El siguiente comando `create-role` crea un rol denominado `Test-Role` y establece una duración máxima de sesión de 7200 segundos (2 horas).

```
aws iam create-role \  
  --role-name Test-Role \  
  --assume-role-policy-document file://Test-Role-Trust-Policy.json \  
  --max-session-duration 7200
```

Salida:

```
{  
  "Role": {
```



```

    "Path": "/",
    "RoleName": "Test-Role",
    "RoleId": "AKIAIOSFODNN7EXAMPLE",
    "Arn": "arn:aws:iam::12345678012:role/Test-Role",
    "CreateDate": "2023-05-24T23:50:25+00:00",
    "AssumeRolePolicyDocument": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Sid": "Statement1",
          "Effect": "Allow",
          "Principal": {
            "AWS": "arn:aws:iam::12345678012:root"
          },
          "Action": "sts:AssumeRole"
        }
      ]
    }
  }
}

```

Para obtener más información, consulte [Modificación de la duración máxima de sesión de un rol \(AWS API\)](#) en la Guía del AWS IAM usuario.

Ejemplo 3: Para crear un IAM rol con etiquetas

El siguiente comando crea un IAM rol Test-Role con etiquetas. En este ejemplo, se utiliza el indicador de `--tags` parámetros con las siguientes etiquetas con JSON formato: `'{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location", "Value": "Seattle"}'` Alternativamente, el indicador `--tags` se puede usar con etiquetas en formato abreviado: `'Key=Department,Value=Accounting Key=Location,Value=Seattle'`.

```

aws iam create-role \
  --role-name Test-Role \
  --assume-role-policy-document file://Test-Role-Trust-Policy.json \
  --tags '{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location",
"Value": "Seattle"}'

```

Salida:

```

{
  "Role": {

```

```
"Path": "/",
"RoleName": "Test-Role",
"RoleId": "AKIAIOSFODNN7EXAMPLE",
"Arn": "arn:aws:iam::123456789012:role/Test-Role",
"CreateDate": "2023-05-25T23:29:41+00:00",
"AssumeRolePolicyDocument": {
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Statement1",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:root"
      },
      "Action": "sts:AssumeRole"
    }
  ]
},
"Tags": [
  {
    "Key": "Department",
    "Value": "Accounting"
  },
  {
    "Key": "Location",
    "Value": "Seattle"
  }
]
}
```

Para obtener más información, consulte [Etiquetado de IAM roles](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [CreateRole](#) la Referencia de AWS CLI comandos.

## **create-saml-provider**

En el siguiente ejemplo de código, se muestra cómo usar `create-saml-provider`.

### AWS CLI

Para crear un SAML proveedor

En este ejemplo se crea un nuevo SAML proveedor en IAM named `MySAMLProvider`. Se describe en el documento de SAML metadatos que se encuentra en el archivo `SAMLMetaData.xml`.

```
aws iam create-saml-provider \  
  --saml-metadata-document file://SAMLMetaData.xml \  
  --name MySAMLProvider
```

Salida:

```
{  
  "SAMLProviderArn": "arn:aws:iam::123456789012:saml-provider/MySAMLProvider"  
}
```

Para obtener más información, consulte [Creación de proveedores de IAM SAML identidad](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [C reateSAMLProvider](#) en la Referencia de AWS CLI comandos.

## create-service-linked-role

En el siguiente ejemplo de código, se muestra cómo usar `create-service-linked-role`.

### AWS CLI

Cómo crear un rol vinculado a un servicio

En el siguiente `create-service-linked-role` ejemplo, se crea un rol vinculado a un servicio para el AWS servicio especificado y se adjunta la descripción especificada.

```
aws iam create-service-linked-role \  
  --aws-service-name lex.amazonaws.com \  
  --description "My service-linked role to support Lex"
```

Salida:

```
{  
  "Role": {  
    "Path": "/aws-service-role/lex.amazonaws.com/",  
    "RoleName": "AWSServiceRoleForLexBots",
```

```

    "RoleId": "ARO1234567890EXAMPLE",
    "Arn": "arn:aws:iam::1234567890:role/aws-service-role/lex.amazonaws.com/
AWSServiceRoleForLexBots",
    "CreateDate": "2019-04-17T20:34:14+00:00",
    "AssumeRolePolicyDocument": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Action": [
            "sts:AssumeRole"
          ],
          "Effect": "Allow",
          "Principal": {
            "Service": [
              "lex.amazonaws.com"
            ]
          }
        }
      ]
    }
  }
}

```

Para obtener más información, consulte [Uso de funciones vinculadas a servicios en la Guía del usuario.AWS IAM](#)

- Para API obtener más información, consulte la Referencia [CreateServiceLinkedRole](#) de AWS CLI comandos.

## create-service-specific-credential

En el siguiente ejemplo de código, se muestra cómo usar `create-service-specific-credential`.

### AWS CLI

Cree un conjunto de credenciales específicas de un servicio para un usuario

El siguiente `create-service-specific-credential` ejemplo crea un nombre de usuario y una contraseña que se pueden usar para acceder únicamente al servicio configurado.

```

aws iam create-service-specific-credential \
  --user-name sofia \

```

```
--service-name codecommit.amazonaws.com
```

Salida:

```
{
  "ServiceSpecificCredential": {
    "CreateDate": "2019-04-18T20:45:36+00:00",
    "ServiceName": "codecommit.amazonaws.com",
    "ServiceUserName": "sofia-at-123456789012",
    "ServicePassword": "k1zPZM6uVxMQ3oxqgoY1NuJPyRTZ1vREs76zTQE3eJk=",
    "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",
    "UserName": "sofia",
    "Status": "Active"
  }
}
```

Para obtener más información, consulta [Crear credenciales de Git para HTTPS las conexiones CodeCommit](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [CreateServiceSpecificCredential](#) la Referencia de AWS CLI comandos.

## create-user

En el siguiente ejemplo de código, se muestra cómo usar `create-user`.

### AWS CLI

Ejemplo 1: Para crear un IAM usuario

El siguiente `create-user` comando crea un IAM usuario con el nombre Bob de la cuenta actual.

```
aws iam create-user \  
  --user-name Bob
```

Salida:

```
{
  "User": {
    "UserName": "Bob",
    "Path": "/",
    "CreateDate": "2023-06-08T03:20:41.270Z",
```

```
    "UserId": "AIDAIOSFODNN7EXAMPLE",
    "Arn": "arn:aws:iam::123456789012:user/Bob"
  }
}
```

Para obtener más información, consulte [Crear un IAM usuario en su AWS cuenta](#) en la Guía del AWS IAM usuario.

Ejemplo 2: Para crear un IAM usuario en una ruta específica

El siguiente `create-user` comando crea un IAM usuario con el nombre Bob de la ruta especificada.

```
aws iam create-user \
  --user-name Bob \
  --path /division_abc/subdivision_xyz/
```

Salida:

```
{
  "User": {
    "Path": "/division_abc/subdivision_xyz/",
    "UserName": "Bob",
    "UserId": "AIDAIOSFODNN7EXAMPLE",
    "Arn": "arn:aws:iam::12345678012:user/division_abc/subdivision_xyz/Bob",
    "CreateDate": "2023-05-24T18:20:17+00:00"
  }
}
```

Para obtener más información, consulte los [IAM identificadores](#) en la Guía del AWS IAM usuario.

Ejemplo 3: Crear un IAM usuario con etiquetas

El siguiente `create-user` comando crea un IAM usuario nombrado Bob con etiquetas. En este ejemplo, se utiliza el indicador de `--tags` parámetros con las siguientes etiquetas con JSON formato: `'{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location", "Value": "Seattle"}'` Alternativamente, el indicador `--tags` se puede usar con etiquetas en formato abreviado: `'Key=Department, Value=Accounting Key=Location, Value=Seattle'`.

```
aws iam create-user \
  --user-name Bob \
```

```
--tags '{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location",  
"Value": "Seattle"}'
```

Salida:

```
{  
  "User": {  
    "Path": "/",  
    "UserName": "Bob",  
    "UserId": "AIDAIOSFODNN7EXAMPLE",  
    "Arn": "arn:aws:iam::12345678012:user/Bob",  
    "CreateDate": "2023-05-25T17:14:21+00:00",  
    "Tags": [  
      {  
        "Key": "Department",  
        "Value": "Accounting"  
      },  
      {  
        "Key": "Location",  
        "Value": "Seattle"  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [Etiquetado de IAM usuarios](#) en la Guía del AWS IAM usuario.

Ejemplo 3: Para crear un IAM usuario con un límite de permisos establecido

El siguiente `create-user` comando crea un nombre IAM de usuario Bob con el límite de permisos de `AmazonS3FullAccess`.

```
aws iam create-user \  
  --user-name Bob \  
  --permissions-boundary arn:aws:iam::aws:policy/AmazonS3FullAccess
```

Salida:

```
{  
  "User": {  
    "Path": "/",
```

```

    "UserName": "Bob",
    "UserId": "AIDAIOSFODNN7EXAMPLE",
    "Arn": "arn:aws:iam::12345678012:user/Bob",
    "CreateDate": "2023-05-24T17:50:53+00:00",
    "PermissionsBoundary": {
      "PermissionsBoundaryType": "Policy",
      "PermissionsBoundaryArn": "arn:aws:iam::aws:policy/AmazonS3FullAccess"
    }
  }
}

```

Para obtener más información, consulte [los límites de los permisos para IAM las entidades](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [CreateUser](#) la Referencia de AWS CLI comandos.

## create-virtual-mfa-device

En el siguiente ejemplo de código, se muestra cómo usar `create-virtual-mfa-device`.

### AWS CLI

Para crear un MFA dispositivo virtual

En este ejemplo se crea un nuevo MFA dispositivo virtual llamado `BobsMFADevice`. Crea un archivo que contiene la información de arranque denominada `QRCode.png` y la coloca en el directorio `C:/`. El método de arranque utilizado en este ejemplo es `QRCodePNG`.

```

aws iam create-virtual-mfa-device \
  --virtual-mfa-device-name BobsMFADevice \
  --outfile C:/QRCode.png \
  --bootstrap-method QRCodePNG

```

Salida:

```

{
  "VirtualMFADevice": {
    "SerialNumber": "arn:aws:iam::210987654321:mfa/BobsMFADevice"
  }
}

```

Para obtener más información, consulte [Uso de la autenticación multifactorial \(MFA\) AWS en la Guía del AWS IAM usuario](#).



- Para API obtener más información, consulte [CreateVirtualMfaDevice](#) la Referencia de AWS CLI comandos.

## deactivate-mfa-device

En el siguiente ejemplo de código, se muestra cómo usar deactivate-mfa-device.

### AWS CLI

Para desactivar un dispositivo MFA

Este comando desactiva el MFA dispositivo virtual con el ARN

arn:aws:iam::210987654321:mfa/BobsMFADevice que está asociado al usuario. Bob

```
aws iam deactivate-mfa-device \  
  --user-name Bob \  
  --serial-number arn:aws:iam::210987654321:mfa/BobsMFADevice
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de la autenticación multifactorial \(MFA\) AWS en la Guía del AWS IAM usuario](#).

- Para API obtener más información, consulte [DeactivateMfaDevice](#) la Referencia de AWS CLI comandos.

## decode-authorization-message

En el siguiente ejemplo de código, se muestra cómo usar decode-authorization-message.

### AWS CLI

Para decodificar un mensaje de error de autorización

En el siguiente decode-authorization-message ejemplo, se decodifica el mensaje devuelto por la EC2 consola al intentar lanzar una instancia sin los permisos necesarios.

```
aws sts decode-authorization-message \  
  --encoded-message LxzA8VEjEvu-s0TTt3PgYCXik9Yak0qsrFJGRZR98xNcyWAxwRq14xIvd-  
npzbgTevuufCTbjeBAaDARg9cbTK1rJbg3awM33o-Vy3ebPErE2-  
mWR9hVYdvX-0zKgV0WF9pWjZaJSMqxB-aLXo-I_8TTvBq88x8IFPbMArNdpu0IjxDjzf22PF3S0E3XvIQ-
```

```
_PE00aUqHCCcsSrFtvxm6yQD1nbm6VTIVrfa0Bzy8LsoMo7SjIaJ2r5vph6SY5vCCwg6o2JKe3hIHTa8zRrDbZSFMkcX  
Xx9AYAAIr6bhcis7C__bZh4dLAAWooHFGKgfoJcWGwgdzgbu9hWlyVvKTpeot5hsb8qANYjJRCPTKpi6PZfdijIkwb6g
```

El resultado tiene el formato de una cadena de JSON texto de una sola línea que se puede analizar con cualquier JSON procesador de texto.

```
{
  "DecodedMessage": "{\"allowed\":false,\"explicitDeny\":false,\"matchedStatements\":"
  "\":{\n\"items\":[],\n\"failures\":{\n\"items\":[],\n\"context\":{\n\"principal\":"
  "\":{\n\"id\":\n\"AIDAV3ZUEFP6J7GY706L0\", \n\"name\":\n\"chain-user\", \n\"arn\":"
  "\":\n\"arn:aws:iam:403299380220:user/chain-user\", \n\"action\":\n\"ec2:RunInstances\", \n\"resource\":"
  "\":\n\"arn:aws:ec2:us-east-2:403299380220:instance/*\", \n\"conditions\":"
  "\":{\n\"items\":[{\n\"key\":\n\"ec2:InstanceMarketType\", \n\"values\":{\n\"items\":[{\n\"value\":"
  "\":\n\"on-demand\"}]}],{\n\"key\":\n\"aws:Resource\", \n\"values\":{\n\"items\":[{\n\"value\":"
  "\":\n\"instance/*\"}]}],{\n\"key\":\n\"aws:Account\", \n\"values\":{\n\"items\":[{\n\"value\":"
  "\":\n\"403299380220\"}]}],{\n\"key\":\n\"ec2:AvailabilityZone\", \n\"values\":{\n\"items\":"
  "\":{\n\"value\":\n\"us-east-2b\"}]}],{\n\"key\":\n\"ec2:ebsoptimized\", \n\"values\":{\n\"items\":"
  "\":{\n\"value\":\n\"false\"}]}],{\n\"key\":\n\"ec2:IsLaunchTemplateResource\", \n\"values\":"
  "\":{\n\"items\":[{\n\"value\":\n\"false\"}]}],{\n\"key\":\n\"ec2:InstanceType\", \n\"values\":"
  "\":{\n\"items\":[{\n\"value\":\n\"t2.micro\"}]}],{\n\"key\":\n\"ec2:RootDeviceType\", \n\"values\":"
  "\":{\n\"items\":[{\n\"value\":\n\"ebs\"}]}],{\n\"key\":\n\"aws:Region\", \n\"values\":"
  "\":{\n\"items\":[{\n\"value\":\n\"us-east-2\"}]}],{\n\"key\":\n\"aws:Service\", \n\"values\":"
  "\":{\n\"items\":[{\n\"value\":\n\"ec2\"}]}],{\n\"key\":\n\"ec2:InstanceID\", \n\"values\":"
  "\":{\n\"items\":[{\n\"value\":\n\"*\"}]}],{\n\"key\":\n\"aws:Type\", \n\"values\":{\n\"items\":"
  "\":{\n\"value\":\n\"instance\"}]}],{\n\"key\":\n\"ec2:Tenancy\", \n\"values\":{\n\"items\":"
  "\":{\n\"value\":\n\"default\"}]}],{\n\"key\":\n\"ec2:Region\", \n\"values\":{\n\"items\":[{\n\"value\":"
  "\":\n\"us-east-2\"}]}],{\n\"key\":\n\"aws:ARN\", \n\"values\":{\n\"items\":[{\n\"value\":"
  "\":\n\"arn:aws:ec2:us-east-2:403299380220:instance/*\"}]}]}]}]"
}
```

Para obtener más información, consulta [¿Cómo puedo decodificar un mensaje de error de autorización tras recibir un error «UnauthorizedOperation» durante el lanzamiento de una instancia? EC2](#) en AWS Re:post.

- Para API obtener más información, consulte la Referencia [DecodeAuthorizationMessage](#) de AWS CLI comandos.

## delete-access-key

En el siguiente ejemplo de código, se muestra cómo usar delete-access-key.

## AWS CLI

Para eliminar una clave de acceso de un IAM usuario

El siguiente `delete-access-key` comando elimina la clave de acceso especificada (identificador de clave de acceso y clave de acceso secreta) del IAM usuario nombrado Bob.

```
aws iam delete-access-key \  
  --access-key-id AKIDPMS9R04H3FEXAMPLE \  
  --user-name Bob
```

Este comando no genera ninguna salida.

Para enumerar las claves de acceso definidas para un IAM usuario, utilice el `list-access-keys` comando.

Para obtener más información, consulte [Administrar las claves de acceso para IAM los usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteAccessKey](#) la Referencia de AWS CLI comandos.

## **delete-account-alias**

En el siguiente ejemplo de código, se muestra cómo usar `delete-account-alias`.

## AWS CLI

Cómo eliminar un alias de la cuenta

El siguiente comando `delete-account-alias` elimina el alias `mycompany` de la cuenta actual.

```
aws iam delete-account-alias \  
  --account-alias mycompany
```

Este comando no genera ninguna salida.

Para obtener más información, consulta el [identificador de AWS cuenta y su alias](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteAccountAlias](#) la Referencia de AWS CLI comandos.

## delete-account-password-policy

En el siguiente ejemplo de código, se muestra cómo usar `delete-account-password-policy`.

### AWS CLI

Cómo eliminar la política de contraseñas actual de la cuenta

El siguiente comando `delete-account-password-policy` elimina la política de contraseñas para la cuenta actual.

```
aws iam delete-account-password-policy
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de una política de contraseñas de cuentas para IAM los usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteAccountPasswordPolicy](#) la Referencia de AWS CLI comandos.

## delete-group-policy

En el siguiente ejemplo de código, se muestra cómo usar `delete-group-policy`.

### AWS CLI

Para eliminar una política de un IAM grupo

El siguiente comando `delete-group-policy` elimina la política denominada `ExamplePolicy` del grupo denominado `Admins`.

```
aws iam delete-group-policy \  
  --group-name Admins \  
  --policy-name ExamplePolicy
```

Este comando no genera ninguna salida.

Para ver las políticas asociadas a un grupo, utilice el comando `list-group-policies`.

Para obtener más información, consulte [Administración de IAM políticas](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteGroupPolicy](#) la Referencia de AWS CLI comandos.

## delete-group

En el siguiente ejemplo de código, se muestra cómo usar delete-group.

### AWS CLI

Para eliminar un IAM grupo

El siguiente delete-group comando elimina un IAM grupo denominado MyTestGroup.

```
aws iam delete-group \  
  --group-name MyTestGroup
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un grupo IAM de usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteGroup](#) la Referencia de AWS CLI comandos.

## delete-instance-profile

En el siguiente ejemplo de código, se muestra cómo usar delete-instance-profile.

### AWS CLI

Cómo eliminar un perfil de instancia

El siguiente comando delete-instance-profile elimina el perfil de instancia denominado ExampleInstanceProfile.

```
aws iam delete-instance-profile \  
  --instance-profile-name ExampleInstanceProfile
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de perfiles de instancia](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteInstanceProfile](#) la Referencia de AWS CLI comandos.

## delete-login-profile

En el siguiente ejemplo de código, se muestra cómo usar delete-login-profile.

### AWS CLI

Para eliminar la contraseña de un IAM usuario

El siguiente delete-login-profile comando elimina la contraseña del IAM usuario nombrado Bob.

```
aws iam delete-login-profile \  
  --user-name Bob
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar las contraseñas de los IAM usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteLoginProfile](#) la Referencia de AWS CLI comandos.

## delete-open-id-connect-provider

En el siguiente ejemplo de código, se muestra cómo usar delete-open-id-connect-provider.

### AWS CLI

Para eliminar un proveedor de IAM identidad de OpenID Connect

En este ejemplo, se elimina el IAM OIDC proveedor que se conecta al proveedor.  
example.oidcprovider.com

```
aws iam delete-open-id-connect-provider \  
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/  
  example.oidcprovider.com
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de proveedores de identidad de OpenID Connect \(OIDC\)](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteOpenIdConnectProvider](#) la Referencia de AWS CLI comandos.

## **delete-policy-version**

En el siguiente ejemplo de código, se muestra cómo usar `delete-policy-version`.

### AWS CLI

Cómo eliminar una versión de una política administrada

En este ejemplo, se elimina la versión identificada como v2 de la política cuya versión ARN es `arn:aws:iam::123456789012:policy/MySamplePolicy`.

```
aws iam delete-policy-version \  
  --policy-arn arn:aws:iam::123456789012:policy/MyPolicy \  
  --version-id v2
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Políticas y permisos IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeletePolicyVersion](#) la Referencia de AWS CLI comandos.

## **delete-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-policy`.

### AWS CLI

Para eliminar una IAM política

En este ejemplo se elimina la política cuyo nombre ARN es `arn:aws:iam::123456789012:policy/MySamplePolicy`.

```
aws iam delete-policy \  
  --policy-name MySamplePolicy
```

```
--policy-arn arn:aws:iam::123456789012:policy/MySamplePolicy
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Políticas y permisos IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeletePolicy](#) la Referencia de AWS CLI comandos.

## **delete-role-permissions-boundary**

En el siguiente ejemplo de código, se muestra cómo usar `delete-role-permissions-boundary`.

AWS CLI

Para eliminar un límite de permisos de un IAM rol

En el siguiente `delete-role-permissions-boundary` ejemplo, se elimina el límite de permisos del IAM rol especificado. Para aplicar un límite de permisos a un rol, usa el comando `put-role-permissions-boundary`.

```
aws iam delete-role-permissions-boundary \  
  --role-name lambda-application-role
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Políticas y permisos IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteRolePermissionsBoundary](#) la Referencia de AWS CLI comandos.

## **delete-role-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-role-policy`.

AWS CLI

Para eliminar una política de un IAM rol



El siguiente comando `delete-role-policy` elimina la política denominada `ExamplePolicy` del rol denominado `Test-Role`.

```
aws iam delete-role-policy \  
  --role-name Test-Role \  
  --policy-name ExamplePolicy
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Modificación de un rol](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteRolePolicy](#) la Referencia de AWS CLI comandos.

## **delete-role**

En el siguiente ejemplo de código, se muestra cómo usar `delete-role`.

### AWS CLI

Para eliminar un IAM rol

El siguiente comando `delete-role` elimina el rol denominado `Test-Role`.

```
aws iam delete-role \  
  --role-name Test-Role
```

Este comando no genera ninguna salida.

Antes de poder eliminar un rol, debe eliminarlo de cualquier perfil de instancia (`remove-role-from-instance-profile`), separar cualquier política administrada (`detach-role-policy`) y eliminar cualquier política insertada que esté asociada al rol (`delete-role-policy`).

Para obtener más información, consulte [Creación de IAM roles](#) y [Uso de perfiles de instancia](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteRole](#) la Referencia de AWS CLI comandos.

## **delete-saml-provider**

En el siguiente ejemplo de código, se muestra cómo usar `delete-saml-provider`.

## AWS CLI

Para eliminar un SAML proveedor

En este ejemplo se elimina el proveedor IAM SAML 2.0 cuyo ARN es `arn:aws:iam::123456789012:saml-provider/SAMLADFSProvider`.

```
aws iam delete-saml-provider \  
--saml-provider-arn arn:aws:iam::123456789012:saml-provider/SAMLADFSProvider
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de proveedores de IAM SAML identidad](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteSAMLProvider](#) en la Referencia de AWS CLI comandos.

## **delete-server-certificate**

En el siguiente ejemplo de código, se muestra cómo usar `delete-server-certificate`.

## AWS CLI

Para eliminar un certificado de servidor de su AWS cuenta

El siguiente `delete-server-certificate` comando elimina el certificado de servidor especificado de su AWS cuenta.

```
aws iam delete-server-certificate \  
--server-certificate-name myUpdatedServerCertificate
```

Este comando no genera ninguna salida.

Para ver una lista de los certificados de servidor disponibles en su AWS cuenta, utilice el `list-server-certificates` comando.

Para obtener más información, consulte [Administrar los certificados de servidor IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteServerCertificate](#) la Referencia de AWS CLI comandos.

## delete-service-linked-role

En el siguiente ejemplo de código, se muestra cómo usar `delete-service-linked-role`.

### AWS CLI

Cómo eliminar un rol vinculado a un servicio

En el siguiente ejemplo de `delete-service-linked-role`, se elimina el rol vinculado a un servicio especificado que ya no necesita. La eliminación se produce de forma asíncrona. Puede comprobar el estado de la eliminación y confirmar cuando se ha realizado con el comando `get-service-linked-role-deletion-status`.

```
aws iam delete-service-linked-role \  
  --role-name AWSServiceRoleForLexBots
```

Salida:

```
{  
  "DeletionTaskId": "task/aws-service-role/lex.amazonaws.com/  
  AWSServiceRoleForLexBots/1a2b3c4d-1234-abcd-7890-abcdeEXAMPLE"  
}
```

Para obtener más información, consulte [Uso de funciones vinculadas a servicios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte la Referencia [DeleteServiceLinkedRole](#) de AWS CLI comandos.

## delete-service-specific-credential

En el siguiente ejemplo de código, se muestra cómo usar `delete-service-specific-credential`.

### AWS CLI

Ejemplo 1: Eliminar una credencial específica del servicio para el usuario solicitante

En el siguiente `delete-service-specific-credential` ejemplo, se elimina la credencial específica del servicio especificada para el usuario que realiza la solicitud. `service-specific-`

`credential-id` Se proporciona al crear la credencial y se puede recuperar mediante el comando. `list-service-specific-credentials`

```
aws iam delete-service-specific-credential \  
  --service-specific-credential-id ACCAEXAMPLE123EXAMPLE
```

Este comando no genera ninguna salida.

Ejemplo 2: Eliminar una credencial específica de un servicio para un usuario específico

En el siguiente `delete-service-specific-credential` ejemplo, se elimina la credencial específica del servicio especificada para el usuario especificado. `service-specific-credential-id` Se proporciona al crear la credencial y se puede recuperar mediante el comando. `list-service-specific-credentials`

```
aws iam delete-service-specific-credential \  
  --user-name sofia \  
  --service-specific-credential-id ACCAEXAMPLE123EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Crear credenciales de Git para HTTPS las conexiones CodeCommit](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [DeleteServiceSpecificCredential](#) la Referencia de AWS CLI comandos.

## **delete-signing-certificate**

En el siguiente ejemplo de código, se muestra cómo usar `delete-signing-certificate`.

### AWS CLI

Para eliminar un certificado de firma de un IAM usuario

El siguiente `delete-signing-certificate` comando elimina el certificado de firma especificado para el IAM usuario nombrado Bob.

```
aws iam delete-signing-certificate \  
  --user-name Bob
```

```
--user-name Bob \  
--certificate-id TA7SMP42TDN5Z260BPJE7EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener el ID de un certificado de firma, use el comando `list-signing-certificates`.

Para obtener más información, consulta [Administrar certificados de firma](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [DeleteSigningCertificate](#) la Referencia de AWS CLI comandos.

## **delete-ssh-public-key**

En el siguiente ejemplo de código, se muestra cómo usar `delete-ssh-public-key`.

### AWS CLI

Para eliminar una clave SSH pública adjunta a un IAM usuario

El siguiente `delete-ssh-public-key` comando elimina la clave SSH pública especificada adjunta al IAM usuario `sofia`.

```
aws iam delete-ssh-public-key \  
--user-name sofia \  
--ssh-public-key-id APKA123456789EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso y SSH uso de SSH las teclas CodeCommit](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteSshPublicKey](#) la Referencia de AWS CLI comandos.

## **delete-user-permissions-boundary**

En el siguiente ejemplo de código, se muestra cómo usar `delete-user-permissions-boundary`.

## AWS CLI

Para eliminar un límite de permisos de un IAM usuario

En el siguiente `delete-user-permissions-boundary` ejemplo, se elimina el límite de permisos adjunto al IAM usuario nombrado `intern`. Para aplicar un límite de permisos a un usuario, use el comando `put-user-permissions-boundary`.

```
aws iam delete-user-permissions-boundary \  
  --user-name intern
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Políticas y permisos IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteUserPermissionsBoundary](#) la Referencia de AWS CLI comandos.

## `delete-user-policy`

En el siguiente ejemplo de código, se muestra cómo usar `delete-user-policy`.

## AWS CLI

Para eliminar una política de un IAM usuario

El siguiente `delete-user-policy` comando elimina la política especificada del IAM usuario nombrado `Bob`.

```
aws iam delete-user-policy \  
  --user-name Bob \  
  --policy-name ExamplePolicy
```

Este comando no genera ninguna salida.

Para obtener una lista de políticas para un IAM usuario, utilice el `list-user-policies` comando.

Para obtener más información, consulte [Crear un IAM usuario en su AWS cuenta](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteUserPolicy](#) la Referencia de AWS CLI comandos.

## delete-user

En el siguiente ejemplo de código, se muestra cómo usar `delete-user`.

### AWS CLI

Para eliminar un IAM usuario

El siguiente `delete-user` comando elimina el nombre Bob de IAM usuario de la cuenta actual.

```
aws iam delete-user \  
  --user-name Bob
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un IAM usuario](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteUser](#) la Referencia de AWS CLI comandos.

## delete-virtual-mfa-device

En el siguiente ejemplo de código, se muestra cómo usar `delete-virtual-mfa-device`.

### AWS CLI

Para eliminar un MFA dispositivo virtual

El siguiente `delete-virtual-mfa-device` comando elimina el MFA dispositivo especificado de la cuenta corriente.

```
aws iam delete-virtual-mfa-device \  
  --serial-number arn:aws:iam::123456789012:mfa/MFATest
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Desactivación de MFA dispositivos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteVirtualMfaDevice](#) la Referencia de AWS CLI comandos.

## detach-group-policy

En el siguiente ejemplo de código, se muestra cómo usar `detach-group-policy`.

### AWS CLI

Cómo desasociar una política de un grupo

En este ejemplo, se elimina la política gestionada con la ARN `arn:aws:iam::123456789012:policy/TesterAccessPolicy` del grupo denominado `Testers`.

```
aws iam detach-group-policy \  
  --group-name Testers \  
  --policy-arn arn:aws:iam::123456789012:policy/TesterAccessPolicy
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar grupos IAM de usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DetachGroupPolicy](#) la Referencia de AWS CLI comandos.

## detach-role-policy

En el siguiente ejemplo de código, se muestra cómo usar `detach-role-policy`.

### AWS CLI

Cómo desasociar una política de un rol

En este ejemplo, se elimina la política gestionada con la ARN `arn:aws:iam::123456789012:policy/FederatedTesterAccessPolicy` del rol denominado `FedTesterRole`.

```
aws iam detach-role-policy \  
  --role-name FedTesterRole \  
  --policy-arn arn:aws:iam::123456789012:policy/FederatedTesterAccessPolicy
```



```
--policy-arn arn:aws:iam::123456789012:policy/FederatedTesterAccessPolicy
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Modificación de un rol](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DetachRolePolicy](#) la Referencia de AWS CLI comandos.

## **detach-user-policy**

En el siguiente ejemplo de código, se muestra cómo usar `detach-user-policy`.

### AWS CLI

Cómo desasociar una política de un usuario

En este ejemplo, se elimina la política gestionada junto con el ARN `arn:aws:iam::123456789012:policy/TesterPolicy` del usuario `Bob`.

```
aws iam detach-user-policy \  
  --user-name Bob \  
  --policy-arn arn:aws:iam::123456789012:policy/TesterPolicy
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cambiar los permisos de un IAM usuario](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DetachUserPolicy](#) la Referencia de AWS CLI comandos.

## **enable-mfa-device**

En el siguiente ejemplo de código, se muestra cómo usar `enable-mfa-device`.

### AWS CLI

Para habilitar un MFA dispositivo

Después de usar el `create-virtual-mfa-device` comando para crear un nuevo MFA dispositivo virtual, puede asignar el MFA dispositivo a un usuario. En el siguiente

`enable-mfa-device` ejemplo, se asigna el MFA dispositivo con el número de serie `arn:aws:iam::210987654321:mfa/BobsMFADevice` al usuario `Bob`. El comando también sincroniza el dispositivo AWS al incluir los dos primeros códigos secuenciales del dispositivo `virtualMFA`.

```
aws iam enable-mfa-device \  
  --user-name Bob \  
  --serial-number arn:aws:iam::210987654321:mfa/BobsMFADevice \  
  --authentication-code1 123456 \  
  --authentication-code2 789012
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Habilitar un dispositivo virtual de autenticación multifactor \(MFA\)](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [EnableMfaDevice](#) la Referencia de AWS CLI comandos.

## **generate-credential-report**

En el siguiente ejemplo de código, se muestra cómo usar `generate-credential-report`.

AWS CLI

Cómo generar un informe de credenciales

En el siguiente ejemplo, se intenta generar un informe de credenciales para la AWS cuenta.

```
aws iam generate-credential-report
```

Salida:

```
{  
  "State": "STARTED",  
  "Description": "No report exists. Starting a new report generation task"  
}
```

Para obtener más información, consulte [Obtener informes de credenciales para su AWS cuenta](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GenerateCredentialReport](#) la Referencia de AWS CLI comandos.

## generate-organizations-access-report

En el siguiente ejemplo de código, se muestra cómo usar `generate-organizations-access-report`.

### AWS CLI

Ejemplo 1: Para generar un informe de acceso para una raíz de una organización

El siguiente `generate-organizations-access-report` ejemplo inicia un trabajo en segundo plano para crear un informe de acceso para la raíz especificada en una organización. Puede mostrar el informe una vez creado ejecutando el `get-organizations-access-report` comando.

```
aws iam generate-organizations-access-report \
  --entity-path o-4fxmplt198/r-c3xb
```

Salida:

```
{
  "JobId": "a8b6c06f-aaa4-8xmp-28bc-81da71836359"
}
```

Ejemplo 2: Para generar un informe de acceso para una cuenta de una organización

En el siguiente `generate-organizations-access-report` ejemplo, se inicia un trabajo en segundo plano para crear un informe de acceso para el identificador 123456789012 de cuenta de la organización `o-4fxmplt198`. Puede mostrar el informe una vez creado ejecutando el `get-organizations-access-report` comando.

```
aws iam generate-organizations-access-report \
  --entity-path o-4fxmplt198/r-c3xb/123456789012
```

Salida:

```
{
  "JobId": "14b6c071-75f6-2xmp-fb77-faf6fb4201d2"
}
```

```
}
```

Ejemplo 3: Para generar un informe de acceso para una cuenta de una unidad organizativa de una organización

En el siguiente `generate-organizations-access-report` ejemplo, se inicia un trabajo en segundo plano para crear un informe de acceso para el identificador de la cuenta 234567890123 en la unidad `ou-c3xb-1mu7j2yg` organizativa de la organización `o-4fxmpl1t198`. Para mostrar el informe una vez creado, ejecuta el `get-organizations-access-report` comando.

```
aws iam generate-organizations-access-report \
  --entity-path o-4fxmpl1t198/r-c3xb/ou-c3xb-1mu7j2yg/234567890123
```

Salida:

```
{
  "JobId": "2eb6c2e6-0xmp-ec04-1425-c937916a64af"
}
```

Para obtener detalles sobre las raíces y las unidades organizativas de tu organización, usa los `organizations list-organizational-units-for-parent` comandos `organizations list-roots` y.

Para obtener más información, consulte [Refinar los permisos al AWS utilizar la información a la que se accedió por última vez](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GenerateOrganizationsAccessReport](#) la Referencia de AWS CLI comandos.

## **generate-service-last-accessed-details**

En el siguiente ejemplo de código, se muestra cómo usar `generate-service-last-accessed-details`.

AWS CLI

Ejemplo 1: para generar un informe de acceso al servicio para una política personalizada

En el siguiente `generate-service-last-accessed-details` ejemplo, se inicia un trabajo en segundo plano para generar un informe que enumere los servicios a los que acceden IAM

los usuarios y otras entidades con una política personalizada denominada `intern-boundary`. Puede mostrar el informe una vez creado ejecutando el comando `get-service-last-accessed-details`.

```
aws iam generate-service-last-accessed-details \  
  --arn arn:aws:iam::123456789012:policy/intern-boundary
```

Salida:

```
{  
  "JobId": "2eb6c2b8-7b4c-3xmp-3c13-03b72c8cdfdc"  
}
```

Ejemplo 2: Para generar un informe de acceso a los servicios para la `AdministratorAccess` política AWS gestionada

En el siguiente `generate-service-last-accessed-details` ejemplo, se inicia un trabajo en segundo plano para generar un informe que enumere los servicios a los que acceden IAM los usuarios y otras entidades con la `AdministratorAccess` política AWS gestionada. Puede mostrar el informe una vez creado ejecutando el comando `get-service-last-accessed-details`.

```
aws iam generate-service-last-accessed-details \  
  --arn arn:aws:iam::aws:policy/AdministratorAccess
```

Salida:

```
{  
  "JobId": "78b6c2ba-d09e-6xmp-7039-ecde30b26916"  
}
```

Para obtener más información, consulte [Refinar los permisos al AWS utilizar la información a la que se accedió por última vez](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GenerateServiceLastAccessedDetails](#) la Referencia de AWS CLI comandos.

## **get-access-key-last-used**

En el siguiente ejemplo de código, se muestra cómo usar `get-access-key-last-used`.

## AWS CLI

Cómo recuperar información acerca de cuándo se utilizó por última vez la clave de acceso especificada

En el siguiente ejemplo se recupera información acerca de cuándo se utilizó por última vez la clave de acceso ABCDEXAMPLE.

```
aws iam get-access-key-last-used \
  --access-key-id ABCDEXAMPLE
```

Salida:

```
{
  "UserName": "Bob",
  "AccessKeyLastUsed": {
    "Region": "us-east-1",
    "ServiceName": "iam",
    "LastUsedDate": "2015-06-16T22:45:00Z"
  }
}
```

Para obtener más información, consulte [Administrar las claves de acceso para IAM los usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetAccessKeyLastUsed](#) la Referencia de AWS CLI comandos.

## get-account-authorization-details

En el siguiente ejemplo de código, se muestra cómo usar `get-account-authorization-details`.

### AWS CLI

Para enumerar IAM los usuarios, grupos, roles y políticas de una AWS cuenta

El siguiente `get-account-authorization-details` comando devuelve información sobre todos IAM los usuarios, grupos, roles y políticas de la AWS cuenta.

```
aws iam get-account-authorization-details
```

## Salida:

```
{
  "RoleDetailList": [
    {
      "AssumeRolePolicyDocument": {
        "Version": "2012-10-17",
        "Statement": [
          {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
              "Service": "ec2.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
          }
        ]
      },
      "RoleId": "AROA1234567890EXAMPLE",
      "CreateDate": "2014-07-30T17:09:20Z",
      "InstanceProfileList": [
        {
          "InstanceProfileId": "AIPA1234567890EXAMPLE",
          "Roles": [
            {
              "AssumeRolePolicyDocument": {
                "Version": "2012-10-17",
                "Statement": [
                  {
                    "Sid": "",
                    "Effect": "Allow",
                    "Principal": {
                      "Service": "ec2.amazonaws.com"
                    },
                    "Action": "sts:AssumeRole"
                  }
                ]
              },
              "RoleId": "AROA1234567890EXAMPLE",
              "CreateDate": "2014-07-30T17:09:20Z",
              "RoleName": "EC2role",
              "Path": "/",
              "Arn": "arn:aws:iam::123456789012:role/EC2role"
            }
          ]
        }
      ]
    }
  ]
}
```

```
    ],
    "CreateDate": "2014-07-30T17:09:20Z",
    "InstanceProfileName": "EC2role",
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:instance-profile/EC2role"
  }
],
"RoleName": "EC2role",
"Path": "/",
"AttachedManagedPolicies": [
  {
    "PolicyName": "AmazonS3FullAccess",
    "PolicyArn": "arn:aws:iam::aws:policy/AmazonS3FullAccess"
  },
  {
    "PolicyName": "AmazonDynamoDBFullAccess",
    "PolicyArn": "arn:aws:iam::aws:policy/AmazonDynamoDBFullAccess"
  }
],
"RoleLastUsed": {
  "Region": "us-west-2",
  "LastUsedDate": "2019-11-13T17:30:00Z"
},
"RolePolicyList": [],
"Arn": "arn:aws:iam::123456789012:role/EC2role"
}
],
"GroupDetailList": [
  {
    "GroupId": "AIDA1234567890EXAMPLE",
    "AttachedManagedPolicies": {
      "PolicyName": "AdministratorAccess",
      "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
    },
    "GroupName": "Admins",
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:group/Admins",
    "CreateDate": "2013-10-14T18:32:24Z",
    "GroupPolicyList": []
  },
  {
    "GroupId": "AIDA1234567890EXAMPLE",
    "AttachedManagedPolicies": {
      "PolicyName": "PowerUserAccess",
```



```

        "PolicyArn": "arn:aws:iam::aws:policy/PowerUserAccess"
    },
    "GroupName": "Dev",
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:group/Dev",
    "CreateDate": "2013-10-14T18:33:55Z",
    "GroupPolicyList": []
},
{
    "GroupId": "AIDA1234567890EXAMPLE",
    "AttachedManagedPolicies": [],
    "GroupName": "Finance",
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:group/Finance",
    "CreateDate": "2013-10-14T18:57:48Z",
    "GroupPolicyList": [
        {
            "PolicyName": "policygen-201310141157",
            "PolicyDocument": {
                "Version": "2012-10-17",
                "Statement": [
                    {
                        "Action": "aws-portal:*",
                        "Sid": "Stmt1381777017000",
                        "Resource": "*",
                        "Effect": "Allow"
                    }
                ]
            }
        }
    ]
}
],
"UserDetailList": [
    {
        "UserName": "Alice",
        "GroupList": [
            "Admins"
        ],
        "CreateDate": "2013-10-14T18:32:24Z",
        "UserId": "AIDA1234567890EXAMPLE",
        "UserPolicyList": [],
        "Path": "/",
        "AttachedManagedPolicies": [],

```

```
    "Arn": "arn:aws:iam::123456789012:user/Alice"
  },
  {
    "UserName": "Bob",
    "GroupList": [
      "Admins"
    ],
    "CreateDate": "2013-10-14T18:32:25Z",
    "UserId": "AIDA1234567890EXAMPLE",
    "UserPolicyList": [
      {
        "PolicyName": "DenyBillingAndIAMPolicy",
        "PolicyDocument": {
          "Version": "2012-10-17",
          "Statement": {
            "Effect": "Deny",
            "Action": [
              "aws-portal:*",
              "iam:*"
            ],
            "Resource": "*"
          }
        }
      }
    ],
    "Path": "/",
    "AttachedManagedPolicies": [],
    "Arn": "arn:aws:iam::123456789012:user/Bob"
  },
  {
    "UserName": "Charlie",
    "GroupList": [
      "Dev"
    ],
    "CreateDate": "2013-10-14T18:33:56Z",
    "UserId": "AIDA1234567890EXAMPLE",
    "UserPolicyList": [],
    "Path": "/",
    "AttachedManagedPolicies": [],
    "Arn": "arn:aws:iam::123456789012:user/Charlie"
  }
],
"Policies": [
  {
```

```
"PolicyName": "create-update-delete-set-managed-policies",
"CreateDate": "2015-02-06T19:58:34Z",
"AttachmentCount": 1,
"IsAttachable": true,
"PolicyId": "ANPA1234567890EXAMPLE",
"DefaultVersionId": "v1",
"PolicyVersionList": [
  {
    "CreateDate": "2015-02-06T19:58:34Z",
    "VersionId": "v1",
    "Document": {
      "Version": "2012-10-17",
      "Statement": {
        "Effect": "Allow",
        "Action": [
          "iam:CreatePolicy",
          "iam:CreatePolicyVersion",
          "iam>DeletePolicy",
          "iam>DeletePolicyVersion",
          "iam:GetPolicy",
          "iam:GetPolicyVersion",
          "iam>ListPolicies",
          "iam>ListPolicyVersions",
          "iam:SetDefaultPolicyVersion"
        ],
        "Resource": "*"
      }
    },
    "IsDefaultVersion": true
  }
],
"Path": "/",
"Arn": "arn:aws:iam::123456789012:policy/create-update-delete-set-
managed-policies",
"UpdateDate": "2015-02-06T19:58:34Z"
},
{
  "PolicyName": "S3-read-only-specific-bucket",
  "CreateDate": "2015-01-21T21:39:41Z",
  "AttachmentCount": 1,
  "IsAttachable": true,
  "PolicyId": "ANPA1234567890EXAMPLE",
  "DefaultVersionId": "v1",
  "PolicyVersionList": [
```

```

        {
            "CreateDate": "2015-01-21T21:39:41Z",
            "VersionId": "v1",
            "Document": {
                "Version": "2012-10-17",
                "Statement": [
                    {
                        "Effect": "Allow",
                        "Action": [
                            "s3:Get*",
                            "s3:List*"
                        ],
                        "Resource": [
                            "arn:aws:s3:::example-bucket",
                            "arn:aws:s3:::example-bucket/*"
                        ]
                    }
                ]
            },
            "IsDefaultVersion": true
        }
    ],
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:policy/S3-read-only-specific-bucket",
    "UpdateDate": "2015-01-21T23:39:41Z"
},
{
    "PolicyName": "AmazonEC2FullAccess",
    "CreateDate": "2015-02-06T18:40:15Z",
    "AttachmentCount": 1,
    "IsAttachable": true,
    "PolicyId": "ANPA1234567890EXAMPLE",
    "DefaultVersionId": "v1",
    "PolicyVersionList": [
        {
            "CreateDate": "2014-10-30T20:59:46Z",
            "VersionId": "v1",
            "Document": {
                "Version": "2012-10-17",
                "Statement": [
                    {
                        "Action": "ec2:*",
                        "Effect": "Allow",
                        "Resource": "*"
                    }
                ]
            }
        }
    ]
}

```

```

        },
        {
            "Effect": "Allow",
            "Action": "elasticloadbalancing:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "cloudwatch:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "autoscaling:*",
            "Resource": "*"
        }
    ]
},
    "IsDefaultVersion": true
}
],
    "Path": "/",
    "Arn": "arn:aws:iam::aws:policy/AmazonEC2FullAccess",
    "UpdateDate": "2015-02-06T18:40:15Z"
}
],
    "Marker": "EXAMPLEkakov9BCuUNFDtxWSyetzYwEx2ADc8dnzfvERF5S6YMvXKx41t6gCl/
    eeaCX3Jo94/bKqezEAg8TEVS99EKFLxm3jtbpl25FDWEXAMPLE",
    "IsTruncated": true
}

```

Para obtener más información, consulte [las directrices AWS de auditoría de seguridad](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetAccountAuthorizationDetails](#) la Referencia de AWS CLI comandos.

## get-account-password-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-account-password-policy`.

## AWS CLI

Cómo ver la política de contraseñas de la cuenta actual

El siguiente comando `get-account-password-policy` muestra detalles sobre la política de contraseñas de la cuenta actual.

```
aws iam get-account-password-policy
```

Salida:

```
{
  "PasswordPolicy": {
    "AllowUsersToChangePassword": false,
    "RequireLowercaseCharacters": false,
    "RequireUppercaseCharacters": false,
    "MinimumPasswordLength": 8,
    "RequireNumbers": true,
    "RequireSymbols": true
  }
}
```

Si no se ha definido una política de contraseñas para la cuenta, el comando devuelve un error `NoSuchEntity`.

Para obtener más información, consulte [Configuración de una política de contraseñas de cuentas para IAM los usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetAccountPasswordPolicy](#) la Referencia de AWS CLI comandos.

## `get-account-summary`

En el siguiente ejemplo de código, se muestra cómo usar `get-account-summary`.

## AWS CLI

Para obtener información sobre el uso y IAM las cuotas de las IAM entidades en la cuenta corriente

El siguiente `get-account-summary` comando devuelve información sobre el uso actual de la IAM entidad y las cuotas actuales de la IAM entidad en la cuenta.

```
aws iam get-account-summary
```

Salida:

```
{
  "SummaryMap": {
    "UsersQuota": 5000,
    "GroupsQuota": 100,
    "InstanceProfiles": 6,
    "SigningCertificatesPerUserQuota": 2,
    "AccountAccessKeysPresent": 0,
    "RolesQuota": 250,
    "RolePolicySizeQuota": 10240,
    "AccountSigningCertificatesPresent": 0,
    "Users": 27,
    "ServerCertificatesQuota": 20,
    "ServerCertificates": 0,
    "AssumeRolePolicySizeQuota": 2048,
    "Groups": 7,
    "MFADevicesInUse": 1,
    "Roles": 3,
    "AccountMFAEnabled": 1,
    "MFADevices": 3,
    "GroupsPerUserQuota": 10,
    "GroupPolicySizeQuota": 5120,
    "InstanceProfilesQuota": 100,
    "AccessKeysPerUserQuota": 2,
    "Providers": 0,
    "UserPolicySizeQuota": 2048
  }
}
```

Para obtener más información sobre las limitaciones de las entidades, consulta [IAM las AWS STS cuotas](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetAccountSummary](#) la Referencia de AWS CLI comandos.

## get-context-keys-for-custom-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-context-keys-for-custom-policy`.

## AWS CLI

Ejemplo 1: Para enumerar las claves de contexto a las que hacen referencia una o más JSON políticas personalizadas proporcionadas como parámetro en la línea de comandos

El siguiente comando `get-context-keys-for-custom-policy` analiza cada política proporcionada y enumera las claves de contexto utilizadas por esas políticas. Utilice este comando para identificar qué valores clave de contexto debe proporcionar para utilizar correctamente los comandos del simulador de políticas `simulate-custom-policy` y `ysimulate-custom-policy`. También puede recuperar la lista de claves de contexto utilizadas por todas las políticas asociadas a un IAM usuario o rol mediante el `get-context-keys-for-custom-policy` comando. Los parámetros que comienzan con `file://` indican al comando que lea el contenido del archivo y lo use como valor del parámetro en lugar del nombre del archivo en sí.

```
aws iam get-context-keys-for-custom-policy \
  --policy-input-list '{"Version":"2012-10-17","Statement":
{"Effect":"Allow","Action":"dynamodb:*","Resource":"arn:aws:dynamodb:us-
west-2:123456789012:table/${aws:username}","Condition":{"DateGreaterThan":
{"aws:CurrentTime":"2015-08-16T12:00:00Z"}}}}'
```

Salida:

```
{
  "ContextKeyNames": [
    "aws:username",
    "aws:CurrentTime"
  ]
}
```

Ejemplo 2: Para enumerar las claves de contexto a las que hacen referencia una o más JSON políticas personalizadas proporcionadas como entrada de archivo

El siguiente comando `get-context-keys-for-custom-policy` es igual al anterior, excepto que las políticas se proporcionan en un archivo y no como un parámetro. Como el comando espera una JSON lista de cadenas y no una lista de JSON estructuras, el archivo debe estructurarse de la siguiente manera, aunque puede comprimirlo en una sola.

```
[
  "Policy1",
```



```
"Policy2"
]
```

Así, por ejemplo, un archivo que contenga la política del ejemplo anterior debe tener el siguiente aspecto. Debe evitar cada comilla doble incrustada en la cadena de la política precediéndola de una barra invertida “.

```
[ "{\"Version\": \"2012-10-17\", \"Statement\": {\"Effect\": \"Allow\", \"Action\": \"dynamodb:*\", \"Resource\": \"arn:aws:dynamodb:us-west-2:128716708097:table/${aws:username}\", \"Condition\": {\"DateGreaterThan\": {\"aws:CurrentTime\": \"2015-08-16T12:00:00Z\"}}}}" ]
```

A continuación, este archivo se puede enviar al siguiente comando.

```
aws iam get-context-keys-for-custom-policy \
  --policy-input-list file://policyfile.json
```

Salida:

```
{
  "ContextKeyNames": [
    "aws:username",
    "aws:CurrentTime"
  ]
}
```

Para obtener más información, consulte [Uso del simulador de IAM políticas \(AWS CLly AWS API\)](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetContextKeysForCustomPolicy](#) la Referencia de AWS CLI comandos.

## get-context-keys-for-principal-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-context-keys-for-principal-policy`.

### AWS CLI

Para enumerar las claves de contexto a las que hacen referencia todas las políticas asociadas a una entidad IAM principal

El siguiente comando `get-context-keys-for-principal-policy` recupera todas las políticas asociadas al usuario `saanvi` y a cualquier grupo al que pertenezca. A continuación, analiza cada una de ellas y enumera las claves de contexto utilizadas por esas políticas. Utilice este comando para identificar qué valores de clave de contexto debe proporcionar para utilizar correctamente los comandos `simulate-custom-policy` y `simulate-principal-policy`. También puede recuperar la lista de claves de contexto utilizadas por una JSON política arbitraria mediante el `get-context-keys-for-custom-policy` comando.

```
aws iam get-context-keys-for-principal-policy \  
--policy-source-arn arn:aws:iam::123456789012:user/saanvi
```

Salida:

```
{  
  "ContextKeyNames": [  
    "aws:username",  
    "aws:CurrentTime"  
  ]  
}
```

Para obtener más información, consulte [Uso del simulador de IAM políticas \(AWS CLI y AWS API\)](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetContextKeysForPrincipalPolicy](#) la Referencia de AWS CLI comandos.

## get-credential-report

En el siguiente ejemplo de código, se muestra cómo usar `get-credential-report`.

### AWS CLI

Cómo obtener un informe de credencial

En este ejemplo se abre el informe devuelto y se envía a la canalización como una matriz de líneas de texto.

```
aws iam get-credential-report
```

Salida:

```
{
  "GeneratedTime": "2015-06-17T19:11:50Z",
  "ReportFormat": "text/csv"
}
```

Para obtener más información, consulte [Obtener informes de credenciales para su AWS cuenta](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetCredentialReport](#) la Referencia de AWS CLI comandos.

## get-group-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-group-policy`.

### AWS CLI

Para obtener información sobre una política asociada a un IAM grupo

El siguiente comando `get-group-policy` obtiene información sobre la política especificada asociada al grupo denominado `Test-Group`.

```
aws iam get-group-policy \
  --group-name Test-Group \
  --policy-name S3-ReadOnly-Policy
```

Salida:

```
{
  "GroupName": "Test-Group",
  "PolicyDocument": {
    "Statement": [
      {
        "Action": [
          "s3:Get*",
          "s3:List*"
        ],
        "Resource": "*",
        "Effect": "Allow"
      }
    ]
  }
}
```

```
  },  
  "PolicyName": "S3-ReadOnly-Policy"  
}
```

Para obtener más información, consulte [Administración de IAM políticas](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetGroupPolicy](#) la Referencia de AWS CLI comandos.

## get-group

En el siguiente ejemplo de código, se muestra cómo usar `get-group`.

### AWS CLI

Para obtener un IAM grupo

En este ejemplo se devuelven detalles sobre el IAM grupo `Admins`.

```
aws iam get-group \  
  --group-name Admins
```

Salida:

```
{  
  "Group": {  
    "Path": "/",  
    "CreateDate": "2015-06-16T19:41:48Z",  
    "GroupId": "AIDGPMS9R04H3FEXAMPLE",  
    "Arn": "arn:aws:iam::123456789012:group/Admins",  
    "GroupName": "Admins"  
  },  
  "Users": []  
}
```

Para obtener más información, consulte [IAM identidades \(usuarios, grupos de usuarios y roles\)](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetGroup](#) la Referencia de AWS CLI comandos.

## get-instance-profile

En el siguiente ejemplo de código, se muestra cómo usar `get-instance-profile`.

### AWS CLI

Cómo obtener información sobre un perfil de instancia

El siguiente comando `get-instance-profile` obtiene información sobre el perfil de instancia denominado `ExampleInstanceProfile`.

```
aws iam get-instance-profile \  
  --instance-profile-name ExampleInstanceProfile
```

Salida:

```
{  
  "InstanceProfile": {  
    "InstanceProfileId": "AID2MAB8DPLSRHEXAMPLE",  
    "Roles": [  
      {  
        "AssumeRolePolicyDocument": "<URL-encoded-JSON>",  
        "RoleId": "AIDGPMS9R04H3FEXAMPLE",  
        "CreateDate": "2013-01-09T06:33:26Z",  
        "RoleName": "Test-Role",  
        "Path": "/",  
        "Arn": "arn:aws:iam::336924118301:role/Test-Role"  
      }  
    ],  
    "CreateDate": "2013-06-12T23:52:02Z",  
    "InstanceProfileName": "ExampleInstanceProfile",  
    "Path": "/",  
    "Arn": "arn:aws:iam::336924118301:instance-profile/ExampleInstanceProfile"  
  }  
}
```

Para obtener más información, consulte [Uso de perfiles de instancia](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetInstanceProfile](#) la Referencia de AWS CLI comandos.

## get-login-profile

En el siguiente ejemplo de código, se muestra cómo usar `get-login-profile`.

### AWS CLI

Para obtener la información de la contraseña de un IAM usuario

El siguiente `get-login-profile` comando obtiene información sobre la contraseña del IAM usuario nombrado Bob.

```
aws iam get-login-profile \  
  --user-name Bob
```

Salida:

```
{  
  "LoginProfile": {  
    "UserName": "Bob",  
    "CreateDate": "2012-09-21T23:03:39Z"  
  }  
}
```

El `get-login-profile` comando se puede utilizar para comprobar que un IAM usuario tiene una contraseña. Si no se ha definido una política de contraseñas para la cuenta, el comando devuelve el error `NoSuchEntity`.

No puede visualizar una contraseña con este comando. Si no se dispone de la contraseña, puede restablecerla (`update-login-profile`) para el usuario. Como alternativa, puede eliminar el perfil de inicio de sesión (`delete-login-profile`) del usuario y, a continuación, crear uno nuevo (`create-login-profile`).

Para obtener más información, consulte [Administrar las contraseñas de IAM los usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetLoginProfile](#) la Referencia de AWS CLI comandos.

## get-mfa-device

En el siguiente ejemplo de código, se muestra cómo usar `get-mfa-device`.

## AWS CLI

Para recuperar información sobre una clave FIDO de seguridad

El siguiente ejemplo de `get-mfa-device` comando recupera información sobre la clave de FIDO seguridad especificada.

```
aws iam get-mfa-device \
  --serial-number arn:aws:iam::123456789012:u2f/user/alice/fidokeyname-
  EXAMPLEBN5FHTECLFG7EXAMPLE
```

Salida:

```
{
  "UserName": "alice",
  "SerialNumber": "arn:aws:iam::123456789012:u2f/user/alice/fidokeyname-
  EXAMPLEBN5FHTECLFG7EXAMPLE",
  "EnableDate": "2023-09-19T01:49:18+00:00",
  "Certifications": {
    "FIDO": "L1"
  }
}
```

Para obtener más información, consulte [Uso de la autenticación multifactorial \(MFA\) AWS en la Guía del AWS IAM usuario](#).

- Para API obtener más información, consulte [GetMfaDevice](#) la Referencia de AWS CLI comandos.

## get-open-id-connect-provider

En el siguiente ejemplo de código, se muestra cómo usar `get-open-id-connect-provider`.

### AWS CLI

Cómo devolver información sobre el proveedor de OpenID Connect especificado

Este ejemplo devuelve detalles sobre el proveedor de OpenID Connect cuyo proveedor es ARN.  
`arn:aws:iam::123456789012:oidc-provider/server.example.com`

```
aws iam get-open-id-connect-provider \
```





```
"JobStatus": "COMPLETED",
"JobCreationDate": "2019-09-30T06:53:36.187Z",
"JobCompletionDate": "2019-09-30T06:53:37.547Z",
"NumberOfServicesAccessible": 188,
"NumberOfServicesNotAccessed": 171,
"AccessDetails": [
  {
    "ServiceName": "Alexa for Business",
    "ServiceNamespace": "a4b",
    "TotalAuthenticatedEntities": 0
  },
  ...
]
```

Para obtener más información, consulte [Refinar los permisos al AWS utilizar la información a la que se accedió por última vez](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetOrganizationsAccessReport](#) la Referencia de AWS CLI comandos.

## get-policy-version

En el siguiente ejemplo de código, se muestra cómo usar `get-policy-version`.

### AWS CLI

Cómo recuperar información acerca de la versión especificada de la política administrada especificada

Este ejemplo devuelve el documento de política correspondiente a la versión v2 de la política cuya política ARN es `arn:aws:iam::123456789012:policy/MyManagedPolicy`.

```
aws iam get-policy-version \
  --policy-arn arn:aws:iam::123456789012:policy/MyPolicy \
  --version-id v2
```

Salida:

```
{
  "PolicyVersion": {
    "Document": {
```

```
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": "iam:*",
        "Resource": "*"
      }
    ]
  },
  "VersionId": "v2",
  "IsDefaultVersion": true,
  "CreateDate": "2023-04-11T00:22:54+00:00"
}
}
```

Para obtener más información, consulte [Políticas y permisos IAM en la Guía del AWS IAM usuario](#).

- Para API obtener más información, consulte [GetPolicyVersion](#) la Referencia de AWS CLI comandos.

## get-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-policy`.

### AWS CLI

Cómo recuperar información sobre una política administrada especificada

Este ejemplo devuelve detalles sobre la política gestionada cuya política ARN es `arn:aws:iam::123456789012:policy/MySamplePolicy`.

```
aws iam get-policy \
  --policy-arn arn:aws:iam::123456789012:policy/MySamplePolicy
```

Salida:

```
{
  "Policy": {
    "PolicyName": "MySamplePolicy",
    "CreateDate": "2015-06-17T19:23:32Z",
```

```
    "AttachmentCount": 0,  
    "IsAttachable": true,  
    "PolicyId": "Z27SI6FQMGNO2EXAMPLE1",  
    "DefaultVersionId": "v1",  
    "Path": "/",  
    "Arn": "arn:aws:iam::123456789012:policy/MySamplePolicy",  
    "UpdateDate": "2015-06-17T19:23:32Z"  
  }  
}
```

Para obtener más información, consulte [Políticas y permisos IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetPolicy](#) la Referencia de AWS CLI comandos.

## get-role-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-role-policy`.

### AWS CLI

Para obtener información sobre una política asociada a un IAM rol

El siguiente comando `get-role-policy` obtiene información sobre la política especificada asociada al rol denominado `Test-Role`.

```
aws iam get-role-policy \  
  --role-name Test-Role \  
  --policy-name ExamplePolicy
```

Salida:

```
{  
  "RoleName": "Test-Role",  
  "PolicyDocument": {  
    "Statement": [  
      {  
        "Action": [  
          "s3:ListBucket",  
          "s3:Put*",  
          "s3:Get*",  
          "s3:*MultipartUpload*"  
        ]  
      }  
    ]  
  }  
}
```

```
        ],
        "Resource": "*",
        "Effect": "Allow",
        "Sid": "1"
    }
]
}
"PolicyName": "ExamplePolicy"
}
```

Para obtener más información, consulte [Creación de IAM roles](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetRolePolicy](#) la Referencia de AWS CLI comandos.

## get-role

En el siguiente ejemplo de código, se muestra cómo usar `get-role`.

### AWS CLI

Para obtener información sobre un IAM rol

El siguiente comando `get-role` obtiene información sobre el rol denominado `Test-Role`.

```
aws iam get-role \
  --role-name Test-Role
```

Salida:

```
{
  "Role": {
    "Description": "Test Role",
    "AssumeRolePolicyDocument": "<URL-encoded-JSON>",
    "MaxSessionDuration": 3600,
    "RoleId": "AROA1234567890EXAMPLE",
    "CreateDate": "2019-11-13T16:45:56Z",
    "RoleName": "Test-Role",
    "Path": "/",
    "RoleLastUsed": {
      "Region": "us-east-1",
      "LastUsedDate": "2019-11-13T17:14:00Z"
    }
  }
}
```

```

    },
    "Arn": "arn:aws:iam::123456789012:role/Test-Role"
  }
}

```

El comando muestra la política de confianza asociada al rol. Utilice el comando `list-role-policies` para enumerar las políticas de permisos asociadas al rol.

Para obtener más información, consulte [Creación de IAM roles](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetRole](#) la Referencia de AWS CLI comandos.

## get-saml-provider

En el siguiente ejemplo de código, se muestra cómo usar `get-saml-provider`.

### AWS CLI

Para recuperar el SAML metadocument del proveedor

En este ejemplo, se recuperan los detalles sobre el proveedor de la SAML versión 2.0 cuyo proveedor es. ARM `arn:aws:iam::123456789012:saml-provider/SAMLADFS` La respuesta incluye el documento de metadatos que recibió del proveedor de identidades para crear la entidad AWS SAML proveedora, así como las fechas de creación y caducidad.

```

aws iam get-saml-provider \
  --saml-provider-arn arn:aws:iam::123456789012:saml-provider/SAMLADFS

```

Salida:

```

{
  "SAMLMetadataDocument": "...SAMLMetadataDocument-XML...",
  "CreateDate": "2017-03-06T22:29:46+00:00",
  "ValidUntil": "2117-03-06T22:29:46.433000+00:00",
  "Tags": [
    {
      "Key": "DeptID",
      "Value": "123456"
    },
    {
      "Key": "Department",

```

```

        "Value": "Accounting"
      }
    ]
  }

```

Para obtener más información, consulte [Creación de proveedores de IAM SAML identidad](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetSamlProvider](#) la Referencia de AWS CLI comandos.

## get-server-certificate

En el siguiente ejemplo de código, se muestra cómo usar `get-server-certificate`.

### AWS CLI

Para obtener detalles sobre un certificado de servidor en su AWS cuenta

El siguiente `get-server-certificate` comando recupera todos los detalles sobre el certificado de servidor especificado en su AWS cuenta.

```

aws iam get-server-certificate \
  --server-certificate-name myUpdatedServerCertificate

```

Salida:

```

{
  "ServerCertificate": {
    "ServerCertificateMetadata": {
      "Path": "/",
      "ServerCertificateName": "myUpdatedServerCertificate",
      "ServerCertificateId": "ASCAEXAMPLE123EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:server-certificate/myUpdatedServerCertificate",
      "UploadDate": "2019-04-22T21:13:44+00:00",
      "Expiration": "2019-10-15T22:23:16+00:00"
    },
    "CertificateBody": "-----BEGIN CERTIFICATE-----
      MIICiTCCAfICCD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAGGA1UEBhMC
      VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6

```

```

b24xFDASBgNVBA5TC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZlZlbnR1eWVhZAd
BgkqhkiG9w0BCQEWEG5vb251QGFTYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAKGA1UEBhMCVVMxMjA0NTIxWjCBiDELMAKGA1UEBh
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC01BTSBDb25z
b2x1MRIwEAYDVQQDEw1UZlZlbnR1eWVhZAdBgkqhkiG9w0BCQEWEG5vb251QGFT
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTc2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVvxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStb
NYiytVbZPQUQ5Yaxu2jXnimvrszlaEXAMPLE=-----END CERTIFICATE-----",
"CertificateChain": "-----BEGIN CERTIFICATE-----\nMIICiTCcAfICCQD6md
7oRw0uX0jANBgkqhkiG9w0BAQQUFADCBiDELMAKGA1UEBhMCVVMxMjA0NTIxWjCBiDELMAKGA1UEBh
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC01BTSBDb25zb2x1MRIwEAYDV
sQQDEw1UZlZlbnR1eWVhZAdBgkqhkiG9w0BCQ
jb20wHhcNMTEwNDI1MjA0NTIxWhcNMTEwNDI0MjA0NTIxWjCBiDELMAKGA1UEBh
MCVVMxMjA0NTIxWjCBiDELMAKGA1UEBhMCVVMxMjA0NTIxWjCBiDELMAKGA1UEBh
WF6b24xFDASBgNVBA5TC01BTSBDb2d5zb2x1MRIwEAYDVQQDEw1UZlZlbnR1eWVhZAdBgkqhkiG9w0BCQ
EWEG5vb251QGFTYXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEB
BQADgY0AMIGJAoGBAMaK0dn+a4GmWIgWJ21uUSfwfEvySwTc2XADZ4nB+BLyGVI
k60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9TrDHudUZg3qX4waLG5M43q7Wgc/MbQ
ITx0USQv7c7ugFFDzQGBzZswY6786m86gjpEIbb30hjZnzcvcQAaRHhd1QWIMm2nr
AgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCku4nUhVvxYUntneD9+h8Mg9q6q+auN
KyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0F1kbFFBjvSfpJI1J00zbhNYS5f6Guo
EDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjS;TbNYiytVbZPQUQ5Yaxu2jXnimvw
3rrszlaEWEG5vb251QGFtsYXpvbiEXAMPLE=\n-----END CERTIFICATE-----"
}
}

```

Para ver una lista de los certificados de servidor disponibles en su AWS cuenta, utilice el `list-server-certificates` comando.

Para obtener más información, consulte [Administrar los certificados de servidor IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetServerCertificate](#) la Referencia de AWS CLI comandos.

## get-service-last-accessed-details-with-entities

En el siguiente ejemplo de código, se muestra cómo usar `get-service-last-accessed-details-with-entities`.

## AWS CLI

Cómo recuperar un informe de acceso a un servicio con los detalles de un servicio

En el siguiente `get-service-last-accessed-details-with-entities` ejemplo, se recupera un informe que contiene detalles sobre IAM los usuarios y otras entidades que accedieron al servicio especificado. Para generar un informe, utilice el comando `generate-service-last-accessed-details`. Para obtener una lista de los servicios a los que se accede con espacios de nombres, utilice `get-service-last-accessed-details`.

```
aws iam get-service-last-accessed-details-with-entities \  
  --job-id 78b6c2ba-d09e-6xmp-7039-ecde30b26916 \  
  --service-namespace lambda
```

Salida:

```
{  
  "JobStatus": "COMPLETED",  
  "JobCreationDate": "2019-10-01T03:55:41.756Z",  
  "JobCompletionDate": "2019-10-01T03:55:42.533Z",  
  "EntityDetailsList": [  
    {  
      "EntityInfo": {  
        "Arn": "arn:aws:iam::123456789012:user/admin",  
        "Name": "admin",  
        "Type": "USER",  
        "Id": "AIDAI02XMPLENQEXAMPLE",  
        "Path": "/"  
      },  
      "LastAuthenticated": "2019-09-30T23:02:00Z"  
    },  
    {  
      "EntityInfo": {  
        "Arn": "arn:aws:iam::123456789012:user/developer",  
        "Name": "developer",  
        "Type": "USER",  
        "Id": "AIDAIBEYXMPL2YEXAMPLE",  
        "Path": "/"  
      },  
      "LastAuthenticated": "2019-09-16T19:34:00Z"  
    }  
  ]  
}
```



```
}
```

Para obtener más información, consulte [Refinar los permisos al AWS utilizar la información a la que se accedió por última vez](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetServiceLastAccessedDetailsWithEntities](#) la Referencia de AWS CLI comandos.

## get-service-last-accessed-details

En el siguiente ejemplo de código, se muestra cómo usar `get-service-last-accessed-details`.

### AWS CLI

Cómo recuperar un informe de acceso a un servicio

El siguiente `get-service-last-accessed-details` ejemplo recupera un informe generado anteriormente que enumera los servicios a los que acceden las IAM entidades. Para generar un informe, utilice el comando `generate-service-last-accessed-details`.

```
aws iam get-service-last-accessed-details \
  --job-id 2eb6c2b8-7b4c-3xmp-3c13-03b72c8cdfdc
```

Salida:

```
{
  "JobStatus": "COMPLETED",
  "JobCreationDate": "2019-10-01T03:50:35.929Z",
  "ServicesLastAccessed": [
    ...
    {
      "ServiceName": "AWS Lambda",
      "LastAuthenticated": "2019-09-30T23:02:00Z",
      "ServiceNamespace": "lambda",
      "LastAuthenticatedEntity": "arn:aws:iam::123456789012:user/admin",
      "TotalAuthenticatedEntities": 6
    },
  ]
}
```

Para obtener más información, consulte [Refinar los permisos al AWS utilizar la información a la que se accedió por última vez](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetServiceLastAccessedDetails](#) la Referencia de AWS CLI comandos.

## **get-service-linked-role-deletion-status**

En el siguiente ejemplo de código, se muestra cómo usar `get-service-linked-role-deletion-status`.

### AWS CLI

Cómo comprobar el estado de una solicitud de eliminación de un rol vinculado a un servicio

En el siguiente ejemplo de `get-service-linked-role-deletion-status`, se muestra el estado de una solicitud anterior para eliminar un rol vinculado a un servicio. La operación de eliminación se produce de forma asíncrona. Al realizar la solicitud, se obtiene un valor de `DeletionTaskId` que proporciona como un parámetro para este comando.

```
aws iam get-service-linked-role-deletion-status \
  --deletion-task-id task/aws-service-role/Lex.amazonaws.com/
AWSServiceRoleForLexBots/1a2b3c4d-1234-abcd-7890-abcdeEXAMPLE
```

Salida:

```
{
  "Status": "SUCCEEDED"
}
```

Para obtener más información, consulte [Uso de funciones vinculadas a servicios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte la Referencia [GetServiceLinkedRoleDeletionStatus](#) de AWS CLI comandos.

## **get-ssh-public-key**

En el siguiente ejemplo de código, se muestra cómo usar `get-ssh-public-key`.

## AWS CLI

Ejemplo 1: Para recuperar una clave SSH pública adjunta a un IAM usuario en forma SSH codificada

El siguiente `get-ssh-public-key` comando recupera la clave SSH pública especificada del IAM usuario `sofia`. El resultado está SSH codificado.

```
aws iam get-ssh-public-key \  
  --user-name sofia \  
  --ssh-public-key-id APKA123456789EXAMPLE \  
  --encoding SSH
```

Salida:

```
{  
  "SSHPublicKey": {  
    "UserName": "sofia",  
    "SSHPublicKeyId": "APKA123456789EXAMPLE",  
    "Fingerprint": "12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef",  
    "SSHPublicKeyBody": "ssh-rsa <<long encoded SSH string>>",  
    "Status": "Inactive",  
    "UploadDate": "2019-04-18T17:04:49+00:00"  
  }  
}
```

Ejemplo 2: Para recuperar una clave SSH pública adjunta a un IAM usuario en forma PEM codificada

El siguiente `get-ssh-public-key` comando recupera la clave SSH pública especificada del IAM usuario `sofia`. El resultado está PEM codificado.

```
aws iam get-ssh-public-key \  
  --user-name sofia \  
  --ssh-public-key-id APKA123456789EXAMPLE \  
  --encoding PEM
```

Salida:

```
{  
  "SSHPublicKey": {  
    "UserName": "sofia",
```

```

    "SSHPublicKeyId": "APKA123456789EXAMPLE",
    "Fingerprint": "12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef",
    "SSHPublicKeyBody": ""-----BEGIN PUBLIC KEY-----\n<<long encoded PEM
string>>\n-----END PUBLIC KEY-----\n"",
    "Status": "Inactive",
    "UploadDate": "2019-04-18T17:04:49+00:00"
  }
}

```

Para obtener más información, consulte [Uso y SSH uso de SSH las teclas CodeCommit](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetSshPublicKey](#) la Referencia de AWS CLI comandos.

## get-user-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-user-policy`.

### AWS CLI

Para enumerar los detalles de la política de un IAM usuario

El siguiente `get-user-policy` comando muestra los detalles de la política especificada que se adjunta al IAM usuario nombrado `Bob`.

```

aws iam get-user-policy \
  --user-name Bob \
  --policy-name ExamplePolicy

```

Salida:

```

{
  "UserName": "Bob",
  "PolicyName": "ExamplePolicy",
  "PolicyDocument": {
    "Version": "2012-10-17",
    "Statement": [
      {
        "Action": "*",
        "Resource": "*",
        "Effect": "Allow"
      }
    ]
  }
}

```

```
    }
  ]
}
}
```

Para obtener una lista de políticas para un IAM usuario, utilice el `list-user-policies` comando.

Para obtener más información, consulte [Políticas y permisos IAM en la Guía del AWS IAM usuario](#).

- Para API obtener más información, consulte [GetUserPolicy](#) la Referencia de AWS CLI comandos.

## get-user

En el siguiente ejemplo de código, se muestra cómo usar `get-user`.

### AWS CLI

Para obtener información sobre un IAM usuario

El siguiente `get-user` comando obtiene información sobre el IAM usuario nombrado `Paulo`.

```
aws iam get-user \
  --user-name Paulo
```

Salida:

```
{
  "User": {
    "UserName": "Paulo",
    "Path": "/",
    "CreateDate": "2019-09-21T23:03:13Z",
    "UserId": "AIDA123456789EXAMPLE",
    "Arn": "arn:aws:iam::123456789012:user/Paulo"
  }
}
```

Para obtener más información, consulte [Administración de IAM usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetUser](#) la Referencia de AWS CLI comandos.

## list-access-keys

En el siguiente ejemplo de código, se muestra cómo usar `list-access-keys`.

### AWS CLI

Para enumerar la clave de acceso IDs de un IAM usuario

El siguiente `list-access-keys` comando muestra las claves IDs de acceso del IAM usuario nombrado `Bob`.

```
aws iam list-access-keys \  
  --user-name Bob
```

Salida:

```
{  
  "AccessKeyMetadata": [  
    {  
      "UserName": "Bob",  
      "Status": "Active",  
      "CreateDate": "2013-06-04T18:17:34Z",  
      "AccessKeyId": "AKIAIOSFODNN7EXAMPLE"  
    },  
    {  
      "UserName": "Bob",  
      "Status": "Inactive",  
      "CreateDate": "2013-06-06T20:42:26Z",  
      "AccessKeyId": "AKIAI44QH8DHBEXAMPLE"  
    }  
  ]  
}
```

No puede enumerar las claves de acceso secretas de IAM los usuarios. Si se pierden las claves de acceso secretas, debe crear nuevas claves de acceso mediante el comando `create-access-keys`.

Para obtener más información, consulte [Administrar las claves de acceso de IAM los usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListAccessKeys](#) la Referencia de AWS CLI comandos.

## list-account-aliases

En el siguiente ejemplo de código, se muestra cómo usar `list-account-aliases`.

### AWS CLI

Cómo enumerar los alias de una cuenta

El siguiente comando `list-account-aliases` enumera los alias de la cuenta actual.

```
aws iam list-account-aliases
```

Salida:

```
{
  "AccountAliases": [
    "mycompany"
  ]
}
```

Para obtener más información, consulte el [identificador de su AWS cuenta y su alias](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListAccountAliases](#) la Referencia de AWS CLI comandos.

## list-attached-group-policies

En el siguiente ejemplo de código, se muestra cómo usar `list-attached-group-policies`.

### AWS CLI

Cómo enumerar todas las políticas administradas que se asocian al grupo especificado

Este ejemplo devuelve los nombres y ARNs las políticas administradas que están asociadas al IAM grupo nombrado `Admins` en la AWS cuenta.

```
aws iam list-attached-group-policies \
  --group-name Admins
```

Salida:

```
{
  "AttachedPolicies": [
    {
      "PolicyName": "AdministratorAccess",
      "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
    },
    {
      "PolicyName": "SecurityAudit",
      "PolicyArn": "arn:aws:iam::aws:policy/SecurityAudit"
    }
  ],
  "IsTruncated": false
}
```

Para obtener más información, consulte [Políticas y permisos IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListAttachedGroupPolicies](#) la Referencia de AWS CLI comandos.

## list-attached-role-policies

En el siguiente ejemplo de código, se muestra cómo usar `list-attached-role-policies`.

### AWS CLI

Cómo enumerar todas las políticas administradas que se asocian al rol especificado

Este comando devuelve los nombres y ARNs las políticas administradas asociadas al IAM rol nombrado `SecurityAuditRole` en la AWS cuenta.

```
aws iam list-attached-role-policies \
  --role-name SecurityAuditRole
```

Salida:

```
{
  "AttachedPolicies": [
    {
      "PolicyName": "SecurityAudit",
      "PolicyArn": "arn:aws:iam::aws:policy/SecurityAudit"
    }
  ]
}
```



```
  ],  
  "IsTruncated": false  
}
```

Para obtener más información, consulte [Políticas y permisos IAM en la Guía del AWS IAM usuario](#).

- Para API obtener más información, consulte [ListAttachedRolePolicies](#) la Referencia de AWS CLI comandos.

## **list-attached-user-policies**

En el siguiente ejemplo de código, se muestra cómo usar `list-attached-user-policies`.

### AWS CLI

Cómo enumerar todas las políticas administradas que se asocian al usuario especificado

Este comando devuelve los nombres y ARNs las políticas administradas del IAM usuario nombrado Bob en la AWS cuenta.

```
aws iam list-attached-user-policies \  
  --user-name Bob
```

Salida:

```
{  
  "AttachedPolicies": [  
    {  
      "PolicyName": "AdministratorAccess",  
      "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"  
    },  
    {  
      "PolicyName": "SecurityAudit",  
      "PolicyArn": "arn:aws:iam::aws:policy/SecurityAudit"  
    }  
  ],  
  "IsTruncated": false  
}
```

Para obtener más información, consulte [Políticas y permisos IAM en la Guía del AWS IAM usuario](#).

- Para API obtener más información, consulte [ListAttachedUserPolicies](#) la Referencia de AWS CLI comandos.

## list-entities-for-policy

En el siguiente ejemplo de código, se muestra cómo usar `list-entities-for-policy`.

### AWS CLI

Cómo enumerar todos los usuarios, grupos y roles a los que está adjunta la política administrada especificada

Este ejemplo devuelve una lista de IAM grupos, funciones y usuarios a los que se ha `arn:aws:iam::123456789012:policy/TestPolicy` adjuntado la política.

```
aws iam list-entities-for-policy \  
  --policy-arn arn:aws:iam::123456789012:policy/TestPolicy
```

Salida:

```
{  
  "PolicyGroups": [  
    {  
      "GroupName": "Admins",  
      "GroupId": "AGPACKCEVSQ6C2EXAMPLE"  
    }  
  ],  
  "PolicyUsers": [  
    {  
      "UserName": "Alice",  
      "UserId": "AIDACKCEVSQ6C2EXAMPLE"  
    }  
  ],  
  "PolicyRoles": [  
    {  
      "RoleName": "DevRole",  
      "RoleId": "AROADBQP57FF2AEXAMPLE"  
    }  
  ],  
  "IsTruncated": false  
}
```

Para obtener más información, consulte [Políticas y permisos IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListEntitiesForPolicy](#) la Referencia de AWS CLI comandos.

## **list-group-policies**

En el siguiente ejemplo de código, se muestra cómo usar `list-group-policies`.

### AWS CLI

Cómo enumerar todas las políticas integradas que se asocian al grupo especificado

El siguiente `list-group-policies` comando muestra los nombres de las políticas integradas que están asociadas al IAM grupo mencionado `Admins` en la cuenta corriente.

```
aws iam list-group-policies \  
  --group-name Admins
```

Salida:

```
{  
  "PolicyNames": [  
    "AdminRoot",  
    "ExamplePolicy"  
  ]  
}
```

Para obtener más información, consulte [Administración de IAM políticas](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListGroupPolicies](#) la Referencia de AWS CLI comandos.

## **list-groups-for-user**

En el siguiente ejemplo de código, se muestra cómo usar `list-groups-for-user`.

### AWS CLI

Para enumerar los grupos a los que pertenece un IAM usuario

El siguiente `list-groups-for-user` comando muestra los grupos a los que Bob pertenece el IAM usuario nombrado.

```
aws iam list-groups-for-user \  
  --user-name Bob
```

Salida:

```
{  
  "Groups": [  
    {  
      "Path": "/",  
      "CreateDate": "2013-05-06T01:18:08Z",  
      "GroupId": "AKIAIOSFODNN7EXAMPLE",  
      "Arn": "arn:aws:iam::123456789012:group/Admin",  
      "GroupName": "Admin"  
    },  
    {  
      "Path": "/",  
      "CreateDate": "2013-05-06T01:37:28Z",  
      "GroupId": "AKIAI44QH8DHBEXAMPLE",  
      "Arn": "arn:aws:iam::123456789012:group/s3-Users",  
      "GroupName": "s3-Users"  
    }  
  ]  
}
```

Para obtener más información, consulte [Administración de grupos IAM de usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListGroupForUser](#) la Referencia de AWS CLI comandos.

## **list-groups**

En el siguiente ejemplo de código, se muestra cómo usar `list-groups`.

AWS CLI

Para ver una lista de los IAM grupos de la cuenta corriente

El siguiente `list-groups` comando muestra los IAM grupos de la cuenta corriente.

```
aws iam list-groups
```

Salida:

```
{
  "Groups": [
    {
      "Path": "/",
      "CreateDate": "2013-06-04T20:27:27.972Z",
      "GroupId": "AIDACKCEVSQ6C2EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:group/Admins",
      "GroupName": "Admins"
    },
    {
      "Path": "/",
      "CreateDate": "2013-04-16T20:30:42Z",
      "GroupId": "AIDGPMS9R04H3FEXAMPLE",
      "Arn": "arn:aws:iam::123456789012:group/S3-Admins",
      "GroupName": "S3-Admins"
    }
  ]
}
```

Para obtener más información, consulte [Administración de grupos IAM de usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListGroupsWith](#) la Referencia de AWS CLI comandos.

## list-instance-profile-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-instance-profile-tags`.

AWS CLI

Para ver una lista de las etiquetas adjuntas a un perfil de instancia

El siguiente `list-instance-profile-tags` comando recupera la lista de etiquetas asociadas al perfil de instancia especificado.

```
aws iam list-instance-profile-tags \
  --instance-profile-name deployment-role
```

Salida:

```
{
  "Tags": [
    {
      "Key": "DeptID",
      "Value": "123456"
    },
    {
      "Key": "Department",
      "Value": "Accounting"
    }
  ]
}
```

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListInstanceProfileTags](#) la Referencia de AWS CLI comandos.

## **list-instance-profiles-for-role**

En el siguiente ejemplo de código, se muestra cómo usar `list-instance-profiles-for-role`.

AWS CLI

Para ver una lista de los perfiles de instancia de un IAM rol

El siguiente comando `list-instance-profiles-for-role` muestra los perfiles de instancia que están asociados con el rol `Test-Role`.

```
aws iam list-instance-profiles-for-role \
  --role-name Test-Role
```

Salida:

```
{
  "InstanceProfiles": [
    {
      "InstanceProfileId": "AIDGPMS9R04H3FEXAMPLE",
      "Roles": [
```

```

        {
            "AssumeRolePolicyDocument": "<URL-encoded-JSON>",
            "RoleId": "AIDACKCEVSQ6C2EXAMPLE",
            "CreateDate": "2013-06-07T20:42:15Z",
            "RoleName": "Test-Role",
            "Path": "/",
            "Arn": "arn:aws:iam::123456789012:role/Test-Role"
        }
    ],
    "CreateDate": "2013-06-07T21:05:24Z",
    "InstanceProfileName": "ExampleInstanceProfile",
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:instance-profile/
ExampleInstanceProfile"
}
]
}

```

Para obtener más información, consulte [Uso de perfiles de instancia](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListInstanceProfilesForRole](#) la Referencia de AWS CLI comandos.

## list-instance-profiles

En el siguiente ejemplo de código, se muestra cómo usar `list-instance-profiles`.

### AWS CLI

Cómo enumerar los perfiles de instancia de la cuenta

El siguiente comando `list-instance-profiles` enumera los perfiles de instancia asociados con la cuenta actual.

```
aws iam list-instance-profiles
```

Salida:

```

{
  "InstanceProfiles": [
    {
      "Path": "/",

```

```

    "InstanceProfileName": "example-dev-role",
    "InstanceProfileId": "AIPAIXEU4NUHUPEXAMPLE",
    "Arn": "arn:aws:iam::123456789012:instance-profile/example-dev-role",
    "CreateDate": "2023-09-21T18:17:41+00:00",
    "Roles": [
      {
        "Path": "/",
        "RoleName": "example-dev-role",
        "RoleId": "AROAJ520TH4H7LEXAMPLE",
        "Arn": "arn:aws:iam::123456789012:role/example-dev-role",
        "CreateDate": "2023-09-21T18:17:40+00:00",
        "AssumeRolePolicyDocument": {
          "Version": "2012-10-17",
          "Statement": [
            {
              "Effect": "Allow",
              "Principal": {
                "Service": "ec2.amazonaws.com"
              },
              "Action": "sts:AssumeRole"
            }
          ]
        }
      }
    ]
  },
  {
    "Path": "/",
    "InstanceProfileName": "example-s3-role",
    "InstanceProfileId": "AIPAJVJVNRIQFREXAMPLE",
    "Arn": "arn:aws:iam::123456789012:instance-profile/example-s3-role",
    "CreateDate": "2023-09-21T18:18:50+00:00",
    "Roles": [
      {
        "Path": "/",
        "RoleName": "example-s3-role",
        "RoleId": "AROAINUBC507XLEXAMPLE",
        "Arn": "arn:aws:iam::123456789012:role/example-s3-role",
        "CreateDate": "2023-09-21T18:18:49+00:00",
        "AssumeRolePolicyDocument": {
          "Version": "2012-10-17",
          "Statement": [
            {
              "Effect": "Allow",

```



```

        "Principal": {
            "Service": "ec2.amazonaws.com"
        },
        "Action": "sts:AssumeRole"
    }
]
}

```

Para obtener más información, consulte [Uso de perfiles de instancia](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListInstanceProfiles](#) la Referencia de AWS CLI comandos.

## list-mfa-device-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-mfa-device-tags`.

### AWS CLI

Para ver una lista de las etiquetas adjuntas a un MFA dispositivo

El siguiente `list-mfa-device-tags` comando recupera la lista de etiquetas asociadas al MFA dispositivo especificado.

```

aws iam list-mfa-device-tags \
  --serial-number arn:aws:iam::123456789012:mfa/alice

```

Salida:

```

{
  "Tags": [
    {
      "Key": "DeptID",
      "Value": "123456"
    },
  ],
}

```

```
{
  "Key": "Department",
  "Value": "Accounting"
}
]
```

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListMfaDeviceTags](#) la Referencia de AWS CLI comandos.

## list-mfa-devices

En el siguiente ejemplo de código, se muestra cómo usar `list-mfa-devices`.

### AWS CLI

Para enumerar todos MFA los dispositivos de un usuario específico

En este ejemplo, se muestran detalles sobre el MFA dispositivo asignado al IAM usuario Bob.

```
aws iam list-mfa-devices \
  --user-name Bob
```

Salida:

```
{
  "MFADevices": [
    {
      "UserName": "Bob",
      "SerialNumber": "arn:aws:iam::123456789012:mfa/Bob",
      "EnableDate": "2019-10-28T20:37:09+00:00"
    },
    {
      "UserName": "Bob",
      "SerialNumber": "GAKT12345678",
      "EnableDate": "2023-02-18T21:44:42+00:00"
    },
    {
      "UserName": "Bob",
```

```

        "SerialNumber": "arn:aws:iam::123456789012:u2f/user/Bob/
fidosecuritykey1-7XNL7NFNLZ123456789EXAMPLE",
        "EnableDate": "2023-09-19T02:25:35+00:00"
    },
    {
        "UserName": "Bob",
        "SerialNumber": "arn:aws:iam::123456789012:u2f/user/Bob/
fidosecuritykey2-VDRQTDBBN5123456789EXAMPLE",
        "EnableDate": "2023-09-19T01:49:18+00:00"
    }
]
}

```

Para obtener más información, consulte [Uso de la autenticación multifactorial \(MFA\) AWS en la Guía del AWS IAM usuario](#).

- Para API obtener más información, consulte [ListMfaDevices](#) la Referencia de AWS CLI comandos.

## list-open-id-connect-provider-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-open-id-connect-provider-tags`.

### AWS CLI

Para enumerar las etiquetas adjuntas a un proveedor de identidad compatible con OpenID Connect (OIDC)

El siguiente `list-open-id-connect-provider-tags` comando recupera la lista de etiquetas asociadas al proveedor de identidad especificado OIDC.

```

aws iam list-open-id-connect-provider-tags \
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
server.example.com

```

Salida:

```

{
  "Tags": [
    {
      "Key": "DeptID",

```

```
        "Value": "123456"
      },
      {
        "Key": "Department",
        "Value": "Accounting"
      }
    ]
  }
}
```

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAMusuario.

- Para API obtener más información, consulte [ListOpenIdConnectProviderTags](#)la Referencia de AWS CLI comandos.

## **list-open-id-connect-providers**

En el siguiente ejemplo de código, se muestra cómo usar `list-open-id-connect-providers`.

### AWS CLI

Para incluir información sobre los proveedores de OpenID Connect de la cuenta AWS

Este ejemplo devuelve una lista ARNS de todos los proveedores de OpenID Connect que están definidos en la cuenta corriente AWS .

```
aws iam list-open-id-connect-providers
```

Salida:

```
{
  "OpenIDConnectProviderList": [
    {
      "Arn": "arn:aws:iam::123456789012:oidc-provider/
example.oidcprovider.com"
    }
  ]
}
```

Para obtener más información, consulte [Creación de proveedores de identidad de OpenID Connect \(OIDC\)](#) en la Guía del AWS IAMusuario.

- Para API obtener más información, consulte [ListOpenIdConnectProviders](#) la Referencia de AWS CLI comandos.

## list-policies-granting-service-access

En el siguiente ejemplo de código, se muestra cómo usar `list-policies-granting-service-access`.

### AWS CLI

Para enumerar las políticas que otorgan un acceso principal al servicio especificado

En el siguiente `list-policies-granting-service-access` ejemplo, se recupera la lista de políticas que otorgan al IAM usuario `sofia` acceso al AWS CodeCommit servicio.

```
aws iam list-policies-granting-service-access \
  --arn arn:aws:iam::123456789012:user/sofia \
  --service-namespaces codecommit
```

Salida:

```
{
  "PoliciesGrantingServiceAccess": [
    {
      "ServiceNamespace": "codecommit",
      "Policies": [
        {
          "PolicyName": "Grant-Sofia-Access-To-CodeCommit",
          "PolicyType": "INLINE",
          "EntityType": "USER",
          "EntityName": "sofia"
        }
      ]
    }
  ],
  "IsTruncated": false
}
```

Para obtener más información, consulte [Uso IAM con CodeCommit: credenciales, SSH claves y claves de AWS acceso de Git](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListPoliciesGrantingServiceAccess](#) la Referencia de AWS CLI comandos.

## list-policies

En el siguiente ejemplo de código, se muestra cómo usar `list-policies`.

### AWS CLI

Para ver una lista de las políticas administradas que están disponibles para su AWS cuenta

En este ejemplo, se devuelve un conjunto de las dos primeras políticas administradas disponibles en la AWS cuenta corriente.

```
aws iam list-policies \  
  --max-items 3
```

Salida:

```
{  
  "Policies": [  
    {  
      "PolicyName": "AWSCloudTrailAccessPolicy",  
      "PolicyId": "ANPAXQE2B5PJ7YEXAMPLE",  
      "Arn": "arn:aws:iam::123456789012:policy/AWSCloudTrailAccessPolicy",  
      "Path": "/",  
      "DefaultVersionId": "v1",  
      "AttachmentCount": 0,  
      "PermissionsBoundaryUsageCount": 0,  
      "IsAttachable": true,  
      "CreateDate": "2019-09-04T17:43:42+00:00",  
      "UpdateDate": "2019-09-04T17:43:42+00:00"  
    },  
    {  
      "PolicyName": "AdministratorAccess",  
      "PolicyId": "ANPAIWMBCKSKIEE64ZLYK",  
      "Arn": "arn:aws:iam::aws:policy/AdministratorAccess",  
      "Path": "/",  
      "DefaultVersionId": "v1",  
      "AttachmentCount": 6,  
      "PermissionsBoundaryUsageCount": 0,  
      "IsAttachable": true,  
    }  
  ]  
}
```

```

        "CreateDate": "2015-02-06T18:39:46+00:00",
        "UpdateDate": "2015-02-06T18:39:46+00:00"
    },
    {
        "PolicyName": "PowerUserAccess",
        "PolicyId": "ANPAJYRXTHIB4FOVS3ZXS",
        "Arn": "arn:aws:iam::aws:policy/PowerUserAccess",
        "Path": "/",
        "DefaultVersionId": "v5",
        "AttachmentCount": 1,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2015-02-06T18:39:47+00:00",
        "UpdateDate": "2023-07-06T22:04:00+00:00"
    }
],
"NextToken": "EXAMPLErZXIi0iBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQi0iA4fQ=="
}

```

Para obtener más información, consulte [Políticas y permisos IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListPolicies](#) la Referencia de AWS CLI comandos.

## list-policy-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-policy-tags`.

### AWS CLI

Para enumerar las etiquetas adjuntas a una política gestionada

El siguiente `list-policy-tags` comando recupera la lista de etiquetas asociadas a la política administrada especificada.

```

aws iam list-policy-tags \
  --policy-arn arn:aws:iam::123456789012:policy/billing-access

```

Salida:

```

{
  "Tags": [
    {

```

```
        "Key": "DeptID",
        "Value": "123456"
    },
    {
        "Key": "Department",
        "Value": "Accounting"
    }
]
}
```

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListPolicyTags](#) la Referencia de AWS CLI comandos.

## list-policy-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-policy-versions`.

### AWS CLI

Cómo enumerar información sobre las versiones de la política administrada especificada

Este ejemplo devuelve la lista de versiones disponibles de la política cuya política ARN es `arn:aws:iam::123456789012:policy/MySamplePolicy`.

```
aws iam list-policy-versions \
  --policy-arn arn:aws:iam::123456789012:policy/MySamplePolicy
```

Salida:

```
{
  "IsTruncated": false,
  "Versions": [
    {
      "VersionId": "v2",
      "IsDefaultVersion": true,
      "CreateDate": "2015-06-02T23:19:44Z"
    },
    {
      "VersionId": "v1",
```



```
    "IsDefaultVersion": false,  
    "CreateDate": "2015-06-02T22:30:47Z"  
  }  
]  
}
```

Para obtener más información, consulte [Políticas y permisos IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListPolicyVersions](#) la Referencia de AWS CLI comandos.

## **list-role-policies**

En el siguiente ejemplo de código, se muestra cómo usar `list-role-policies`.

### AWS CLI

Para enumerar las políticas asociadas a un IAM rol

El siguiente `list-role-policies` comando muestra los nombres de las políticas de permisos del IAM rol especificado.

```
aws iam list-role-policies \  
  --role-name Test-Role
```

Salida:

```
{  
  "PolicyNames": [  
    "ExamplePolicy"  
  ]  
}
```

Para ver la política de confianza asociada a un rol, utilice el comando `get-role`. Para ver los detalles de una política de permisos, utilice el comando `get-role-policy`.

Para obtener más información, consulte [Creación de IAM funciones](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListRolePolicies](#) la Referencia de AWS CLI comandos.

## list-role-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-role-tags`.

### AWS CLI

Cómo enumerar las etiquetas adjuntas a un rol

El siguiente comando `list-role-tags` recupera la lista de etiquetas asociadas al rol especificado.

```
aws iam list-role-tags \  
  --role-name production-role
```

Salida:

```
{  
  "Tags": [  
    {  
      "Key": "Department",  
      "Value": "Accounting"  
    },  
    {  
      "Key": "DeptID",  
      "Value": "12345"  
    }  
  ],  
  "IsTruncated": false  
}
```

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListRoleTags](#) la Referencia de AWS CLI comandos.

## list-roles

En el siguiente ejemplo de código, se muestra cómo usar `list-roles`.

### AWS CLI

Para enumerar IAM las funciones de la cuenta corriente

El siguiente `list-roles` comando muestra las IAM funciones de la cuenta actual.

```
aws iam list-roles
```

Salida:

```
{
  "Roles": [
    {
      "Path": "/",
      "RoleName": "ExampleRole",
      "RoleId": "AR0AJ520TH4H7LEXAMPLE",
      "Arn": "arn:aws:iam::123456789012:role/ExampleRole",
      "CreateDate": "2017-09-12T19:23:36+00:00",
      "AssumeRolePolicyDocument": {
        "Version": "2012-10-17",
        "Statement": [
          {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
              "Service": "ec2.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
          }
        ]
      },
      "MaxSessionDuration": 3600
    },
    {
      "Path": "/example_path/",
      "RoleName": "ExampleRoleWithPath",
      "RoleId": "AROAI4QRP7UFT7EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:role/example_path/
ExampleRoleWithPath",
      "CreateDate": "2023-09-21T20:29:38+00:00",
      "AssumeRolePolicyDocument": {
        "Version": "2012-10-17",
        "Statement": [
          {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
```

```

        "Service": "ec2.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
},
"MaxSessionDuration": 3600
}
]
}

```

Para obtener más información, consulte [Creación de IAM roles](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListRoles](#) la Referencia de AWS CLI comandos.

## list-saml-provider-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-saml-provider-tags`.

### AWS CLI

Para ver una lista de las etiquetas adjuntas a un SAML proveedor

El siguiente `list-saml-provider-tags` comando recupera la lista de etiquetas asociadas al SAML proveedor especificado.

```

aws iam list-saml-provider-tags \
  --saml-provider-arn arn:aws:iam::123456789012:saml-provider/ADFS

```

Salida:

```

{
  "Tags": [
    {
      "Key": "DeptID",
      "Value": "123456"
    },
    {
      "Key": "Department",
      "Value": "Accounting"
    }
  ]
}

```

```
}
```

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListSamlProviderTags](#) la Referencia de AWS CLI comandos.

## **list-saml-providers**

En el siguiente ejemplo de código, se muestra cómo usar `list-saml-providers`.

### AWS CLI

Para enumerar los SAML proveedores de la AWS cuenta

En este ejemplo, se recupera la lista de proveedores SAML 2.0 creada en la AWS cuenta corriente.

```
aws iam list-saml-providers
```

Salida:

```
{
  "SAMLProviderList": [
    {
      "Arn": "arn:aws:iam::123456789012:saml-provider/SAML-ADFS",
      "ValidUntil": "2015-06-05T22:45:14Z",
      "CreateDate": "2015-06-05T22:45:14Z"
    }
  ]
}
```

Para obtener más información, consulte [Creación de proveedores de IAM SAML identidad](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListSAMLProviders](#) en la Referencia de AWS CLI comandos.

## **list-server-certificate-tags**

En el siguiente ejemplo de código, se muestra cómo usar `list-server-certificate-tags`.

## AWS CLI

Para ver una lista de las etiquetas adjuntas a un certificado de servidor

El siguiente `list-server-certificate-tags` comando recupera la lista de etiquetas asociadas al certificado de servidor especificado.

```
aws iam list-server-certificate-tags \  
  --server-certificate-name ExampleCertificate
```

Salida:

```
{  
  "Tags": [  
    {  
      "Key": "DeptID",  
      "Value": "123456"  
    },  
    {  
      "Key": "Department",  
      "Value": "Accounting"  
    }  
  ]  
}
```

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListServerCertificateTags](#) la Referencia de AWS CLI comandos.

## **list-server-certificates**

En el siguiente ejemplo de código, se muestra cómo usar `list-server-certificates`.

## AWS CLI

Para ver una lista de los certificados de servidor de su AWS cuenta

El siguiente `list-server-certificates` comando muestra todos los certificados de servidor almacenados y disponibles para su uso en su AWS cuenta.

**aws iam list-server-certificates**

Salida:

```
{
  "ServerCertificateMetadataList": [
    {
      "Path": "/",
      "ServerCertificateName": "myUpdatedServerCertificate",
      "ServerCertificateId": "ASCAEXAMPLE123EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:server-certificate/
myUpdatedServerCertificate",
      "UploadDate": "2019-04-22T21:13:44+00:00",
      "Expiration": "2019-10-15T22:23:16+00:00"
    },
    {
      "Path": "/cloudfront/",
      "ServerCertificateName": "MyTestCert",
      "ServerCertificateId": "ASCAEXAMPLE456EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:server-certificate/0rg1/0rg2/
MyTestCert",
      "UploadDate": "2015-04-21T18:14:16+00:00",
      "Expiration": "2018-01-14T17:52:36+00:00"
    }
  ]
}
```

Para obtener más información, consulte [Administrar los certificados de servidor IAM en la Guía del AWS IAM usuario](#).

- Para API obtener más información, consulte [ListServerCertificates](#) la Referencia de AWS CLI comandos.

**list-service-specific-credential**

En el siguiente ejemplo de código, se muestra cómo usar `list-service-specific-credential`.

AWS CLI

Ejemplo 1: Listar las credenciales específicas del servicio para un usuario

El siguiente `list-service-specific-credentials` ejemplo muestra todas las credenciales específicas del servicio asignadas al usuario especificado. Las contraseñas no se incluyen en la respuesta.

```
aws iam list-service-specific-credentials \  
  --user-name sofia
```

Salida:

```
{  
  "ServiceSpecificCredential": {  
    "CreateDate": "2019-04-18T20:45:36+00:00",  
    "ServiceName": "codecommit.amazonaws.com",  
    "ServiceUserName": "sofia-at-123456789012",  
    "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",  
    "UserName": "sofia",  
    "Status": "Active"  
  }  
}
```

Ejemplo 2: Enumere las credenciales específicas del servicio de un usuario filtrado a un servicio específico

En el siguiente `list-service-specific-credentials` ejemplo, se muestran las credenciales específicas del servicio asignadas al usuario que realiza la solicitud. La lista se filtra para incluir solo las credenciales del servicio especificado. Las contraseñas no se incluyen en la respuesta.

```
aws iam list-service-specific-credentials \  
  --service-name codecommit.amazonaws.com
```

Salida:

```
{  
  "ServiceSpecificCredential": {  
    "CreateDate": "2019-04-18T20:45:36+00:00",  
    "ServiceName": "codecommit.amazonaws.com",  
    "ServiceUserName": "sofia-at-123456789012",  
    "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",  
    "UserName": "sofia",
```



```

    "Status": "Active"
  }
}

```

Para obtener más información, consulta [Crear credenciales de Git para HTTPS las conexiones CodeCommit](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [ListServiceSpecificCredential](#) la Referencia de AWS CLI comandos.

## list-service-specific-credentials

En el siguiente ejemplo de código, se muestra cómo usar `list-service-specific-credentials`.

### AWS CLI

Para recuperar una lista de credenciales

En el siguiente `list-service-specific-credentials` ejemplo, se enumeran las credenciales generadas para el HTTPS acceso a AWS CodeCommit los repositorios de un usuario denominado `developer`.

```

aws iam list-service-specific-credentials \
  --user-name developer \
  --service-name codecommit.amazonaws.com

```

Salida:

```

{
  "ServiceSpecificCredentials": [
    {
      "UserName": "developer",
      "Status": "Inactive",
      "ServiceUserName": "developer-at-123456789012",
      "CreateDate": "2019-10-01T04:31:41Z",
      "ServiceSpecificCredentialId": "ACCAQF0DXMPL4YFHP7DZE",
      "ServiceName": "codecommit.amazonaws.com"
    },
    {
      "UserName": "developer",

```

```

        "Status": "Active",
        "ServiceUserName": "developer+1-at-123456789012",
        "CreateDate": "2019-10-01T04:31:45Z",
        "ServiceSpecificCredentialId": "ACCAQFOX MPL6VW57M7AJP",
        "ServiceName": "codecommit.amazonaws.com"
    }
]
}

```

Para obtener más información, consulta [Crear credenciales de Git para HTTPS las conexiones CodeCommit](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [ListServiceSpecificCredentials](#) la Referencia de AWS CLI comandos.

## list-signing-certificates

En el siguiente ejemplo de código, se muestra cómo usar `list-signing-certificates`.

### AWS CLI

Para ver una lista de los certificados de firma de un IAM usuario

El siguiente `list-signing-certificates` comando muestra los certificados de firma del IAM usuario nombrado `Bob`.

```
aws iam list-signing-certificates \
  --user-name Bob
```

Salida:

```

{
  "Certificates": [
    {
      "UserName": "Bob",
      "Status": "Inactive",
      "CertificateBody": "-----BEGIN CERTIFICATE-----<certificate-body>-----
END CERTIFICATE-----",
      "CertificateId": "TA7SMP42TDN5Z260BPJE7EXAMPLE",
      "UploadDate": "2013-06-06T21:40:08Z"
    }
  ]
}

```

```
]
}
```

Para obtener más información, consulta [Administrar certificados de firma](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [ListSigningCertificates](#) la Referencia de AWS CLI comandos.

## list-ssh-public-keys

En el siguiente ejemplo de código, se muestra cómo usar `list-ssh-public-keys`.

### AWS CLI

Para ver una lista de las claves SSH públicas adjuntas a un IAM usuario

En el siguiente `list-ssh-public-keys` ejemplo, se enumeran las claves SSH públicas adjuntas al IAM usuario `sofia`.

```
aws iam list-ssh-public-keys \
  --user-name sofia
```

Salida:

```
{
  "SSHPublicKeys": [
    {
      "UserName": "sofia",
      "SSHPublicKeyId": "APKA1234567890EXAMPLE",
      "Status": "Inactive",
      "UploadDate": "2019-04-18T17:04:49+00:00"
    }
  ]
}
```

Para obtener más información, consulte [Uso y SSH uso de SSH las claves CodeCommit](#) en la Guía del AWS IAM usuario

- Para API obtener más información, consulte [ListSshPublicKeys](#) la Referencia de AWS CLI comandos.

## list-user-policies

En el siguiente ejemplo de código, se muestra cómo usar `list-user-policies`.

### AWS CLI

Para enumerar las políticas de un IAM usuario

El siguiente `list-user-policies` comando muestra las políticas asociadas al IAM usuario nombrado `Bob`.

```
aws iam list-user-policies \  
  --user-name Bob
```

Salida:

```
{  
  "PolicyNames": [  
    "ExamplePolicy",  
    "TestPolicy"  
  ]  
}
```

Para obtener más información, consulte [Crear un IAM usuario en su AWS cuenta](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListUserPolicies](#) la Referencia de AWS CLI comandos.

## list-user-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-user-tags`.

### AWS CLI

Cómo enumerar las etiquetas adjuntas a un usuario

El siguiente `list-user-tags` comando recupera la lista de etiquetas asociadas al IAM usuario especificado.

```
aws iam list-user-tags \  
  --user-name Bob
```

```
--user-name alice
```

Salida:

```
{
  "Tags": [
    {
      "Key": "Department",
      "Value": "Accounting"
    },
    {
      "Key": "DeptID",
      "Value": "12345"
    }
  ],
  "IsTruncated": false
}
```

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListUserTags](#) la Referencia de AWS CLI comandos.

## list-users

En el siguiente ejemplo de código, se muestra cómo usar `list-users`.

AWS CLI

Para enumerar IAM los usuarios

El siguiente `list-users` comando muestra los IAM usuarios de la cuenta actual.

```
aws iam list-users
```

Salida:

```
{
  "Users": [
    {
      "UserName": "Adele",
```

```
    "Path": "/",
    "CreateDate": "2013-03-07T05:14:48Z",
    "UserId": "AKIAI44QH8DHBEXAMPLE",
    "Arn": "arn:aws:iam::123456789012:user/Adele"
  },
  {
    "UserName": "Bob",
    "Path": "/",
    "CreateDate": "2012-09-21T23:03:13Z",
    "UserId": "AKIAIOSFODNN7EXAMPLE",
    "Arn": "arn:aws:iam::123456789012:user/Bob"
  }
]
}
```

Para obtener más información, consulte [Listar IAM los usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListUsers](#) la Referencia de AWS CLI comandos.

## **list-virtual-mfa-devices**

En el siguiente ejemplo de código, se muestra cómo usar `list-virtual-mfa-devices`.

### AWS CLI

Para enumerar MFA los dispositivos virtuales

El siguiente `list-virtual-mfa-devices` comando muestra los MFA dispositivos virtuales que se han configurado para la cuenta actual.

```
aws iam list-virtual-mfa-devices
```

Salida:

```
{
  "VirtualMFADevices": [
    {
      "SerialNumber": "arn:aws:iam::123456789012:mfa/ExampleMFADevice"
    },
    {
      "SerialNumber": "arn:aws:iam::123456789012:mfa/Fred"
    }
  ]
}
```

```
}
```

Para obtener más información, consulte [Habilitar un dispositivo virtual de autenticación multifactor \(MFA\)](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListVirtualMfaDevices](#) la Referencia de AWS CLI comandos.

## **put-group-policy**

En el siguiente ejemplo de código, se muestra cómo usar `put-group-policy`.

### AWS CLI

Cómo agregar una política a un grupo

El siguiente `put-group-policy` comando agrega una política al IAM grupo denominado `Admins`.

```
aws iam put-group-policy \  
  --group-name Admins \  
  --policy-document file://AdminPolicy.json \  
  --policy-name AdminRoot
```

Este comando no genera ninguna salida.

La política se define como un JSON documento del `AdminPolicyarchivo.json`. (El nombre y la extensión del archivo no son significativos).

Para obtener más información, consulte [Administración de IAM políticas](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [PutGroupPolicy](#) la Referencia de AWS CLI comandos.

## **put-role-permissions-boundary**

En el siguiente ejemplo de código, se muestra cómo usar `put-role-permissions-boundary`.

### AWS CLI

Ejemplo 1: aplicar a un IAM rol un límite de permisos basado en una política personalizada

En el siguiente `put-role-permissions-boundary` ejemplo, se aplica la política personalizada `intern-boundary` denominada límite de permisos para el IAM rol especificado.

```
aws iam put-role-permissions-boundary \  
  --permissions-boundary arn:aws:iam::123456789012:policy/intern-boundary \  
  --role-name lambda-application-role
```

Este comando no genera ninguna salida.

Ejemplo 2: Para aplicar a un IAM rol un límite de permisos basado en una política AWS administrada

El siguiente `put-role-permissions-boundary` ejemplo aplica la `PowerUserAccess` política AWS administrada como límite de permisos para el IAM rol especificado.

```
aws iam put-role-permissions-boundary \  
  --permissions-boundary arn:aws:iam::aws:policy/PowerUserAccess \  
  --role-name x-account-admin
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Modificación de un rol](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [PutRolePermissionsBoundary](#) la Referencia de AWS CLI comandos.

## **put-role-policy**

En el siguiente ejemplo de código, se muestra cómo usar `put-role-policy`.

### AWS CLI

Para adjuntar una política de permisos a un IAM rol

El siguiente comando `put-role-policy` agrega una política de permisos al rol denominado `Test-Role`.

```
aws iam put-role-policy \  
  --role-name Test-Role \  
  --policy-name ExamplePolicy \  
  --policy-document ExamplePolicyDocument
```



```
--policy-document file://AdminPolicy.json
```

Este comando no genera ninguna salida.

La política se define como un JSON documento del AdminPolicyarchivo.json. (El nombre y la extensión del archivo no son significativos).

Para asociar una política de confianza a un rol, utilice el comando `update-assume-role-policy`.

Para obtener más información, consulte [Modificación de un rol](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [PutRolePolicy](#) la Referencia de AWS CLI comandos.

## **put-user-permissions-boundary**

En el siguiente ejemplo de código, se muestra cómo usar `put-user-permissions-boundary`.

### AWS CLI

Ejemplo 1: aplicar a un IAM usuario un límite de permisos basado en una política personalizada

En el siguiente `put-user-permissions-boundary` ejemplo, se aplica una política personalizada `intern-boundary` denominada límite de permisos para el IAM usuario especificado.

```
aws iam put-user-permissions-boundary \  
  --permissions-boundary arn:aws:iam::123456789012:policy/intern-boundary \  
  --user-name intern
```

Este comando no genera ninguna salida.

Ejemplo 2: Para aplicar a un IAM usuario un límite de permisos basado en una política AWS administrada

El siguiente `put-user-permissions-boundary` ejemplo aplica la política AWS administrada `PowerUserAccess` denominada límite de permisos para el usuario especificado IAM.

```
aws iam put-user-permissions-boundary \  
  --permissions-boundary arn:aws:iam::123456789012:policy/PowerUserAccess \  
  --user-name intern
```

```
--permissions-boundary arn:aws:iam::aws:policy/PowerUserAccess \  
--user-name developer
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir y eliminar permisos de IAM identidad](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [PutUserPermissionsBoundary](#) la Referencia de AWS CLI comandos.

## put-user-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-user-policy`.

### AWS CLI

Para adjuntar una política a un IAM usuario

El siguiente `put-user-policy` comando adjunta una política al IAM usuario nombrado `Bob`.

```
aws iam put-user-policy \  
  --user-name Bob \  
  --policy-name ExamplePolicy \  
  --policy-document file://AdminPolicy.json
```

Este comando no genera ninguna salida.

La política se define como un JSON documento del `AdminPolicyarchivo.json`. (El nombre y la extensión del archivo no son significativos).

Para obtener más información, consulte [Añadir y eliminar permisos de IAM identidad](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [PutUserPolicy](#) la Referencia de AWS CLI comandos.

## remove-client-id-from-open-id-connect-provider

En el siguiente ejemplo de código, se muestra cómo usar `remove-client-id-from-open-id-connect-provider`.

## AWS CLI

Para eliminar el ID de cliente especificado de la lista de clientes IDs registrados para el proveedor de IAM OpenID Connect especificado

En este ejemplo, se elimina el ID `My-TestApp-3` de cliente de la lista de clientes IDs asociados al IAM OIDC proveedor del que ARN se encuentra `arn:aws:iam::123456789012:oidc-provider/example.oidcprovider.com`.

```
aws iam remove-client-id-from-open-id-connect-provider \
  --client-id My-TestApp-3 \
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
example.oidcprovider.com
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de proveedores de identidad de OpenID Connect \(OIDC\)](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [RemoveClientIdFromOpenIdConnectProvider](#) la Referencia de AWS CLI comandos.

## **remove-role-from-instance-profile**

En el siguiente ejemplo de código, se muestra cómo usar `remove-role-from-instance-profile`.

## AWS CLI

Cómo eliminar un rol de un perfil de instancia

El siguiente comando `remove-role-from-instance-profile` elimina el rol llamado `Test-Role` del perfil de instancia denominado `ExampleInstanceProfile`.

```
aws iam remove-role-from-instance-profile \
  --instance-profile-name ExampleInstanceProfile \
  --role-name Test-Role
```

Para obtener más información, consulte [Uso de perfiles de instancia](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [RemoveRoleFromInstanceProfile](#) la Referencia de AWS CLI comandos.

## **remove-user-from-group**

En el siguiente ejemplo de código, se muestra cómo usar `remove-user-from-group`.

### AWS CLI

Para eliminar un usuario de un IAM grupo

El siguiente `remove-user-from-group` comando elimina el usuario nombrado Bob del IAM grupo denominado Admins.

```
aws iam remove-user-from-group \  
  --user-name Bob \  
  --group-name Admins
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir y eliminar usuarios de un grupo IAM de usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [RemoveUserFromGroup](#) la Referencia de AWS CLI comandos.

## **reset-service-specific-credential**

En el siguiente ejemplo de código, se muestra cómo usar `reset-service-specific-credential`.

### AWS CLI

Ejemplo 1: restablecer la contraseña de una credencial específica del servicio adjunta al usuario que realiza la solicitud

El siguiente `reset-service-specific-credential` ejemplo genera una nueva contraseña criptográficamente segura para la credencial específica del servicio especificada adjunta al usuario que realiza la solicitud.

```
aws iam reset-service-specific-credential \  
  --user-name Bob \  
  --group-name Admins \  
  --service-name AmazonS3 \  
  --password-length 16
```

```
--service-specific-credential-id ACCAEXAMPLE123EXAMPLE
```

Salida:

```
{
  "ServiceSpecificCredential": {
    "CreateDate": "2019-04-18T20:45:36+00:00",
    "ServiceName": "codecommit.amazonaws.com",
    "ServiceUserName": "sofia-at-123456789012",
    "ServicePassword": "+oaFsNk7tLco+C/obP9Ghhc0zGcK0ayTmE3LnAmAmH4=",
    "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",
    "UserName": "sofia",
    "Status": "Active"
  }
}
```

Ejemplo 2: restablecer la contraseña de una credencial específica del servicio adjunta a un usuario específico

El siguiente `reset-service-specific-credential` ejemplo genera una nueva contraseña segura desde el punto de vista criptográfico para una credencial específica del servicio adjunta al usuario especificado.

```
aws iam reset-service-specific-credential \  
  --user-name sofia \  
  --service-specific-credential-id ACCAEXAMPLE123EXAMPLE
```

Salida:

```
{
  "ServiceSpecificCredential": {
    "CreateDate": "2019-04-18T20:45:36+00:00",
    "ServiceName": "codecommit.amazonaws.com",
    "ServiceUserName": "sofia-at-123456789012",
    "ServicePassword": "+oaFsNk7tLco+C/obP9Ghhc0zGcK0ayTmE3LnAmAmH4=",
    "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",
    "UserName": "sofia",
    "Status": "Active"
  }
}
```

Para obtener más información, consulta [Crear credenciales de Git para HTTPS las conexiones CodeCommit](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [ResetServiceSpecificCredential](#) la Referencia de AWS CLI comandos.

## **resync-mfa-device**

En el siguiente ejemplo de código, se muestra cómo usar `resync-mfa-device`.

### AWS CLI

Para sincronizar un dispositivo MFA

El siguiente `resync-mfa-device` ejemplo sincroniza el MFA dispositivo que está asociado al IAM usuario Bob y cuyo dispositivo ARN está asociado `arn:aws:iam::123456789012:mfa/BobsMFADevice` con un programa de autenticación que proporcionó los dos códigos de autenticación.

```
aws iam resync-mfa-device \  
  --user-name Bob \  
  --serial-number arn:aws:iam::210987654321:mfa/BobsMFADevice \  
  --authentication-code1 123456 \  
  --authentication-code2 987654
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de la autenticación multifactorial \(MFA\) AWS en la Guía del AWS IAM usuario](#).

- Para API obtener más información, consulte [ResyncMfaDevice](#) la Referencia de AWS CLI comandos.

## **set-default-policy-version**

En el siguiente ejemplo de código, se muestra cómo usar `set-default-policy-version`.

### AWS CLI

Cómo establecer la versión especificada de la política especificada como la versión predeterminada de la política.

En este ejemplo, se establece la v2 versión de la política `arn:aws:iam::123456789012:policy/MyPolicy` que ARN es la versión activa predeterminada.

```
aws iam set-default-policy-version \  
  --policy-arn arn:aws:iam::123456789012:policy/MyPolicy \  
  --version-id v2
```

Para obtener más información, consulte [Políticas y permisos IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [SetDefaultPolicyVersion](#) la Referencia de AWS CLI comandos.

## **set-security-token-service-preferences**

En el siguiente ejemplo de código, se muestra cómo usar `set-security-token-service-preferences`.

AWS CLI

Para configurar la versión del token de punto final global

El siguiente `set-security-token-service-preferences` ejemplo configura Amazon STS para que utilice los tokens de la versión 2 al autenticarse en el punto final global.

```
aws iam set-security-token-service-preferences \  
  --global-endpoint-token-version v2Token
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar AWS STS en una AWS región en la](#) Guía del AWS IAM usuario.

- Para API obtener más información, consulte [SetSecurityTokenServicePreferences](#) la Referencia de AWS CLI comandos.

## **simulate-custom-policy**

En el siguiente ejemplo de código, se muestra cómo usar `simulate-custom-policy`.

## AWS CLI

Ejemplo 1: simular los efectos de todas IAM las políticas asociadas a un IAM usuario o rol

A continuación, se `simulate-custom-policy` muestra cómo proporcionar la política y definir los valores de las variables y simular una API llamada para ver si está permitida o denegada. El siguiente ejemplo muestra una política que permite el acceso a la base de datos solo después de una fecha y hora especificadas. La simulación se realiza correctamente porque las acciones simuladas y la `aws:CurrentTime` variable especificada cumplen todos los requisitos de la política.

```
aws iam simulate-custom-policy \
  --policy-input-list '{"Version":"2012-10-17","Statement":
{"Effect":"Allow","Action":"dynamodb:*","Resource":"*","Condition":
{"DateGreaterThan":{"aws:CurrentTime":"2018-08-16T12:00:00Z"}}}' \
  --action-names dynamodb>CreateBackup \
  --context-
entries "ContextKeyName='aws:CurrentTime',ContextKeyValues='2019-04-25T11:00:00Z',ContextKey
```

Salida:

```
{
  "EvaluationResults": [
    {
      "EvalActionName": "dynamodb>CreateBackup",
      "EvalResourceName": "*",
      "EvalDecision": "allowed",
      "MatchedStatements": [
        {
          "SourcePolicyId": "PolicyInputList.1",
          "StartPosition": {
            "Line": 1,
            "Column": 38
          },
          "EndPosition": {
            "Line": 1,
            "Column": 167
          }
        }
      ],
      "MissingContextValues": []
    }
  ]
}
```



```
]
}
```

## Ejemplo 2: simular un comando prohibido por la política

El siguiente `simulate-custom-policy` ejemplo muestra los resultados de la simulación de un comando prohibido por la política. En este ejemplo, la fecha proporcionada es anterior a la requerida por la condición de la política.

```
aws iam simulate-custom-policy \
  --policy-input-list '{"Version":"2012-10-17","Statement":
{"Effect":"Allow","Action":"dynamodb:*","Resource":"*","Condition":
{"DateGreaterThan":{"aws:CurrentTime":"2018-08-16T12:00:00Z"}}}' \
  --action-names dynamodb:CreateBackup \
  --context-
entries "ContextKeyName='aws:CurrentTime',ContextKeyValues='2014-04-25T11:00:00Z',ContextKey
```

Salida:

```
{
  "EvaluationResults": [
    {
      "EvalActionName": "dynamodb:CreateBackup",
      "EvalResourceName": "*",
      "EvalDecision": "implicitDeny",
      "MatchedStatements": [],
      "MissingContextValues": []
    }
  ]
}
```

Para obtener más información, consulte [Probar IAM políticas con el simulador IAM de políticas](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [SimulateCustomPolicy](#) la Referencia de AWS CLI comandos.

## `simulate-principal-policy`

En el siguiente ejemplo de código, se muestra cómo usar `simulate-principal-policy`.

## AWS CLI

### Ejemplo 1: simular los efectos de una IAM política arbitraria

A continuación, se `simulate-principal-policy` muestra cómo simular que un usuario solicita una API acción y cómo determinar si las políticas asociadas a ese usuario permiten o deniegan la acción. En el siguiente ejemplo, el usuario tiene una política que solo permite la `codecommit:ListRepositories` acción.

```
aws iam simulate-principal-policy \  
  --policy-source-arn arn:aws:iam::123456789012:user/alejandro \  
  --action-names codecommit:ListRepositories
```

Salida:

```
{  
  "EvaluationResults": [  
    {  
      "EvalActionName": "codecommit:ListRepositories",  
      "EvalResourceName": "*",  
      "EvalDecision": "allowed",  
      "MatchedStatements": [  
        {  
          "SourcePolicyId": "Grant-Access-To-CodeCommit-ListRepo",  
          "StartPosition": {  
            "Line": 3,  
            "Column": 19  
          },  
          "EndPosition": {  
            "Line": 9,  
            "Column": 10  
          }  
        }  
      ],  
      "MissingContextValues": []  
    }  
  ]  
}
```

### Ejemplo 2: simular los efectos de un comando prohibido

El siguiente `simulate-custom-policy` ejemplo muestra los resultados de la simulación de un comando prohibido por una de las políticas del usuario. En el ejemplo siguiente, el usuario tiene una política que permite el acceso a una base de datos de DynamoDB solo después de una fecha y hora determinadas. En la simulación, el usuario intenta acceder a la base de datos con un `aws:CurrentTime` valor anterior a lo que permite la condición de la política.

```
aws iam simulate-principal-policy \  
  --policy-source-arn arn:aws:iam::123456789012:user/alejandro \  
  --action-names dynamodb:CreateBackup \  
  --context-  
entries "ContextKeyName='aws:CurrentTime',ContextKeyValues='2018-04-25T11:00:00Z',ContextKey
```

Salida:

```
{  
  "EvaluationResults": [  
    {  
      "EvalActionName": "dynamodb:CreateBackup",  
      "EvalResourceName": "*",  
      "EvalDecision": "implicitDeny",  
      "MatchedStatements": [],  
      "MissingContextValues": []  
    }  
  ]  
}
```

Para obtener más información, consulte [Probar IAM políticas con el simulador IAM de políticas](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [SimulatePrincipalPolicy](#) la Referencia de AWS CLI comandos.

## tag-instance-profile

En el siguiente ejemplo de código, se muestra cómo usar `tag-instance-profile`.

### AWS CLI

Para añadir una etiqueta a un perfil de instancia

El siguiente `tag-instance-profile` comando agrega una etiqueta con el nombre de un departamento al perfil de instancia especificado.

```
aws iam tag-instance-profile \  
  --instance-profile-name deployment-role \  
  --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [TagInstanceProfile](#) la Referencia de AWS CLI comandos.

## tag-mfa-device

En el siguiente ejemplo de código, se muestra cómo usar tag-mfa-device.

### AWS CLI

Para añadir una etiqueta a un MFA dispositivo

El siguiente tag-mfa-device comando agrega una etiqueta con el nombre de un departamento al MFA dispositivo especificado.

```
aws iam tag-mfa-device \  
  --serial-number arn:aws:iam::123456789012:mfa/alice \  
  --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [TagMfaDevice](#) la Referencia de AWS CLI comandos.

## tag-open-id-connect-provider

En el siguiente ejemplo de código, se muestra cómo usar tag-open-id-connect-provider.

### AWS CLI

Para añadir una etiqueta a un proveedor de identidad compatible con OpenID Connect (OIDC)

El siguiente `tag-open-id-connect-provider` comando agrega una etiqueta con el nombre de un departamento al proveedor de OIDC identidad especificado.

```
aws iam tag-open-id-connect-provider \  
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/  
server.example.com \  
  --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [TagOpenIdConnectProvider](#) la Referencia de AWS CLI comandos.

## tag-policy

En el siguiente ejemplo de código, se muestra cómo usar `tag-policy`.

### AWS CLI

Para añadir una etiqueta a una política gestionada por el cliente

El siguiente `tag-policy` comando agrega una etiqueta con el nombre de un departamento a la política administrada por el cliente especificada.

```
aws iam tag-policy \  
  --policy-arn arn:aws:iam::123456789012:policy/billing-access \  
  --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [TagPolicy](#) la Referencia de AWS CLI comandos.

## tag-role

En el siguiente ejemplo de código, se muestra cómo usar `tag-role`.

## AWS CLI

Cómo añadir una etiqueta a un rol

El siguiente comando `tag-role` agrega una etiqueta con el nombre de un departamento al rol especificado.

```
aws iam tag-role --role-name my-role \  
  --tags '{"Key": "Department", "Value": "Accounting"}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [TagRole](#) la Referencia de AWS CLI comandos.

## tag-saml-provider

En el siguiente ejemplo de código, se muestra cómo usar `tag-saml-provider`.

## AWS CLI

Para añadir una etiqueta a un SAML proveedor

El siguiente `tag-saml-provider` comando agrega una etiqueta con el nombre de un departamento al SAML proveedor especificado.

```
aws iam tag-saml-provider \  
  --saml-provider-arn arn:aws:iam::123456789012:saml-provider/ADFS \  
  --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [TagSamlProvider](#) la Referencia de AWS CLI comandos.

## tag-server-certificate

En el siguiente ejemplo de código, se muestra cómo usar `tag-server-certificate`.

### AWS CLI

Para añadir una etiqueta a un certificado de servidor

El siguiente `tag-saml-provider` comando agrega una etiqueta con el nombre de un departamento al certificado de servidor especificado.

```
aws iam tag-server-certificate \  
  --server-certificate-name ExampleCertificate \  
  --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [TagServerCertificate](#) la Referencia de AWS CLI comandos.

## tag-user

En el siguiente ejemplo de código, se muestra cómo usar `tag-user`.

### AWS CLI

Cómo añadir una etiqueta a un usuario

El siguiente comando `tag-user` agrega una etiqueta con el departamento asociado al usuario especificado.

```
aws iam tag-user \  
  --user-name alice \  
  --tags '{"Key": "Department", "Value": "Accounting"}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [IAM los recursos de etiquetado](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [TagUser](#) la Referencia de AWS CLI comandos.

## untag-instance-profile

En el siguiente ejemplo de código, se muestra cómo usar `untag-instance-profile`.

### AWS CLI

Para eliminar una etiqueta de un perfil de instancia

El siguiente `untag-instance-profile` comando elimina cualquier etiqueta con el nombre clave «Departamento» del perfil de instancia especificado.

```
aws iam untag-instance-profile \  
  --instance-profile-name deployment-role \  
  --tag-keys Department
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UntagInstanceProfile](#) la Referencia de AWS CLI comandos.

## untag-mfa-device

En el siguiente ejemplo de código, se muestra cómo usar `untag-mfa-device`.

### AWS CLI

Para eliminar una etiqueta de un MFA dispositivo

El siguiente `untag-mfa-device` comando elimina cualquier etiqueta con el nombre clave «Departamento» del MFA dispositivo especificado.

```
aws iam untag-mfa-device \  
  --serial-number arn:aws:iam::123456789012:mfa/alice \  
  --tag-keys Department
```

Este comando no genera ninguna salida.



Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UntagMfaDevice](#) la Referencia de AWS CLI comandos.

## **untag-open-id-connect-provider**

En el siguiente ejemplo de código, se muestra cómo usar `untag-open-id-connect-provider`.

### AWS CLI

Para eliminar una etiqueta de un proveedor OIDC de identidad

El siguiente `untag-open-id-connect-provider` comando elimina cualquier etiqueta con el nombre clave «Departamento» del proveedor de OIDC identidad especificado.

```
aws iam untag-open-id-connect-provider \  
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/  
server.example.com \  
  --tag-keys Department
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UntagOpenIdConnectProvider](#) la Referencia de AWS CLI comandos.

## **untag-policy**

En el siguiente ejemplo de código, se muestra cómo usar `untag-policy`.

### AWS CLI

Para eliminar una etiqueta de una política gestionada por el cliente

El siguiente `untag-policy` comando elimina cualquier etiqueta con el nombre clave «Departamento» de la política gestionada por el cliente especificada.

```
aws iam untag-policy \  

```

```
--policy-arn arn:aws:iam::452925170507:policy/billing-access \  
--tag-keys Department
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UntagPolicy](#) la Referencia de AWS CLI comandos.

## untag-role

En el siguiente ejemplo de código, se muestra cómo usar `untag-role`.

### AWS CLI

Cómo eliminar una etiqueta de un rol

El siguiente comando `untag-role` elimina cualquier etiqueta con el nombre clave «Departamento» del rol especificado.

```
aws iam untag-role \  
--role-name my-role \  
--tag-keys Department
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [IAM los recursos de etiquetado](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UntagRole](#) la Referencia de AWS CLI comandos.

## untag-saml-provider

En el siguiente ejemplo de código, se muestra cómo usar `untag-saml-provider`.

### AWS CLI

Para eliminar una etiqueta de un SAML proveedor

El siguiente `untag-saml-provider` comando elimina cualquier etiqueta con el nombre clave «Departamento» del perfil de instancia especificado.

```
aws iam untag-saml-provider \  
  --saml-provider-arn arn:aws:iam::123456789012:saml-provider/ADFS \  
  --tag-keys Department
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UntagSamlProvider](#) la Referencia de AWS CLI comandos.

## untag-server-certificate

En el siguiente ejemplo de código, se muestra cómo usar `untag-server-certificate`.

### AWS CLI

Para eliminar una etiqueta de un certificado de servidor

El siguiente `untag-server-certificate` comando elimina cualquier etiqueta con el nombre clave «Departamento» del certificado de servidor especificado.

```
aws iam untag-server-certificate \  
  --server-certificate-name ExampleCertificate \  
  --tag-keys Department
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de IAM recursos](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UntagServerCertificate](#) la Referencia de AWS CLI comandos.

## untag-user

En el siguiente ejemplo de código, se muestra cómo usar `untag-user`.

### AWS CLI

Cómo eliminar una etiqueta de un usuario

El siguiente comando `untag-user` elimina cualquier etiqueta con el nombre clave “Departamento” del usuario especificado.

```
aws iam untag-user \  
  --user-name alice \  
  --tag-keys Department
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [IAM los recursos de etiquetado](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UntagUser](#) la Referencia de AWS CLI comandos.

## update-access-key

En el siguiente ejemplo de código, se muestra cómo usar `update-access-key`.

### AWS CLI

Para activar o desactivar una clave de acceso para un usuario IAM

El siguiente `update-access-key` comando desactiva la clave de acceso especificada (identificador de clave de acceso y clave de acceso secreta) para el IAM usuario nombrado. Bob

```
aws iam update-access-key \  
  --access-key-id AKIAIOSFODNN7EXAMPLE \  
  --status Inactive \  
  --user-name Bob
```

Este comando no genera ninguna salida.

Si se desactiva la clave, no se puede utilizar para acceder a ella mediante programación. AWS Sin embargo, la clave sigue disponible y se puede reactivar.

Para obtener más información, consulte [Administrar las claves de acceso para IAM los usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateAccessKey](#) la Referencia de AWS CLI comandos.

## update-account-password-policy

En el siguiente ejemplo de código, se muestra cómo usar `update-account-password-policy`.

### AWS CLI

Cómo ver la política de contraseñas de la cuenta actual

El siguiente comando `update-account-password-policy` establece que la política de contraseñas requiera una longitud mínima de ocho caracteres y que la contraseña contenga uno o más números.

```
aws iam update-account-password-policy \  
  --minimum-password-length 8 \  
  --require-numbers
```

Este comando no genera ninguna salida.

Los cambios en la política de contraseñas de una cuenta afectan a cualquier contraseña nueva que se cree para IAM los usuarios de la cuenta. Los cambios en la política de contraseñas no afectan a las contraseñas existentes.

Para obtener más información, consulte [Establecer una política de contraseñas de cuentas para IAM los usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateAccountPasswordPolicy](#) la Referencia de AWS CLI comandos.

## update-assume-role-policy

En el siguiente ejemplo de código, se muestra cómo usar `update-assume-role-policy`.

### AWS CLI

Para actualizar la política de confianza de un IAM rol

El siguiente comando `update-assume-role-policy` actualiza la política de confianza del rol denominado `Test-Role`.

```
aws iam update-assume-role-policy \  
  --role-name Test-Role \  
  --policy-document file:///Test-Role-Trust-Policy.json
```

Este comando no genera ninguna salida.

La política de confianza se define como un JSON documento del archivo `test-role-trust-policy.json`. (El nombre y la extensión del archivo no son significativos). La política de confianza debe especificar una entidad principal.

Utilice el comando `put-role-policy` para actualizar la política de permisos de un rol.

[Para obtener más información, consulte Creación de funciones en la Guía del usuario. IAM AWS IAM](#)

- Para API obtener más información, consulte [UpdateAssumeRolePolicy](#) la Referencia de AWS CLI comandos.

## update-group

En el siguiente ejemplo de código, se muestra cómo usar `update-group`.

AWS CLI

Para cambiar el nombre de un grupo IAM

El siguiente `update-group` comando cambia el nombre del IAM grupo `Test` a `Test-1`.

```
aws iam update-group \  
  --group-name Test \  
  --new-group-name Test-1
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cambiar el nombre de un grupo IAM de usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateGroup](#) la Referencia de AWS CLI comandos.

## update-login-profile

En el siguiente ejemplo de código, se muestra cómo usar `update-login-profile`.

AWS CLI

Para actualizar la contraseña de un IAM usuario

El siguiente `update-login-profile` comando crea una contraseña nueva para el IAM usuario nombrado `Bob`.

```
aws iam update-login-profile \  
  --user-name Bob \  
  --password <password>
```

Este comando no genera ninguna salida.

Para establecer una política de contraseñas para la cuenta, use el comando `update-account-password-policy`. Si la nueva contraseña infringe la política de contraseñas de la cuenta, el comando devuelve un error de `PasswordPolicyViolation`.

Si la política de contraseñas de la cuenta lo permite, IAM los usuarios pueden cambiar sus propias contraseñas mediante el `change-password` comando.

Guarde la nueva contraseña en un lugar seguro. Si se pierde la contraseña, no se puede recuperar y debe crear una nueva con el comando `create-login-profile`.

Para obtener más información, consulte [Administrar las contraseñas de IAM los usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateLoginProfile](#) la Referencia de AWS CLI comandos.

## **update-open-id-connect-provider-thumbprint**

En el siguiente ejemplo de código, se muestra cómo usar `update-open-id-connect-provider-thumbprint`.

### AWS CLI

Cómo reemplazar la lista existente de huellas digitales de certificados de servidor por una nueva lista

En este ejemplo, se actualiza la lista de huellas digitales del certificado del OIDC proveedor que ARN va `arn:aws:iam::123456789012:oidc-provider/example.oidcprovider.com` a utilizar una nueva huella digital.

```
aws iam update-open-id-connect-provider-thumbprint \  
  --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/  
example.oidcprovider.com \  
  --thumbprint <thumbprint>
```

```
--thumbprint-list 7359755EXAMPLEabc3060bce3EXAMPLEec4542a3
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de proveedores de identidad de OpenID Connect \(OIDC\)](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateOpenIdConnectProviderThumbprint](#) la Referencia de AWS CLI comandos.

## update-role-description

En el siguiente ejemplo de código, se muestra cómo usar `update-role-description`.

### AWS CLI

Para cambiar la descripción de un IAM rol

El siguiente `update-role` comando cambia la descripción del IAM rol `production-role` a `Main production role`.

```
aws iam update-role-description \  
  --role-name production-role \  
  --description 'Main production role'
```

Salida:

```
{  
  "Role": {  
    "Path": "/",  
    "RoleName": "production-role",  
    "RoleId": "ARO0A1234567890EXAMPLE",  
    "Arn": "arn:aws:iam::123456789012:role/production-role",  
    "CreateDate": "2017-12-06T17:16:37+00:00",  
    "AssumeRolePolicyDocument": {  
      "Version": "2012-10-17",  
      "Statement": [  
        {  
          "Effect": "Allow",  
          "Principal": {  
            "AWS": "arn:aws:iam::123456789012:root"  
          }  
        }  
      ],  
    }  
  }  
}
```



```
        "Action": "sts:AssumeRole",
        "Condition": {}
      }
    ]
  },
  "Description": "Main production role"
}
```

Para obtener más información, consulte [Modificación de un rol](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateRoleDescription](#) la Referencia de AWS CLI comandos.

## update-role

En el siguiente ejemplo de código, se muestra cómo usar `update-role`.

### AWS CLI

Para cambiar la descripción de un IAM rol o la duración de la sesión

El siguiente `update-role` comando cambia la descripción del IAM rol `production-role` a 12 horas `Main production role` y establece la duración máxima de la sesión en 12 horas.

```
aws iam update-role \
  --role-name production-role \
  --description 'Main production role' \
  --max-session-duration 43200
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Modificación de un rol](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateRole](#) la Referencia de AWS CLI comandos.

## update-saml-provider

En el siguiente ejemplo de código, se muestra cómo usar `update-saml-provider`.

### AWS CLI

Para actualizar el documento de metadatos de un SAML proveedor existente

En este ejemplo, se actualiza el SAML proveedor IAM en el ARN que se encuentra `arn:aws:iam::123456789012:saml-provider/SAMLADFS` con un nuevo documento de SAML metadatos del archivo `SAMLMetaData.xml`.

```
aws iam update-saml-provider \  
  --saml-metadata-document file://SAMLMetaData.xml \  
  --saml-provider-arn arn:aws:iam::123456789012:saml-provider/SAMLADFS
```

Salida:

```
{  
  "SAMLProviderArn": "arn:aws:iam::123456789012:saml-provider/SAMLADFS"  
}
```

Para obtener más información, consulte [Creación de proveedores de IAM SAML identidad](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateSamlProvider](#) la Referencia de AWS CLI comandos.

## update-server-certificate

En el siguiente ejemplo de código, se muestra cómo usar `update-server-certificate`.

AWS CLI

Para cambiar la ruta o el nombre de un certificado de servidor de su AWS cuenta

El siguiente comando `update-server-certificate` cambia el nombre del certificado de `myServerCertificate` a `myUpdatedServerCertificate`. También cambia la ruta para `/cloudfront/` que el CloudFront servicio de Amazon pueda acceder a ella. Este comando no genera ninguna salida. Puede ver los resultados de la actualización al ejecutar el comando `list-server-certificates`.

```
aws-iam update-server-certificate \  
  --server-certificate-name myServerCertificate \  
  --new-server-certificate-name myUpdatedServerCertificate \  
  --new-path /cloudfront/
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión de certificados de servidor IAM en](#) la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateServerCertificate](#) la Referencia de AWS CLI comandos.

## **update-service-specific-credential**

En el siguiente ejemplo de código, se muestra cómo usar `update-service-specific-credential`.

### AWS CLI

Ejemplo 1: Para actualizar el estado de la credencial específica del servicio del usuario solicitante

El siguiente `update-service-specific-credential` ejemplo cambia el estado de la credencial especificada para el usuario al que realiza la solicitud. `Inactive`

```
aws iam update-service-specific-credential \  
  --service-specific-credential-id ACCAEXAMPLE123EXAMPLE \  
  --status Inactive
```

Este comando no genera ninguna salida.

Ejemplo 2: Para actualizar el estado de la credencial específica del servicio de un usuario específico

En el siguiente `update-service-specific-credential` ejemplo, se cambia el estado de la credencial del usuario especificado a `Inactivo`.

```
aws iam update-service-specific-credential \  
  --user-name sofia \  
  --service-specific-credential-id ACCAEXAMPLE123EXAMPLE \  
  --status Inactive
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Crear credenciales de Git para HTTPS las conexiones a CodeCommit](#) en la Guía del AWS CodeCommit usuario

- Para API obtener más información, consulte [UpdateServiceSpecificCredential](#) la Referencia de AWS CLI comandos.

## update-signing-certificate

En el siguiente ejemplo de código, se muestra cómo usar `update-signing-certificate`.

### AWS CLI

Para activar o desactivar un certificado de firma para un usuario IAM

El siguiente `update-signing-certificate` comando desactiva el certificado de firma especificado para el IAM usuario nombrado. Bob

```
aws iam update-signing-certificate \  
  --certificate-id TA7SMP42TDN5Z260BPJE7EXAMPLE \  
  --status Inactive \  
  --user-name Bob
```

Para obtener el ID de un certificado de firma, use el comando `list-signing-certificates`.

Para obtener más información, consulta [Administrar certificados de firma](#) en la Guía del EC2 usuario de Amazon.

- Para API obtener más información, consulte [UpdateSigningCertificate](#) la Referencia de AWS CLI comandos.

## update-ssh-public-key

En el siguiente ejemplo de código, se muestra cómo usar `update-ssh-public-key`.

### AWS CLI

Para cambiar el estado de una clave SSH pública

El siguiente `update-ssh-public-key` comando cambia el estado de la clave pública especificada a `Inactive`.

```
aws iam update-ssh-public-key \  
  --user-name sofia \  
  --ssh-public-key-id APKA1234567890EXAMPLE \  
  --status Inactive
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso y SSH uso de SSH las teclas CodeCommit](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateSshPublicKey](#) la Referencia de AWS CLI comandos.

## update-user

En el siguiente ejemplo de código, se muestra cómo usar `update-user`.

### AWS CLI

Para cambiar el nombre de un IAM usuario

El siguiente `update-user` comando cambia el nombre del IAM usuario Bob aRobert.

```
aws iam update-user \  
  --user-name Bob \  
  --new-user-name Robert
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cambiar el nombre de un grupo IAM de usuarios](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateUser](#) la Referencia de AWS CLI comandos.

## upload-server-certificate

En el siguiente ejemplo de código, se muestra cómo usar `upload-server-certificate`.

### AWS CLI

Para cargar un certificado de servidor en su AWS cuenta

El siguiente `upload-server-certificate` comando carga un certificado de servidor en su AWS cuenta. En este ejemplo, el certificado está en el archivo `public_key_cert_file.pem`, la clave privada asociada está en el archivo `my_private_key.pem` y la cadena de certificados proporcionada por la entidad de certificación (CA) está en el archivo `my_certificate_chain_file.pem`. Cuando el archivo haya terminado de cargarse, estará disponible con ese nombre. `myServerCertificate` Los parámetros que comienzan con `file://`

indican al comando que lea el contenido del archivo y lo use como valor del parámetro en lugar del nombre del archivo en sí.

```
aws iam upload-server-certificate \  
  --server-certificate-name myServerCertificate \  
  --certificate-body file://public_key_cert_file.pem \  
  --private-key file://my_private_key.pem \  
  --certificate-chain file://my_certificate_chain_file.pem
```

Salida:

```
{  
  "ServerCertificateMetadata": {  
    "Path": "/",  
    "ServerCertificateName": "myServerCertificate",  
    "ServerCertificateId": "ASCAEXAMPLE123EXAMPLE",  
    "Arn": "arn:aws:iam::1234567989012:server-certificate/myServerCertificate",  
    "UploadDate": "2019-04-22T21:13:44+00:00",  
    "Expiration": "2019-10-15T22:23:16+00:00"  
  }  
}
```

Para obtener más información, consulte Creación, carga y eliminación de certificados de servidor en la guía de uso IAM.

- Para API obtener más información, consulte [UploadServerCertificate](#) la Referencia de AWS CLI comandos.

## upload-signing-certificate

En el siguiente ejemplo de código, se muestra cómo usar `upload-signing-certificate`.

AWS CLI

Para cargar un certificado de firma para un IAM usuario

El siguiente `upload-signing-certificate` comando carga un certificado de firma para el IAM usuario nombrado Bob.

```
aws iam upload-signing-certificate \  
  --user-name Bob \  
  --certificate-body file://my_certificate.pem \  
  --private-key file://my_private_key.pem
```

```
--certificate-body file://certificate.pem
```

Salida:

```
{
  "Certificate": {
    "UserName": "Bob",
    "Status": "Active",
    "CertificateBody": "-----BEGIN CERTIFICATE-----<certificate-body>-----END
CERTIFICATE-----",
    "CertificateId": "TA7SMP42TDN5Z260BPJE7EXAMPLE",
    "UploadDate": "2013-06-06T21:40:08.121Z"
  }
}
```

El certificado está en un archivo con el formato certificate.pem. PEM

Para obtener más información, consulte [Creación y carga de un certificado de firma de usuario](#) en la guía de uso. IAM

- Para API obtener más información, consulte [UploadSigningCertificate](#) la Referencia de AWS CLI comandos.

## upload-ssh-public-key

En el siguiente ejemplo de código, se muestra cómo usar upload-ssh-public-key.

AWS CLI

Para cargar una clave SSH pública y asociarla a un usuario

El siguiente upload-ssh-public-key comando carga la clave pública que se encuentra en el archivo sshkey.pub y la adjunta al usuario. sofia

```
aws iam upload-ssh-public-key \
  --user-name sofia \
  --ssh-public-key-body file://sshkey.pub
```

Salida:

```
{
```

```
"SSHPublicKey": {
  "UserName": "sofia",
  "SSHPublicKeyId": "APKA1234567890EXAMPLE",
  "Fingerprint": "12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef",
  "SSHPublicKeyBody": "ssh-rsa <long string generated by ssh-keygen
command>>",
  "Status": "Active",
  "UploadDate": "2019-04-18T17:04:49+00:00"
}
```

Para obtener más información sobre cómo generar claves en un formato adecuado para este comando, consulte [SSHLinux, macOS o Unix: configurar las claves públicas y privadas para Git CodeCommit](#) o [SSHWindows: configurar las claves públicas y privadas para Git y CodeCommit](#) en la Guía del AWS CodeCommit usuario.

- Para API obtener más información, consulte [UploadSshPublicKey](#) la Referencia de AWS CLI comandos.

## IAMEjemplos de Access Analyzer utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el AWS Command Line Interface uso de IAM Access Analyzer.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

#### **apply-archive-rule**

En el siguiente ejemplo de código, se muestra cómo usar `apply-archive-rule`.



## AWS CLI

Para aplicar una regla de archivado a los hallazgos existentes que cumplan con los criterios de la regla de archivado

En el siguiente `apply-archive-rule` ejemplo, se aplica una regla de archivado a los hallazgos existentes que cumplen los criterios de la regla de archivado.

```
aws accessanalyzer apply-archive-rule \  
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
UnusedAccess-ConsoleAnalyzer-organization \  
  --rule-name MyArchiveRule
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [las reglas de archivado](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ApplyArchiveRule](#) la Referencia de AWS CLI comandos.

## cancel-policy-generation

En el siguiente ejemplo de código, se muestra cómo usar `cancel-policy-generation`.

### AWS CLI

Para cancelar la generación de políticas solicitada

El siguiente `cancel-policy-generation` ejemplo cancela el identificador del trabajo de generación de políticas solicitado.

```
aws accessanalyzer cancel-policy-generation \  
  --job-id 923a56b0-ebb8-4e80-8a3c-a11ccfbcd6f2
```

Este comando no genera ninguna salida.

Para obtener más información, consulte la sección [Generación de políticas de IAM Access Analyzer](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [CancelPolicyGeneration](#) la Referencia de AWS CLI comandos.

## check-access-not-granted

En el siguiente ejemplo de código, se muestra cómo usar `check-access-not-granted`.

### AWS CLI

Para comprobar si una política no permite el acceso especificado

En el siguiente `check-access-not-granted` ejemplo, se comprueba si una política no permite el acceso especificado.

```
aws accessanalyzer check-access-not-granted \  
  --policy-document file://myfile.json \  
  --access actions="s3:DeleteBucket","s3:GetBucketLocation" \  
  --policy-type IDENTITY_POLICY
```

Contenidos de `myfile.json`:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "s3:GetObject",  
        "s3:ListBucket"  
      ],  
      "Resource": [  
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET",  
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"  
      ]  
    }  
  ]  
}
```

Salida:

```
{  
  "result": "PASS",  
  "message": "The policy document does not grant access to perform the listed  
actions."
```

```
}
```

Para obtener más información, consulte Vista [previa del acceso con IAM Access Analyzer APIs](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte la Referencia [CheckAccessNotGranted](#) de AWS CLI comandos.

## check-no-new-access

En el siguiente ejemplo de código, se muestra cómo usar `check-no-new-access`.

### AWS CLI

Para comprobar si se permite un nuevo acceso a una política actualizada en comparación con la política existente

El siguiente `check-no-new-access` ejemplo comprueba si se permite un nuevo acceso para una política actualizada en comparación con la política existente.

```
aws accessanalyzer check-no-new-access \  
  --existing-policy-document file://existing-policy.json \  
  --new-policy-document file://new-policy.json \  
  --policy-type IDENTITY_POLICY
```

Contenidos de `existing-policy.json`:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "s3:GetObject",  
        "s3:ListBucket"  
      ],  
      "Resource": [  
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET",  
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"  
      ]  
    }  
  ]  
}
```

```
]
}
```

Contenidos de `new-policy.json`:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:GetObjectAcl",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
      ]
    }
  ]
}
```

Salida:

```
{
  "result": "FAIL",
  "message": "The modified permissions grant new access compared to your existing policy.",
  "reasons": [
    {
      "description": "New access in the statement with index: 0.",
      "statementIndex": 0
    }
  ]
}
```

Para obtener más información, consulte Vista [previa del acceso con IAM Access Analyzer APIs](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte la Referencia [CheckNoNewAccess](#) de AWS CLI comandos.

## create-access-preview

En el siguiente ejemplo de código, se muestra cómo usar `create-access-preview`.

### AWS CLI

Para crear una vista previa de acceso que le permita obtener una vista previa de los resultados de IAM Access Analyzer para su recurso antes de implementar los permisos del recurso

En el siguiente `create-access-preview` ejemplo, se crea una vista previa del acceso que le permite obtener una vista previa de los resultados de IAM Access Analyzer para su recurso antes de implementar los permisos de recursos en su AWS cuenta.

```
aws accessanalyzer create-access-preview \  
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-account \  
  --configurations file://myfile.json
```

Contenidos de `myfile.json`:

```
{  
  "arn:aws:s3:::DOC-EXAMPLE-BUCKET": {  
    "s3Bucket": {  
      "bucketPolicy": "{\n\"Version\":\n\"2012-10-17\",\n\"Statement\":[\n{\n\"Effect\n\":\n\"Allow\",\n\"Principal\":{\n\"AWS\":[\n\"arn:aws:iam::111122223333:root\n\"}],\n\"Action\n\":[\n\"s3:PutObject\",\n\"s3:PutObjectAcl\n\"],\n\"Resource\n\":\n\"arn:aws:s3:::DOC-EXAMPLE-  
BUCKET/*\n\"}}]",  
      "bucketPublicAccessBlock": {  
        "ignorePublicAcls": true,  
        "restrictPublicBuckets": true  
      },  
      "bucketAclGrants": [  
        {  
          "grantee": {  
            "id":  
"79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be"  
          },  
          "permission": "READ"  
        }  
      ]  
    }  
  }  
}
```

```
}
```

Salida:

```
{  
  "id": "3c65eb13-6ef9-4629-8919-a32043619e6b"  
}
```

Para obtener más información, consulte Vista [previa del acceso con IAM Access Analyzer APIs](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte la Referencia [CreateAccessPreview](#) de AWS CLI comandos.

## create-analyzer

En el siguiente ejemplo de código, se muestra cómo usar `create-analyzer`.

AWS CLI

Para crear un analizador

En el siguiente `create-analyzer` ejemplo, se crea un analizador en su cuenta. AWS

```
aws accessanalyzer create-analyzer \  
  --analyzer-name example \  
  --type ACCOUNT
```

Salida:

```
{  
  "arn": "arn:aws:access-analyzer:us-east-2:111122223333:analyzer/example"  
}
```

Para obtener más información, consulte [Introducción a AWS Identity and Access Management Access Analyzer en la](#) Guía del AWS IAM usuario.

- Para API obtener más información, consulte [CreateAnalyzer](#) la Referencia de AWS CLI comandos.

## create-archive-rule

En el siguiente ejemplo de código, se muestra cómo usar `create-archive-rule`.

### AWS CLI

Para crear una regla de archivado para el analizador especificado

En el siguiente `create-archive-rule` ejemplo, se crea una regla de archivo para el analizador especificado en su AWS cuenta.

```
aws accessanalyzer create-archive-rule \  
  --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \  
  --rule-name MyRule \  
  --filter '{"resource": {"contains": ["Cognito"]}, "resourceType": {"eq":  
  ["AWS::IAM::Role"]}}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [las reglas de archivado](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [CreateArchiveRule](#) la Referencia de AWS CLI comandos.

## delete-analyzer

En el siguiente ejemplo de código, se muestra cómo usar `delete-analyzer`.

### AWS CLI

Para eliminar el analizador especificado

En el siguiente `delete-analyzer` ejemplo, se elimina el analizador especificado de su cuenta.

AWS

```
aws accessanalyzer delete-analyzer \  
  --analyzer-name example
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [las reglas de archivado](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteAnalyzer](#) la Referencia de AWS CLI comandos.

## delete-archive-rule

En el siguiente ejemplo de código, se muestra cómo usar `delete-archive-rule`.

### AWS CLI

Para eliminar la regla de archivado especificada

En el siguiente `delete-archive-rule` ejemplo, se elimina la regla de archivo especificada de su AWS cuenta.

```
aws accessanalyzer delete-archive-rule \  
  --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \  
  --rule-name MyRule
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [las reglas de archivado](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DeleteArchiveRule](#) la Referencia de AWS CLI comandos.

## get-access-preview

En el siguiente ejemplo de código, se muestra cómo usar `get-access-preview`.

### AWS CLI

Para recuperar información sobre una vista previa de acceso al analizador especificado

En el siguiente `get-access-preview` ejemplo, se recupera información sobre una vista previa del acceso al analizador especificado en su cuenta. AWS

```
aws accessanalyzer get-access-preview \  
  --access-preview-id 3c65eb13-6ef9-4629-8919-a32043619e6b \  
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-account
```

Salida:



```

{
  "accessPreview": {
    "id": "3c65eb13-6ef9-4629-8919-a32043619e6b",
    "analyzerArn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account",
    "configurations": {
      "arn:aws:s3:::DOC-EXAMPLE-BUCKET": {
        "s3Bucket": {
          "bucketPolicy": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\n\"Effect\":"Allow\", \"Principal\":{\n\"AWS\":[\n\"arn:aws:iam::111122223333:root\"
],\n\"Action\":[\n\"s3:PutObject\", \"s3:PutObjectAcl\"],\n\"Resource\":\n\"arn:aws:s3:::DOC-
EXAMPLE-BUCKET/*\"}]}",
          "bucketAclGrants": [
            {
              "permission": "READ",
              "grantee": {
                "id":
"79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be"
              }
            }
          ],
          "bucketPublicAccessBlock": {
            "ignorePublicAcls": true,
            "restrictPublicBuckets": true
          }
        }
      }
    },
    "createdAt": "2024-02-17T00:18:44+00:00",
    "status": "COMPLETED"
  }
}

```

Para obtener más información, consulte Vista [previa del acceso con IAM Access Analyzer APIs](#) en la Guía del usuario.AWS IAM

- Para API obtener más información, consulte la Referencia [GetAccessPreview](#) de AWS CLI comandos.

## get-analyzed-resource

En el siguiente ejemplo de código, se muestra cómo usar get-analyzed-resource.

## AWS CLI

Para recuperar información sobre un recurso que se analizó

En el siguiente `get-analyzed-resource` ejemplo, se recupera información sobre un recurso que se analizó en su AWS cuenta.

```
aws accessanalyzer get-analyzed-resource \  
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-account \  
  --resource-arn arn:aws:s3:::DOC-EXAMPLE-BUCKET
```

Salida:

```
{  
  "resource": {  
    "analyzedAt": "2024-02-15T18:01:53.002000+00:00",  
    "isPublic": false,  
    "resourceArn": "arn:aws:s3:::DOC-EXAMPLE-BUCKET",  
    "resourceOwnerAccount": "111122223333",  
    "resourceType": "AWS::S3::Bucket"  
  }  
}
```

Para obtener más información, consulte [Uso del analizador de acceso a AWS Identity and Access Management](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetAnalyzedResource](#) la Referencia de AWS CLI comandos.

## get-analyzer

En el siguiente ejemplo de código, se muestra cómo usar `get-analyzer`.

## AWS CLI

Para recuperar información sobre el analizador especificado

En el siguiente `get-analyzer` ejemplo, se recupera información sobre el analizador especificado en su cuenta. AWS

```
aws accessanalyzer get-analyzer \  
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-account
```

```
--analyzer-name ConsoleAnalyzer-account
```

Salida:

```
{
  "analyzer": {
    "arn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account",
    "createdAt": "2019-12-03T07:28:17+00:00",
    "lastResourceAnalyzed": "arn:aws:sns:us-west-2:111122223333:config-topic",
    "lastResourceAnalyzedAt": "2024-02-15T18:01:53.003000+00:00",
    "name": "ConsoleAnalyzer-account",
    "status": "ACTIVE",
    "tags": {
      "auto-delete": "no"
    },
    "type": "ACCOUNT"
  }
}
```

Para obtener más información, consulte [Uso del analizador de acceso a AWS Identity and Access Management](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetAnalyzer](#) la Referencia de AWS CLI comandos.

## get-archive-rule

En el siguiente ejemplo de código, se muestra cómo usar `get-archive-rule`.

AWS CLI

Para recuperar información sobre una regla de archivado

En el siguiente `get-archive-rule` ejemplo, se recupera información sobre una regla de archivo de su AWS cuenta.

```
aws accessanalyzer get-archive-rule \
  --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \
  --rule-name MyArchiveRule
```

Salida:

```
{
  "archiveRule": {
    "createdAt": "2024-02-15T00:49:27+00:00",
    "filter": {
      "resource": {
        "contains": [
          "Cognito"
        ]
      },
      "resourceType": {
        "eq": [
          "AWS::IAM::Role"
        ]
      }
    },
    "ruleName": "MyArchiveRule",
    "updatedAt": "2024-02-15T00:49:27+00:00"
  }
}
```

Para obtener más información, consulte [las reglas de archivado](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetArchiveRule](#) la Referencia de AWS CLI comandos.

## get-finding-v2

En el siguiente ejemplo de código, se muestra cómo usar `get-finding-v2`.

### AWS CLI

Para recuperar información sobre el hallazgo especificado

En el siguiente `get-finding-v2` ejemplo, se recupera información sobre el hallazgo especificado en su AWS cuenta.

```
aws accessanalyzer get-finding-v2 \
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-organization \
  --id 0910eedb-381e-4e95-adda-0d25c19e6e90
```

Salida:

```
{
  "findingDetails": [
    {
      "externalAccessDetails": {
        "action": [
          "sts:AssumeRoleWithWebIdentity"
        ],
        "condition": {
          "cognito-identity.amazonaws.com:aud": "us-
west-2:EXAMPLE0-0000-0000-0000-000000000000"
        },
        "isPublic": false,
        "principal": {
          "Federated": "cognito-identity.amazonaws.com"
        }
      }
    }
  ],
  "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role",
  "status": "ACTIVE",
  "error": null,
  "createdAt": "2021-02-26T21:17:50.905000+00:00",
  "resourceType": "AWS::IAM::Role",
  "findingType": "ExternalAccess",
  "resourceOwnerAccount": "111122223333",
  "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
  "id": "0910eedb-381e-4e95-adda-0d25c19e6e90",
  "updatedAt": "2021-02-26T21:17:50.905000+00:00"
}
```

Para obtener más información, consulte [Revisión de los resultados](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte la sección [GetFindingV2](#) en la Referencia de AWS CLI comandos.

## get-finding

En el siguiente ejemplo de código, se muestra cómo usar `get-finding`.

### AWS CLI

Para recuperar información sobre el hallazgo especificado

En el siguiente `get-finding` ejemplo, se recupera información sobre el hallazgo especificado en su AWS cuenta.

```
aws accessanalyzer get-finding \  
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-organization \  
  --id 0910eedb-381e-4e95-adda-0d25c19e6e90
```

Salida:

```
{  
  "finding": {  
    "id": "0910eedb-381e-4e95-adda-0d25c19e6e90",  
    "principal": {  
      "Federated": "cognito-identity.amazonaws.com"  
    },  
    "action": [  
      "sts:AssumeRoleWithWebIdentity"  
    ],  
    "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role",  
    "isPublic": false,  
    "resourceType": "AWS::IAM::Role",  
    "condition": {  
      "cognito-identity.amazonaws.com:aud": "us-  
west-2:EXAMPLE0-0000-0000-0000-000000000000"  
    },  
    "createdAt": "2021-02-26T21:17:50.905000+00:00",  
    "analyzedAt": "2024-02-16T18:17:47.888000+00:00",  
    "updatedAt": "2021-02-26T21:17:50.905000+00:00",  
    "status": "ACTIVE",  
    "resourceOwnerAccount": "111122223333"  
  }  
}
```

Para obtener más información, consulte [Revisión de los resultados](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetFinding](#) la Referencia de AWS CLI comandos.

## get-generated-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-generated-policy`.

## AWS CLI

Para recuperar la política que se generó con el `StartPolicyGeneration` API

En el siguiente `get-generated-policy` ejemplo, se recupera la política que se generó con la `StartPolicyGeneration` API de su AWS cuenta.

```
aws accessanalyzer get-generated-policy \
  --job-id c557dc4a-0338-4489-95dd-739014860ff9
```

Salida:

```
{
  "generatedPolicyResult": {
    "generatedPolicies": [
      {
        "policy": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\n\"Sid\":\n\"SupportedServiceSid0\", \"Effect\": \"Allow\", \"Action\":
[\"access-analyzer:GetAnalyzer\", \"access-analyzer:ListAnalyzers\",
\n\"access-analyzer:ListArchiveRules\", \"access-analyzer:ListFindings
\n\", \"cloudtrail:DescribeTrails\", \"cloudtrail:GetEventDataStore\",
\n\"cloudtrail:GetEventSelectors\", \"cloudtrail:GetInsightSelectors
\n\", \"cloudtrail:GetTrailStatus\", \"cloudtrail:ListChannels\",
\n\"cloudtrail:ListEventDataStores\", \"cloudtrail:ListQueries\", \"cloudtrail:ListTags
\n\", \"cloudtrail:LookupEvents\", \"ec2:DescribeRegions\", \"iam:GetAccountSummary
\n\", \"iam:GetOpenIDConnectProvider\", \"iam:GetRole\", \"iam:ListAccessKeys\",
\n\"iam:ListAccountAliases\", \"iam:ListOpenIDConnectProviders\", \"iam:ListRoles
\n\", \"iam:ListSAMLProviders\", \"kms:ListAliases\", \"s3:GetBucketLocation\",
\n\"s3:ListAllMyBuckets\"], \"Resource\": \"*\"]}]"
      }
    ],
    "properties": {
      "cloudTrailProperties": {
        "endTime": "2024-02-14T22:44:40+00:00",
        "startTime": "2024-02-13T00:30:00+00:00",
        "trailProperties": [
          {
            "allRegions": true,
            "cloudTrailArn": "arn:aws:cloudtrail:us-
west-2:111122223333:trail/my-trail",
            "regions": []
          }
        ]
      }
    ]
  }
}
```

```

    },
    "isComplete": false,
    "principalArn": "arn:aws:iam::111122223333:role/Admin"
  }
},
"jobDetails": {
  "completedOn": "2024-02-14T22:47:01+00:00",
  "jobId": "c557dc4a-0338-4489-95dd-739014860ff9",
  "startedOn": "2024-02-14T22:44:41+00:00",
  "status": "SUCCEEDED"
}
}

```

Para obtener más información, consulte la sección [Generación de políticas de IAM Access Analyzer](#) en la Guía del AWS IAM Usuario.

- Para API obtener más información, consulte [GetGeneratedPolicy](#) la Referencia de AWS CLI comandos.

## list-access-preview-findings

En el siguiente ejemplo de código, se muestra cómo usar `list-access-preview-findings`.

### AWS CLI

Para recuperar una lista de los resultados de la vista previa de acceso generados por la vista previa de acceso especificada

En el siguiente `list-access-preview-findings` ejemplo, se recupera una lista de los resultados de la vista previa de acceso generada por la vista previa de acceso especificada en su AWS cuenta.

```

aws accessanalyzer list-access-preview-findings \
  --access-preview-id 3c65eb13-6ef9-4629-8919-a32043619e6b \
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/ConsoleAnalyzer-account

```

Salida:

```

{
  "findings": [
    {

```



```
    "id": "e22fc158-1c87-4c32-9464-e7f405ce8d74",
    "principal": {
      "AWS": "111122223333"
    },
    "action": [
      "s3:PutObject",
      "s3:PutObjectAcl"
    ],
    "condition": {},
    "resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
    "isPublic": false,
    "resourceType": "AWS::S3::Bucket",
    "createdAt": "2024-02-17T00:18:46+00:00",
    "changeType": "NEW",
    "status": "ACTIVE",
    "resourceOwnerAccount": "111122223333",
    "sources": [
      {
        "type": "POLICY"
      }
    ]
  }
]
```

Para obtener más información, consulte Vista [previa del acceso con IAM Access Analyzer APIs](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte la Referencia [ListAccessPreviewFindings](#) de AWS CLI comandos.

## list-access-previews

En el siguiente ejemplo de código, se muestra cómo usar `list-access-previews`.

### AWS CLI

Para recuperar una lista de vistas previas de acceso al analizador especificado

En el siguiente `list-access-previews` ejemplo, se recupera una lista de vistas previas de acceso del analizador especificado en su cuenta. AWS

```
aws accessanalyzer list-access-previews \
```

```
--analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/ConsoleAnalyzer-account
```

Salida:

```
{
  "accessPreviews": [
    {
      "id": "3c65eb13-6ef9-4629-8919-a32043619e6b",
      "analyzerArn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/ConsoleAnalyzer-account",
      "createdAt": "2024-02-17T00:18:44+00:00",
      "status": "COMPLETED"
    }
  ]
}
```

Para obtener más información, consulte Vista [previa del acceso con IAM Access Analyzer APIs](#) en la Guía del usuario.AWS IAM

- Para API obtener más información, consulte la Referencia [ListAccessPreviews](#) de AWS CLI comandos.

## list-analyzed-resources

En el siguiente ejemplo de código, se muestra cómo usar `list-analyzed-resources`.

AWS CLI

Para ver una lista de los widgets disponibles

En el siguiente `list-analyzed-resources` ejemplo, se enumeran los widgets disponibles en su AWS cuenta.

```
aws accessanalyzer list-analyzed-resources \  
--analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/ConsoleAnalyzer-account \  
--resource-type AWS::IAM::Role
```

Salida:

```
{
```

```
"analyzedResources": [
  {
    "resourceArn": "arn:aws:sns:us-west-2:111122223333:Validation-Email",
    "resourceOwnerAccount": "111122223333",
    "resourceType": "AWS::SNS::Topic"
  },
  {
    "resourceArn": "arn:aws:sns:us-west-2:111122223333:admin-alerts",
    "resourceOwnerAccount": "111122223333",
    "resourceType": "AWS::SNS::Topic"
  },
  {
    "resourceArn": "arn:aws:sns:us-west-2:111122223333:config-topic",
    "resourceOwnerAccount": "111122223333",
    "resourceType": "AWS::SNS::Topic"
  },
  {
    "resourceArn": "arn:aws:sns:us-west-2:111122223333:inspector-topic",
    "resourceOwnerAccount": "111122223333",
    "resourceType": "AWS::SNS::Topic"
  }
]
```

Para obtener más información, consulte [Uso del analizador de acceso a AWS Identity and Access Management](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListAnalyzedResources](#) la Referencia de AWS CLI comandos.

## list-analyzers

En el siguiente ejemplo de código, se muestra cómo usar `list-analyzers`.

### AWS CLI

Para recuperar una lista de analizadores

En el siguiente `list-analyzers` ejemplo, se recupera una lista de los analizadores de su cuenta. AWS

```
aws accessanalyzer list-analyzers
```

**Salida:**

```
{
  "analyzers": [
    {
      "arn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
UnusedAccess-ConsoleAnalyzer-organization",
      "createdAt": "2024-02-15T00:46:40+00:00",
      "name": "UnusedAccess-ConsoleAnalyzer-organization",
      "status": "ACTIVE",
      "tags": {
        "auto-delete": "no"
      },
      "type": "ORGANIZATION_UNUSED_ACCESS"
    },
    {
      "arn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-organization",
      "createdAt": "2020-04-25T07:43:28+00:00",
      "lastResourceAnalyzed": "arn:aws:s3::DOC-EXAMPLE-BUCKET",
      "lastResourceAnalyzedAt": "2024-02-15T21:51:56.517000+00:00",
      "name": "ConsoleAnalyzer-organization",
      "status": "ACTIVE",
      "tags": {
        "auto-delete": "no"
      },
      "type": "ORGANIZATION"
    },
    {
      "arn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account",
      "createdAt": "2019-12-03T07:28:17+00:00",
      "lastResourceAnalyzed": "arn:aws:sns:us-west-2:111122223333:config-
topic",
      "lastResourceAnalyzedAt": "2024-02-15T18:01:53.003000+00:00",
      "name": "ConsoleAnalyzer-account",
      "status": "ACTIVE",
      "tags": {
        "auto-delete": "no"
      },
      "type": "ACCOUNT"
    }
  ]
}
```

```
}
```

Para obtener más información, consulte [Uso del analizador de acceso a AWS Identity and Access Management](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListAnalyzers](#) la Referencia de AWS CLI comandos.

## list-archive-rules

En el siguiente ejemplo de código, se muestra cómo usar `list-archive-rules`.

### AWS CLI

Para recuperar una lista de reglas de archivo creadas para el analizador especificado

En el siguiente `list-archive-rules` ejemplo, se recupera una lista de reglas de archivo creadas para el analizador especificado en su cuenta. AWS

```
aws accessanalyzer list-archive-rules \  
  --analyzer-name UnusedAccess-ConsoleAnalyzer-organization
```

Salida:

```
{  
  "archiveRules": [  
    {  
      "createdAt": "2024-02-15T00:49:27+00:00",  
      "filter": {  
        "resource": {  
          "contains": [  
            "Cognito"  
          ]  
        },  
        "resourceType": {  
          "eq": [  
            "AWS::IAM::Role"  
          ]  
        }  
      },  
      "ruleName": "MyArchiveRule",
```

```

    "updatedAt": "2024-02-15T00:49:27+00:00"
  },
  {
    "createdAt": "2024-02-15T23:27:45+00:00",
    "filter": {
      "findingType": {
        "eq": [
          "UnusedIAMUserAccessKey"
        ]
      }
    },
    "ruleName": "ArchiveRule-56125a39-e517-4ff8-afb1-ef06f58db612",
    "updatedAt": "2024-02-15T23:27:45+00:00"
  }
]
}

```

Para obtener más información, consulte [Uso del analizador de acceso a AWS Identity and Access Management](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListArchiveRules](#) la Referencia de AWS CLI comandos.

## list-findings-v2

En el siguiente ejemplo de código, se muestra cómo usar `list-findings-v2`.

### AWS CLI

Para recuperar una lista de los hallazgos generados por el analizador especificado

En el siguiente `list-findings-v2` ejemplo, se recupera una lista de los resultados generados por el analizador especificado en su cuenta. AWS En este ejemplo, se filtran los resultados para incluir solo las IAM funciones cuyo nombre contenga. Cognito

```

aws accessanalyzer list-findings-v2 \
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  

ConsoleAnalyzer-account \  

  --filter '{"resource": {"contains": ["Cognito"]}, "resourceType": {"eq":  

  ["AWS::IAM::Role"]}}'

```

Salida:

```
{
  "findings": [
    {
      "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
      "createdAt": "2021-02-26T21:17:24.710000+00:00",
      "id": "597f3bc2-3adc-4c18-9879-5c4b23485e46",
      "resource": "arn:aws:iam::111122223333:role/
Cognito_testpoolUnauth_Role",
      "resourceType": "AWS::IAM::Role",
      "resourceOwnerAccount": "111122223333",
      "status": "ACTIVE",
      "updatedAt": "2021-02-26T21:17:24.710000+00:00",
      "findingType": "ExternalAccess"
    },
    {
      "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
      "createdAt": "2021-02-26T21:17:50.905000+00:00",
      "id": "ce0e221a-85b9-4d52-91ff-d7678075442f",
      "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role",
      "resourceType": "AWS::IAM::Role",
      "resourceOwnerAccount": "111122223333",
      "status": "ACTIVE",
      "updatedAt": "2021-02-26T21:17:50.905000+00:00",
      "findingType": "ExternalAccess"
    }
  ]
}
```

Para obtener más información, consulte [Uso del analizador de acceso a AWS Identity and Access Management](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte la sección [ListFindingsV2](#) en la Referencia de AWS CLI comandos.

## list-findings

En el siguiente ejemplo de código, se muestra cómo usar `list-findings`.

### AWS CLI

Para recuperar una lista de los hallazgos generados por el analizador especificado

En el siguiente `list-findings` ejemplo, se recupera una lista de los resultados generados por el analizador especificado en su cuenta. AWS En este ejemplo, se filtran los resultados para incluir solo las IAM funciones cuyo nombre contenga. Cognito

```
aws accessanalyzer list-findings \  
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-account \  
  --filter '{"resource": {"contains": ["Cognito"]}, "resourceType": {"eq":  
["AWS::IAM::Role"]}}'
```

Salida:

```
{  
  "findings": [  
    {  
      "id": "597f3bc2-3adc-4c18-9879-5c4b23485e46",  
      "principal": {  
        "Federated": "cognito-identity.amazonaws.com"  
      },  
      "action": [  
        "sts:AssumeRoleWithWebIdentity"  
      ],  
      "resource": "arn:aws:iam::111122223333:role/  
Cognito_testpoolUnauth_Role",  
      "isPublic": false,  
      "resourceType": "AWS::IAM::Role",  
      "condition": {  
        "cognito-identity.amazonaws.com:aud": "us-  
west-2:EXAMPLE0-0000-0000-0000-000000000000"  
      },  
      "createdAt": "2021-02-26T21:17:24.710000+00:00",  
      "analyzedAt": "2024-02-16T18:17:47.888000+00:00",  
      "updatedAt": "2021-02-26T21:17:24.710000+00:00",  
      "status": "ACTIVE",  
      "resourceOwnerAccount": "111122223333"  
    },  
    {  
      "id": "ce0e221a-85b9-4d52-91ff-d7678075442f",  
      "principal": {  
        "Federated": "cognito-identity.amazonaws.com"  
      },  
      "action": [  
        "sts:AssumeRoleWithWebIdentity"  
      ]  
    }  
  ]  
}
```



```

    ],
    "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role",
    "isPublic": false,
    "resourceType": "AWS::IAM::Role",
    "condition": {
      "cognito-identity.amazonaws.com:aud": "us-
west-2:EXAMPLE0-0000-0000-0000-000000000000"
    },
    "createdAt": "2021-02-26T21:17:50.905000+00:00",
    "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
    "updatedAt": "2021-02-26T21:17:50.905000+00:00",
    "status": "ACTIVE",
    "resourceOwnerAccount": "111122223333"
  }
]
}

```

Para obtener más información, consulte [Uso del analizador de acceso a AWS Identity and Access Management](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListFindings](#) la Referencia de AWS CLI comandos.

## list-policy-generations

En el siguiente ejemplo de código, se muestra cómo usar `list-policy-generations`.

### AWS CLI

Para enumerar todas las generaciones de políticas solicitadas en los últimos siete días

En el siguiente `list-policy-generations` ejemplo, se enumeran todas las generaciones de políticas solicitadas en los últimos siete días en su AWS cuenta.

```
aws accessanalyzer list-policy-generations
```

Salida:

```

{
  "policyGenerations": [
    {
      "completedOn": "2024-02-14T23:43:38+00:00",
      "jobId": "923a56b0-ebb8-4e80-8a3c-a11ccfbcd6f2",
      "principalArn": "arn:aws:iam::111122223333:role/Admin",

```

```

        "startedOn": "2024-02-14T23:43:02+00:00",
        "status": "CANCELED"
    },
    {
        "completedOn": "2024-02-14T22:47:01+00:00",
        "jobId": "c557dc4a-0338-4489-95dd-739014860ff9",
        "principalArn": "arn:aws:iam::111122223333:role/Admin",
        "startedOn": "2024-02-14T22:44:41+00:00",
        "status": "SUCCEEDED"
    }
]
}

```

Para obtener más información, consulte la sección [Generación de políticas de IAM Access Analyzer](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListPolicyGenerations](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para recuperar una lista de etiquetas aplicadas al recurso especificado

En el siguiente `list-tags-for-resource` ejemplo, se recupera una lista de etiquetas aplicadas al recurso especificado de su AWS cuenta.

```

aws accessanalyzer list-tags-for-resource \
  --resource-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  

ConsoleAnalyzer-account

```

Salida:

```

{
  "tags": {
    "Zone-of-trust": "Account",
    "Name": "ConsoleAnalyzer"
  }
}

```

Para obtener más información, consulte la sección [Generación de políticas de IAM Access Analyzer](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## start-policy-generation

En el siguiente ejemplo de código, se muestra cómo usar `start-policy-generation`.

### AWS CLI

Para iniciar una solicitud de generación de políticas

En el siguiente `start-policy-generation` ejemplo, se inicia una solicitud de generación de políticas en su AWS cuenta.

```
aws accessanalyzer start-policy-generation \
  --policy-generation-details '{"principalArn": "arn:aws:iam::111122223333:role/
Admin"}' \
  --cloud-trail-details file://myfile.json
```

Contenidos de `myfile.json`:

```
{
  "accessRole": "arn:aws:iam::111122223333:role/service-role/
AccessAnalyzerMonitorServiceRole",
  "startTime": "2024-02-13T00:30:00Z",
  "trails": [
    {
      "allRegions": true,
      "cloudTrailArn": "arn:aws:cloudtrail:us-west-2:111122223333:trail/my-
trail"
    }
  ]
}
```

Salida:

```
{
  "jobId": "c557dc4a-0338-4489-95dd-739014860ff9"
```

```
}
```

Para obtener más información, consulte la sección [Generación de políticas de IAM Access Analyzer](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [StartPolicyGeneration](#) la Referencia de AWS CLI comandos.

## **start-resource-scan**

En el siguiente ejemplo de código, se muestra cómo usar `start-resource-scan`.

### AWS CLI

Para iniciar inmediatamente un análisis de las políticas aplicadas al recurso especificado

En el siguiente `start-resource-scan` ejemplo, se inicia inmediatamente un análisis de las políticas aplicadas al recurso especificado en su AWS cuenta.

```
aws accessanalyzer start-resource-scan \  
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-account \  
  --resource-arn arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role
```

Este comando no genera ninguna salida.

Para obtener más información, consulte la sección [Generación de políticas de IAM Access Analyzer](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [StartResourceScan](#) la Referencia de AWS CLI comandos.

## **tag-resource**

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir una etiqueta al recurso especificado

En el siguiente `tag-resource` ejemplo, se agrega una etiqueta al recurso especificado de la AWS cuenta.

```
aws accessanalyzer tag-resource \  
  --resource-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-account \  
  --tags Environment=dev, Purpose=testing
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso del analizador de acceso a AWS Identity and Access Management](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar untag-resource.

### AWS CLI

Para eliminar etiquetas de los recursos especificados

En el siguiente untag-resource ejemplo, se eliminan las etiquetas del recurso especificado de la AWS cuenta.

```
aws accessanalyzer untag-resource \  
  --resource-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
ConsoleAnalyzer-account \  
  --tag-keys Environment Purpose
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso del analizador de acceso a AWS Identity and Access Management](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-archive-rule

En el siguiente ejemplo de código, se muestra cómo usar update-archive-rule.

## AWS CLI

Para actualizar los criterios y valores de la regla de archivado especificada

En el siguiente `update-archive-rule` ejemplo, se actualizan los criterios y valores de la regla de archivo especificada en su AWS cuenta.

```
aws accessanalyzer update-archive-rule \  
  --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \  
  --rule-name MyArchiveRule \  
  --filter '{"resource": {"contains": ["Cognito"]}, "resourceType": {"eq":  
  ["AWS::IAM::Role"]}}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [las reglas de archivado](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateArchiveRule](#) la Referencia de AWS CLI comandos.

## update-findings

En el siguiente ejemplo de código, se muestra cómo usar `update-findings`.

## AWS CLI

Para actualizar el estado de los hallazgos especificados

En el siguiente `update-findings` ejemplo, se actualiza el estado de los hallazgos especificados en su AWS cuenta.

```
aws accessanalyzer update-findings \  
  --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/  
  UnusedAccess-ConsoleAnalyzer-organization \  
  --ids 4f319ac3-2e0c-4dc4-bf51-7013a086b6ae 780d586a-2cce-4f72-aff6-359d450e7500  
 \  
  --status ARCHIVED
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso del analizador de acceso a AWS Identity and Access Management](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [UpdateFindings](#) la Referencia de AWS CLI comandos.

## validate-policy

En el siguiente ejemplo de código, se muestra cómo usar `validate-policy`.

### AWS CLI

Para solicitar la validación de una política y devuelve una lista de los resultados

En el siguiente `validate-policy` ejemplo, se solicita la validación de una política y se devuelve una lista de los resultados. La política del ejemplo es una política de confianza de roles para un rol de Amazon Cognito utilizado para la federación de identidades web. Los resultados generados a partir de la política de confianza se refieren a un valor de `Sid` elemento vacío y a un principio de política no coincidente debido a que se utilizó una acción incorrecta de asumir el rol. `sts:AssumeRole` La acción correcta de asumir un rol para usarla con Cognito es. `sts:AssumeRoleWithWebIdentity`

```
aws accessanalyzer validate-policy \  
  --policy-document file://myfile.json \  
  --policy-type RESOURCE_POLICY
```

Contenidos de `myfile.json`:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "",  
      "Effect": "Allow",  
      "Principal": {  
        "Federated": "cognito-identity.amazonaws.com"  
      },  
      "Action": [  
        "sts:AssumeRole",  
        "sts:TagSession"  
      ],  
      "Condition": {  
        "StringEquals": {  
          "cognito-identity.amazonaws.com:aud": "us-west-2_EXAMPLE"  
        }  
      }  
    }  
  ]  
}
```

```

    }
  }
]
}

```

**Salida:**

```

{
  "findings": [
    {
      "findingDetails": "Add a value to the empty string in the Sid element.",
      "findingType": "SUGGESTION",
      "issueCode": "EMPTY_SID_VALUE",
      "learnMoreLink": "https://docs.aws.amazon.com/IAM/latest/UserGuide/
access-analyzer-reference-policy-checks.html#access-analyzer-reference-policy-
checks-suggestion-empty-sid-value",
      "locations": [
        {
          "path": [
            {
              "value": "Statement"
            },
            {
              "index": 0
            },
            {
              "value": "Sid"
            }
          ],
          "span": {
            "end": {
              "column": 21,
              "line": 5,
              "offset": 81
            },
            "start": {
              "column": 19,
              "line": 5,
              "offset": 79
            }
          }
        }
      ]
    }
  ]
}

```



```
    ]
  },
  {
    "findingDetails": "The sts:AssumeRole action is invalid with the
following principal(s): cognito-identity.amazonaws.com. Use a SAML provider
principal with the sts:AssumeRoleWithSAML action or use an OIDC provider principal
with the sts:AssumeRoleWithWebIdentity action. Ensure the provider is Federated if
you use either of the two options.",
    "findingType": "ERROR",
    "issueCode": "MISMATCHED_ACTION_FOR_PRINCIPAL",
    "learnMoreLink": "https://docs.aws.amazon.com/IAM/latest/UserGuide/
access-analyzer-reference-policy-checks.html#access-analyzer-reference-policy-
checks-error-mismatched-action-for-principal",
    "locations": [
      {
        "path": [
          {
            "value": "Statement"
          },
          {
            "index": 0
          },
          {
            "value": "Action"
          },
          {
            "index": 0
          }
        ],
        "span": {
          "end": {
            "column": 32,
            "line": 11,
            "offset": 274
          },
          "start": {
            "column": 16,
            "line": 11,
            "offset": 258
          }
        }
      },
      {
        "path": [
```

```

        {
            "value": "Statement"
        },
        {
            "index": 0
        },
        {
            "value": "Principal"
        },
        {
            "value": "Federated"
        }
    ],
    "span": {
        "end": {
            "column": 61,
            "line": 8,
            "offset": 202
        },
        "start": {
            "column": 29,
            "line": 8,
            "offset": 170
        }
    }
}
]
},
{
    "findingDetails": "The following actions: sts:TagSession are not
supported by the condition key cognito-identity.amazonaws.com:aud. The condition
will not be evaluated for these actions. We recommend that you move these actions
to a different statement without this condition key.",
    "findingType": "ERROR",
    "issueCode": "UNSUPPORTED_ACTION_FOR_CONDITION_KEY",
    "learnMoreLink": "https://docs.aws.amazon.com/IAM/latest/UserGuide/
access-analyzer-reference-policy-checks.html#access-analyzer-reference-policy-
checks-error-unsupported-action-for-condition-key",
    "locations": [
        {
            "path": [
                {
                    "value": "Statement"
                }
            ],

```

```
    {
      "index": 0
    },
    {
      "value": "Action"
    },
    {
      "index": 1
    }
  ],
  "span": {
    "end": {
      "column": 32,
      "line": 12,
      "offset": 308
    },
    "start": {
      "column": 16,
      "line": 12,
      "offset": 292
    }
  }
},
{
  "path": [
    {
      "value": "Statement"
    },
    {
      "index": 0
    },
    {
      "value": "Condition"
    },
    {
      "value": "StringEquals"
    },
    {
      "value": "cognito-identity.amazonaws.com:aud"
    }
  ],
  "span": {
    "end": {
      "column": 79,
```

```
    "line": 16,  
    "offset": 464  
  },  
  "start": {  
    "column": 58,  
    "line": 16,  
    "offset": 443  
  }  
}  
]  
]  
]  
]  
]
```

Para obtener más información, consulte [Comprobaciones de validación de políticas](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [ValidatePolicy](#) la Referencia de AWS CLI comandos.

## Ejemplos de Image Builder utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Image Builder.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **create-component**

En el siguiente ejemplo de código, se muestra cómo usar `create-component`.

## AWS CLI

Para crear un componente

En el siguiente `create-component` ejemplo, se crea un componente que utiliza un archivo de JSON documento y hace referencia a un documento del componente en un YAML formato que se carga en un bucket de Amazon S3.

```
aws imagebuilder create-component \  
  --cli-input-json file://create-component.json
```

Contenidos de `create-component.json`:

```
{  
  "name": "MyExampleComponent",  
  "semanticVersion": "2019.12.02",  
  "description": "An example component that builds, validates and tests an image",  
  "changeDescription": "Initial version.",  
  "platform": "Windows",  
  "uri": "s3://s3-bucket-name/s3-bucket-path/component.yaml"  
}
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
  "componentBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/examplecomponent/2019.12.02/1"  
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [CreateComponent](#) la Referencia de AWS CLI comandos.

## **create-distribution-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `create-distribution-configuration`.

## AWS CLI

Para crear una configuración de distribución

El siguiente `create-distribution-configuration` ejemplo crea una configuración de distribución mediante un JSON archivo.

```
aws imagebuilder create-distribution-configuration \  
--cli-input-json file:/create-distribution-configuration.json
```

Contenidos de `create-distribution-configuration.json`:

```
{  
  "name": "MyExampleDistribution",  
  "description": "Copies AMI to eu-west-1",  
  "distributions": [  
    {  
      "region": "us-west-2",  
      "amiDistributionConfiguration": {  
        "name": "Name {{imagebuilder:buildDate}}",  
        "description": "An example image name with parameter references",  
        "amiTags": {  
          "KeyName": "{{ssm:parameter_name}}"  
        },  
        "launchPermission": {  
          "userIds": [  
            "123456789012"  
          ]  
        }  
      },  
    },  
    {  
      "region": "eu-west-1",  
      "amiDistributionConfiguration": {  
        "name": "My {{imagebuilder:buildVersion}} image  
{{imagebuilder:buildDate}}",  
        "amiTags": {  
          "KeyName": "Value"  
        },  
        "launchPermission": {  
          "userIds": [  
            "123456789012"  
          ]  
        }  
      }  
    }  
  ]  
}
```

```

    }
  }
]
}

```

Salida:

```

{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexampledistribution"
}

```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [CreateDistributionConfiguration](#) la Referencia de AWS CLI comandos.

## create-image-pipeline

En el siguiente ejemplo de código, se muestra cómo usar create-image-pipeline.

### AWS CLI

Para crear una canalización de imágenes

En el siguiente create-image-pipeline ejemplo, se crea una canalización de imágenes mediante un JSON archivo.

```

aws imagebuilder create-image-pipeline \
  --cli-input-json file://create-image-pipeline.json

```

Contenidos de create-image-pipeline.json:

```

{
  "name": "MyWindows2016Pipeline",
  "description": "Builds Windows 2016 Images",
  "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
mybasicrecipe/2019.12.03",

```

```

    "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",
    "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexampledistribution",
    "imageTestsConfiguration": {
        "imageTestsEnabled": true,
        "timeoutMinutes": 60
    },
    "schedule": {
        "scheduleExpression": "cron(0 0 * * SUN)",
        "pipelineExecutionStartCondition":
"EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
    },
    "status": "ENABLED"
}

```

Salida:

```

{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "imagePipelineArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline"
}

```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [CreateImagePipeline](#) la Referencia de AWS CLI comandos.

## create-image-recipe

En el siguiente ejemplo de código, se muestra cómo usar `create-image-recipe`.

### AWS CLI

Para crear una receta

En el siguiente `create-image-recipe` ejemplo, se crea una receta de imagen mediante un JSON archivo. Los componentes se instalan en el orden en que se especificaron.

```
aws imagebuilder create-image-recipe \
```



```
--cli-input-json file://create-image-recipe.json
```

Contenidos de create-image-recipe.json:

```
{
  "name": "MyBasicRecipe",
  "description": "This example image recipe creates a Windows 2016 image.",
  "semanticVersion": "2019.12.03",
  "components":
  [
    {
      "componentArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/myexamplecomponent/2019.12.02/1"
    },
    {
      "componentArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/myimportedcomponent/1.0.0/1"
    }
  ],
  "parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/windows-server-2016-english-full-base-x86/xxxx.x.x"
}
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/mybasicrecipe/2019.12.03"
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [CreateImageRecipe](#) la Referencia de AWS CLI comandos.

## create-image

En el siguiente ejemplo de código, se muestra cómo usar create-image.

## AWS CLI

Para crear una imagen

En el `create-image` ejemplo siguiente se crea una imagen.

```
aws imagebuilder create-image \  
  --image-recipe-arn arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/mybasicrecipe/2019.12.03 \  
  --infrastructure-configuration-arn arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/myexampleinfrastructure
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
  "imageBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/mybasicrecipe/2019.12.03/1"  
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [CreateImage](#) la Referencia de AWS CLI comandos.

## **create-infrastructure-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `create-infrastructure-configuration`.

## AWS CLI

Para crear una configuración de infraestructura

El siguiente `create-infrastructure-configuration` ejemplo crea una configuración de infraestructura mediante un JSON archivo.

```
aws imagebuilder create-infrastructure-configuration \  
  --cli-input-json file://create-infrastructure-configuration.json
```

## Contenidos de create-infrastructure-configuration.json:

```
{
  "name": "MyExampleInfrastructure",
  "description": "An example that will retain instances of failed builds",
  "instanceTypes": [
    "m5.large", "m5.xlarge"
  ],
  "instanceProfileName": "EC2InstanceProfileForImageBuilder",
  "securityGroupIds": [
    "sg-a1b2c3d4"
  ],
  "subnetId": "subnet-a1b2c3d4",
  "logging": {
    "s3Logs": {
      "s3BucketName": "bucket-name",
      "s3KeyPrefix": "bucket-path"
    }
  },
  "keyPair": "key-pair-name",
  "terminateInstanceOnFailure": false,
  "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:sns-topic-name"
}
```

## Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/myexampleinfrastructure"
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [CreateInfrastructureConfiguration](#) la Referencia de AWS CLI comandos.

## delete-component

En el siguiente ejemplo de código, se muestra cómo usar delete-component.

## AWS CLI

Para eliminar un componente

En el siguiente `delete-component` ejemplo, se elimina la versión de compilación de un componente especificando su ARN versión.

```
aws imagebuilder delete-component \  
  --component-build-version-arn arn:aws:imagebuilder:us-  
west-2:123456789012:component/myexamplecomponent/2019.12.02/1
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "componentBuildVersionArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:component/myexamplecomponent/2019.12.02/1"  
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [DeleteComponent](#) la Referencia de AWS CLI comandos.

## `delete-image-pipeline`

En el siguiente ejemplo de código, se muestra cómo usar `delete-image-pipeline`.

## AWS CLI

Para eliminar una canalización de imágenes

En el siguiente `delete-image-pipeline` ejemplo, se elimina una canalización de imágenes especificando su ARN.

```
aws imagebuilder delete-image-pipeline \  
  --image-pipeline-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/  
my-example-pipeline
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "imagePipelineArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline"
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [DeleteImagePipeline](#) la Referencia de AWS CLI comandos.

## delete-image-recipe

En el siguiente ejemplo de código, se muestra cómo usar `delete-image-recipe`.

### AWS CLI

Para eliminar una receta de imagen

En el siguiente `delete-image-recipe` ejemplo, se elimina una receta de imagen especificando su ARN receta.

```
aws imagebuilder delete-image-recipe \
  --image-recipe-arn arn:aws:imagebuilder:us-east-1:123456789012:image-recipe/
mybasicrecipe/2019.12.03
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
mybasicrecipe/2019.12.03"
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [DeleteImageRecipe](#) la Referencia de AWS CLI comandos.

## delete-image

En el siguiente ejemplo de código, se muestra cómo usar `delete-image`.

### AWS CLI

Para eliminar una imagen

En el siguiente `delete-image` ejemplo, se elimina una versión de compilación de imágenes especificando su ARN versión.

```
aws imagebuilder delete-image \  
  --image-build-version-arn arn:aws:imagebuilder:us-west-2:123456789012:image/my-  
example-image/2019.12.02/1
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "imageBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/  
mybasicrecipe/2019.12.03/1"  
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [DeleteImage](#) la Referencia de AWS CLI comandos.

## delete-infrastructure-configuration

En el siguiente ejemplo de código, se muestra cómo usar `delete-infrastructure-configuration`.

### AWS CLI

Para eliminar una configuración de infraestructura

En el siguiente `delete-infrastructure-configuration` ejemplo, se elimina una canalización de imágenes especificando su ARN.

```
aws imagebuilder delete-infrastructure-configuration \  
  --image-build-version-arn arn:aws:imagebuilder:us-west-2:123456789012:image/my-  
example-image/2019.12.02/1
```

```
--infrastructure-configuration-arn arn:aws:imagebuilder:us-east-1:123456789012:infrastructure-configuration/myexampleinfrastructure
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/myexampleinfrastructure"
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [DeleteInfrastructureConfiguration](#) la Referencia de AWS CLI comandos.

## get-component-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-component-policy`.

AWS CLI

Para obtener detalles de la política de componentes

En el siguiente `get-component-policy` ejemplo, se enumeran los detalles de una política de componentes especificando suARN.

```
aws imagebuilder get-component-policy \
  --component-arn arn:aws:imagebuilder:us-west-2:123456789012:component/my-example-component/2019.12.03/1
```

Salida:

```
{
  "Policy": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\": \"Allow\", \"Principal\": { \"AWS\": [ \"123456789012\" ] }, \"Action\": [ \"imagebuilder:GetComponent\", \"imagebuilder:ListComponents\" ], \"Resource\": [ \"arn:aws:imagebuilder:us-west-2:123456789012:component/my-example-component/2019.12.03/1\" ] } ] }"
```

Para obtener más información, consulte Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI < <https://docs.aws.amazon.com/imagebuilder/latest/userguide/managing-image-builder-cli.html> >`\_\_ en la Guía del usuario de Image Builder. EC2

- Para obtener API más información, [GetComponentPolicy](#) consulte AWS CLI la Referencia de comandos.

## get-component

En el siguiente ejemplo de código, se muestra cómo usar `get-component`.

### AWS CLI

Para obtener los detalles de los componentes

En el siguiente `get-component` ejemplo se enumeran los detalles de un componente especificando suARN.

```
aws imagebuilder get-component \  
  --component-build-version-arn arn:aws:imagebuilder:us-  
west-2:123456789012:component/component-name/1.0.0/1
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "component": {  
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/component-  
name/1.0.0/1",  
    "name": "component-name",  
    "version": "1.0.0",  
    "type": "TEST",  
    "platform": "Linux",  
    "owner": "123456789012",  
    "data": "name: HelloWorldTestingDocument\ndescription: This is hello world  
testing document.\nschemaVersion: 1.0\n\nphases:\n - name: test\n   steps:\n - name: HelloWorldStep\n   action: ExecuteBash\n   inputs:\n commands:\n - echo \"Hello World! Test.\"\n",  
    "encrypted": true,  
    "dateCreated": "2020-01-27T20:43:30.306Z",  
    "tags": {}  
  }  
}
```



```
}
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [GetComponent](#) la Referencia de AWS CLI comandos.

## get-distribution-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-distribution-configuration`.

### AWS CLI

Para obtener los detalles de una configuración de distribución

El siguiente `get-distribution-configuration` ejemplo muestra los detalles de una configuración de distribución especificando su configuraciónARN.

```
aws imagebuilder get-distribution-configuration \
  --distribution-configuration-arn arn:aws:imagebuilder:us-
  west-2:123456789012:distribution-configuration/myexampledistribution
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "distributionConfiguration": {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:distribution-
  configuration/myexampledistribution",
    "name": "MyExampleDistribution",
    "description": "Copies AMI to eu-west-1 and exports to S3",
    "distributions": [
      {
        "region": "us-west-2",
        "amiDistributionConfiguration": {
          "name": "Name {{imagebuilder:buildDate}}",
          "description": "An example image name with parameter
  references",
          "amiTags": {
```

```

        "KeyName": "{{ssm:parameter_name}}"
    },
    "launchPermission": {
        "userIds": [
            "123456789012"
        ]
    }
},
{
    "region": "eu-west-1",
    "amiDistributionConfiguration": {
        "name": "My {{imagebuilder:buildVersion}} image
{{imagebuilder:buildDate}}",
        "amiTags": {
            "KeyName": "Value"
        },
        "launchPermission": {
            "userIds": [
                "123456789012"
            ]
        }
    }
},
{
    "dateCreated": "2020-02-19T18:40:10.529Z",
    "tags": {}
}
}

```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [GetDistributionConfiguration](#) la Referencia de AWS CLI comandos.

## get-image-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `get-image-pipeline`.

### AWS CLI

Para obtener detalles sobre la canalización de imágenes

En el siguiente `get-image-pipeline` ejemplo, se enumeran los detalles de una canalización de imágenes especificando suARN.

```
aws imagebuilder get-image-pipeline \  
  --image-pipeline-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/  
mywindows2016pipeline
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "imagePipeline": {  
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/  
mywindows2016pipeline",  
    "name": "MyWindows2016Pipeline",  
    "description": "Builds Windows 2016 Images",  
    "platform": "Windows",  
    "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/  
mybasicrecipe/2019.12.03",  
    "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",  
    "distributionConfigurationArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:distribution-configuration/myexampledistribution",  
    "imageTestsConfiguration": {  
      "imageTestsEnabled": true,  
      "timeoutMinutes": 60  
    },  
    "schedule": {  
      "scheduleExpression": "cron(0 0 * * SUN)",  
      "pipelineExecutionStartCondition":  
"EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"  
    },  
    "status": "ENABLED",  
    "dateCreated": "2020-02-19T19:04:01.253Z",  
    "dateUpdated": "2020-02-19T19:04:01.253Z",  
    "tags": {}  
  }  
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [GetImagePipeline](#) la Referencia de AWS CLI comandos.

## get-image-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-image-policy`.

### AWS CLI

Para obtener detalles de la política de imágenes

En el siguiente `get-image-policy` ejemplo, se enumeran los detalles de una política de imágenes especificando suARN.

```
aws imagebuilder get-image-policy \  
  --image-arn arn:aws:imagebuilder:us-west-2:123456789012:image/my-example-  
image/2019.12.03/1
```

Salida:

```
{  
  "Policy": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\": \"Allow\",  
  \"Principal\": { \"AWS\": [ \"123456789012\" ] }, \"Action\": [ \"imagebuilder:GetImage\",  
  \"imagebuilder:ListImages\" ], \"Resource\": [ \"arn:aws:imagebuilder:us-  
west-2:123456789012:image/my-example-image/2019.12.03/1\" ] } ] }\"  
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [GetImagePolicy](#) la Referencia de AWS CLI comandos.

## get-image-recipe-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-image-recipe-policy`.

### AWS CLI

Para obtener información detallada sobre la política de recetas con imágenes

En el siguiente `get-image-recipe-policy` ejemplo, se enumeran los detalles de una política de recetas de imágenes especificando su políticaARN.

```
aws imagebuilder get-image-recipe-policy \
  --image-recipe-arn arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-example-image-recipe/2019.12.03/1
```

Salida:

```
{
  "Policy": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\":
  \"Allow\", \"Principal\": { \"AWS\": [ \"123456789012\" ] }, \"Action\":
  [ \"imagebuilder:GetImageRecipe\", \"imagebuilder:ListImageRecipes\" ], \"Resource\":
  [ \"arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-example-image-
  recipe/2019.12.03/1\" ] } ] }"
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [GetImageRecipePolicy](#) la Referencia de AWS CLI comandos.

## get-image

En el siguiente ejemplo de código, se muestra cómo usar `get-image`.

### AWS CLI

Para obtener detalles de la imagen

En el siguiente `get-image` ejemplo se enumeran los detalles de una imagen especificando suARN.

```
aws imagebuilder get-image \
  --image-build-version-arn arn:aws:imagebuilder:us-west-2:123456789012:image/mybasicrecipe/2019.12.03/1
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
```

```
"image": {
  "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/1",
  "name": "MyBasicRecipe",
  "version": "2019.12.03/1",
  "platform": "Windows",
  "state": {
    "status": "BUILDING"
  },
  "imageRecipe": {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
mybasicrecipe/2019.12.03",
    "name": "MyBasicRecipe",
    "description": "This example image recipe creates a Windows 2016
image.",
    "platform": "Windows",
    "version": "2019.12.03",
    "components": [
      {
        "componentArn": "arn:aws:imagebuilder:us-
west-2:123456789012:component/myexamplecomponent/2019.12.02/1"
      },
      {
        "componentArn": "arn:aws:imagebuilder:us-
west-2:123456789012:component/myimportedcomponent/1.0.0/1"
      }
    ],
    "parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/windows-
server-2016-english-full-base-x86/2019.12.17/1",
    "dateCreated": "2020-02-14T19:46:16.904Z",
    "tags": {}
  },
  "infrastructureConfiguration": {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-
configuration/myexampleinfrastructure",
    "name": "MyExampleInfrastructure",
    "description": "An example that will retain instances of failed builds",
    "instanceTypes": [
      "m5.large",
      "m5.xlarge"
    ],
    "instanceProfileName": "EC2InstanceProfileForImageFactory",
    "securityGroupIds": [
      "sg-a1b2c3d4"
    ]
  }
}
```

```
    ],
    "subnetId": "subnet-a1b2c3d4",
    "logging": {
      "s3Logs": {
        "s3BucketName": "bucket-name",
        "s3KeyPrefix": "bucket-path"
      }
    },
    "keyPair": "Sam",
    "terminateInstanceOnFailure": false,
    "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:sns-name",
    "dateCreated": "2020-02-14T21:21:05.098Z",
    "tags": {}
  },
  "imageTestsConfiguration": {
    "imageTestsEnabled": true,
    "timeoutMinutes": 720
  },
  "dateCreated": "2020-02-14T23:14:13.597Z",
  "outputResources": {
    "amis": []
  },
  "tags": {}
}
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [GetImage](#) la Referencia de AWS CLI comandos.

## get-infrastructure-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-infrastructure-configuration`.

### AWS CLI

Para obtener detalles de la configuración de la infraestructura

En el siguiente `get-infrastructure-configuration` ejemplo, se enumeran los detalles de una configuración de infraestructura especificando su configuraciónARN.

```
aws imagebuilder get-infrastructure-configuration \  
  --infrastructure-configuration-arn arn:aws:imagebuilder:us-  
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "infrastructureConfiguration": {  
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-  
configuration/myexampleinfrastructure",  
    "name": "MyExampleInfrastructure",  
    "description": "An example that will retain instances of failed builds",  
    "instanceTypes": [  
      "m5.large",  
      "m5.xlarge"  
    ],  
    "instanceProfileName": "EC2InstanceProfileForImageBuilder",  
    "securityGroupIds": [  
      "sg-a48c95ef"  
    ],  
    "subnetId": "subnet-a48c95ef",  
    "logging": {  
      "s3Logs": {  
        "s3BucketName": "bucket-name",  
        "s3KeyPrefix": "bucket-path"  
      }  
    },  
    "keyPair": "Name",  
    "terminateInstanceOnFailure": false,  
    "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:sns-name",  
    "dateCreated": "2020-02-19T19:11:51.858Z",  
    "tags": {}  
  }  
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [GetInfrastructureConfiguration](#) la Referencia de AWS CLI comandos.



## import-component

En el siguiente ejemplo de código, se muestra cómo usar `import-component`.

### AWS CLI

Para importar un componente

El siguiente `import-component` ejemplo importa un script preexistente mediante un JSON archivo.

```
aws imagebuilder import-component \  
  --cli-input-json file://import-component.json
```

Contenidos de `import-component.json`:

```
{  
  "name": "MyImportedComponent",  
  "semanticVersion": "1.0.0",  
  "description": "An example of how to import a component",  
  "changeDescription": "First commit message.",  
  "format": "SHELL",  
  "platform": "Windows",  
  "type": "BUILD",  
  "uri": "s3://s3-bucket-name/s3-bucket-path/component.yaml"  
}
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
  "componentBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/myimportedcomponent/1.0.0/1"  
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [ImportComponent](#) la Referencia de AWS CLI comandos.

## list-component-build-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-component-build-versions`.

### AWS CLI

Para enumerar las versiones de compilación de los componentes

En el siguiente `list-component-build-versions` ejemplo, se enumeran las versiones de compilación de los componentes con una versión semántica específica.

```
aws imagebuilder list-component-build-versions --component-  
version-arn arn:aws:imagebuilder:us-west-2:123456789012:component/  
myexamplecomponent/2019.12.02
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "componentSummaryList": [  
    {  
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/  
myexamplecomponent/2019.12.02/1",  
      "name": "MyExampleComponent",  
      "version": "2019.12.02",  
      "platform": "Windows",  
      "type": "BUILD",  
      "owner": "123456789012",  
      "description": "An example component that builds, validates and tests an  
image",  
      "changeDescription": "Initial version.",  
      "dateCreated": "2020-02-19T18:53:45.940Z",  
      "tags": {  
        "KeyName": "KeyValue"  
      }  
    }  
  ]  
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [ListComponentBuildVersions](#) la Referencia de AWS CLI comandos.

## list-components

En el siguiente ejemplo de código, se muestra cómo usar `list-components`.

### AWS CLI

Para enumerar todas las versiones semánticas de los componentes

En el siguiente `list-components` ejemplo, se enumeran todas las versiones semánticas de los componentes a las que tiene acceso. Si lo desea, puede filtrar si desea incluir componentes de su propiedad, de Amazon o que otras cuentas hayan compartido con usted.

```
aws imagebuilder list-components
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "componentVersionList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/component-name/1.0.0",
      "name": "component-name",
      "version": "1.0.0",
      "platform": "Linux",
      "type": "TEST",
      "owner": "123456789012",
      "dateCreated": "2020-01-27T20:43:30.306Z"
    }
  ]
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [ListComponents](#) la Referencia de AWS CLI comandos.

## list-distribution-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-distribution-configurations`.

AWS CLI

Para enumerar las distribuciones

En el siguiente `list-distribution-configurations` ejemplo, se enumeran todas las distribuciones.

```
aws imagebuilder list-distribution-configurations
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "distributionConfigurationSummaryList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:distribution-configuration/myexampledistribution",
      "name": "MyExampleDistribution",
      "description": "Copies AMI to eu-west-1 and exports to S3",
      "dateCreated": "2020-02-19T18:40:10.529Z",
      "tags": {
        "KeyName": "KeyValue"
      }
    }
  ]
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [ListDistributionConfigurations](#) la Referencia de AWS CLI comandos.

## list-image-build-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-image-build-versions`.

## AWS CLI

Para ver una lista de las versiones de compilación de imágenes

En el siguiente `list-image-build-versions` ejemplo, se enumeran todas las versiones de creación de imágenes con una versión semántica.

```
aws imagebuilder list-image-build-versions \  
  --image-version-arn arn:aws:imagebuilder:us-west-2:123456789012:image/  
mybasicrecipe/2019.12.03
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "imageSummaryList": [  
    {  
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/  
mybasicrecipe/2019.12.03/7",  
      "name": "MyBasicRecipe",  
      "version": "2019.12.03/7",  
      "platform": "Windows",  
      "state": {  
        "status": "FAILED",  
        "reason": "Can't start SSM Automation for arn  
arn:aws:imagebuilder:us-west-2:123456789012:image/mybasicrecipe/2019.12.03/7 during  
building. Parameter \"iamInstanceProfileName\" has a null value."  
      },  
      "owner": "123456789012",  
      "dateCreated": "2020-02-19T18:56:11.511Z",  
      "outputResources": {  
        "amis": []  
      },  
      "tags": {}  
    },  
    {  
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/  
mybasicrecipe/2019.12.03/6",  
      "name": "MyBasicRecipe",  
      "version": "2019.12.03/6",  
      "platform": "Windows",  
      "state": {  
        "status": "FAILED",
```

```

        "reason": "An internal error has occurred."
    },
    "owner": "123456789012",
    "dateCreated": "2020-02-18T22:49:08.142Z",
    "outputResources": {
        "amis": [
            {
                "region": "us-west-2",
                "image": "ami-a1b2c3d4567890ab",
                "name": "MyBasicRecipe 2020-02-18T22-49-38.704Z",
                "description": "This example image recipe creates a Windows
2016 image."
            },
            {
                "region": "us-west-2",
                "image": "ami-a1b2c3d4567890ab",
                "name": "Name 2020-02-18T22-49-08.131Z",
                "description": "Copies AMI to eu-west-2 and exports to S3"
            },
            {
                "region": "eu-west-2",
                "image": "ami-a1b2c3d4567890ab",
                "name": "My 6 image 2020-02-18T22-49-08.131Z",
                "description": "Copies AMI to eu-west-2 and exports to S3"
            }
        ]
    },
    "tags": {}
},
{
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/5",
    "name": "MyBasicRecipe",
    "version": "2019.12.03/5",
    "platform": "Windows",
    "state": {
        "status": "AVAILABLE"
    },
    "owner": "123456789012",
    "dateCreated": "2020-02-18T16:51:48.403Z",
    "outputResources": {
        "amis": [
            {
                "region": "us-west-2",

```

```

        "image": "ami-a1b2c3d4567890ab",
        "name": "MyBasicRecipe 2020-02-18T16-52-18.965Z",
        "description": "This example image recipe creates a Windows
2016 image."
    }
  ]
},
"tags": {}
},
{
  "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/4",
  "name": "MyBasicRecipe",
  "version": "2019.12.03/4",
  "platform": "Windows",
  "state": {
    "status": "AVAILABLE"
  },
  "owner": "123456789012",
  "dateCreated": "2020-02-18T16:50:01.827Z",
  "outputResources": {
    "amis": [
      {
        "region": "us-west-2",
        "image": "ami-a1b2c3d4567890ab",
        "name": "MyBasicRecipe 2020-02-18T16-50-32.280Z",
        "description": "This example image recipe creates a Windows
2016 image."
      }
    ]
  },
  "tags": {}
},
{
  "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/3",
  "name": "MyBasicRecipe",
  "version": "2019.12.03/3",
  "platform": "Windows",
  "state": {
    "status": "AVAILABLE"
  },
  "owner": "123456789012",
  "dateCreated": "2020-02-14T23:14:13.597Z",

```

```

    "outputResources": {
      "amis": [
        {
          "region": "us-west-2",
          "image": "ami-a1b2c3d4567890ab",
          "name": "MyBasicRecipe 2020-02-14T23-14-44.243Z",
          "description": "This example image recipe creates a Windows
2016 image."
        }
      ]
    },
    "tags": {}
  },
  {
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/2",
    "name": "MyBasicRecipe",
    "version": "2019.12.03/2",
    "platform": "Windows",
    "state": {
      "status": "FAILED",
      "reason": "SSM execution 'a1b2c3d4-5678-90ab-cdef-EXAMPLE11111'
failed with status = 'Failed' and failure message = 'Step fails when it is
verifying the command has completed. Command a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
returns unexpected invocation result: \n{Status=[Failed], ResponseCode=[1],
Output=[\n-----ERROR-----\nfailed to run commands: exit status 1],
OutputPayload=[{\"Status\": \"Failed\", \"ResponseCode\": 1, \"Output\": \"\
\n-----ERROR-----\nfailed to run commands: exit status 1\", \"CommandId\":
\n\"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111\"}], CommandId=[a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111]}. Please refer to Automation Service Troubleshooting Guide for more
diagnosis details.'"
    },
    "owner": "123456789012",
    "dateCreated": "2020-02-14T22:57:42.593Z",
    "outputResources": {
      "amis": []
    },
    "tags": {}
  }
]
}

```



Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [ListImageBuildVersions](#) la Referencia de AWS CLI comandos.

## list-image-pipeline-images

En el siguiente ejemplo de código, se muestra cómo usar `list-image-pipeline-images`.

### AWS CLI

Para ver una lista de imágenes: imágenes de canalización

En el siguiente `list-image-pipeline-images` ejemplo, se enumeran todas las imágenes que se crearon mediante una canalización de imágenes específica.

```
aws imagebuilder list-image-pipeline-images \  
  --image-pipeline-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/  
mywindows2016pipeline
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "imagePipelineList": [  
    {  
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/  
mywindows2016pipeline",  
      "name": "MyWindows2016Pipeline",  
      "description": "Builds Windows 2016 Images",  
      "platform": "Windows",  
      "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-  
recipe/mybasicrecipe/2019.12.03",  
      "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",  
      "distributionConfigurationArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:distribution-configuration/myexampledistribution",  
      "imageTestsConfiguration": {  
        "imageTestsEnabled": true,  
        "timeoutMinutes": 60  
      },  
      "schedule": {
```

```

        "scheduleExpression": "cron(0 0 * * SUN)",
        "pipelineExecutionStartCondition":
"EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
    },
    "status": "ENABLED",
    "dateCreated": "2020-02-19T19:04:01.253Z",
    "dateUpdated": "2020-02-19T19:04:01.253Z",
    "tags": {
        "KeyName": "KeyValue"
    }
},
{
    "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/sam",
    "name": "PipelineName",
    "platform": "Linux",
    "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-
recipe/recipe-name-a1b2c3d45678/1.0.0",
    "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/infrastructureconfiguration-name-
a1b2c3d45678",
    "imageTestsConfiguration": {
        "imageTestsEnabled": true,
        "timeoutMinutes": 720
    },
    "status": "ENABLED",
    "dateCreated": "2019-12-16T18:19:02.068Z",
    "dateUpdated": "2019-12-16T18:19:02.068Z",
    "tags": {
        "KeyName": "KeyValue"
    }
}
]
}

```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [ListImagePipelineImages](#) la Referencia de AWS CLI comandos.

## list-image-recipes

En el siguiente ejemplo de código, se muestra cómo usar `list-image-recipes`.

## AWS CLI

Para ver una lista de recetas con imágenes

En el siguiente `list-image-recipes` ejemplo, se enumeran todas las recetas con imágenes.

```
aws imagebuilder list-image-recipes
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "imageRecipeSummaryList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/mybasicrecipe/2019.12.03",
      "name": "MyBasicRecipe",
      "platform": "Windows",
      "owner": "123456789012",
      "parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/windows-server-2016-english-full-base-x86/2019.x.x",
      "dateCreated": "2020-02-19T18:54:25.975Z",
      "tags": {
        "KeyName": "KeyValue"
      }
    },
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/recipe-name-a1b2c3d45678/1.0.0",
      "name": "recipe-name-a1b2c3d45678",
      "platform": "Linux",
      "owner": "123456789012",
      "parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/amazon-linux-2-x86/2019.11.21",
      "dateCreated": "2019-12-16T18:19:00.120Z",
      "tags": {
        "KeyName": "KeyValue"
      }
    }
  ]
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [ListImageRecipes](#) la Referencia de AWS CLI comandos.

## list-images

En el siguiente ejemplo de código, se muestra cómo usar `list-images`.

### AWS CLI

Para enumerar imágenes

En el siguiente `list-images` ejemplo, se enumeran todas las versiones semánticas a las que tiene acceso.

```
aws imagebuilder list-images
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "imageVersionList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/mybasicrecipe/2019.12.03",
      "name": "MyBasicRecipe",
      "version": "2019.12.03",
      "platform": "Windows",
      "owner": "123456789012",
      "dateCreated": "2020-02-14T21:29:18.810Z"
    }
  ]
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [ListImages](#) la Referencia de AWS CLI comandos.

## list-infrastructure-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-infrastructure-configurations`.

### AWS CLI

Para enumerar las configuraciones de infraestructura

En el siguiente `list-infrastructure-configurations` ejemplo, se enumeran todas las configuraciones de infraestructura.

```
aws imagebuilder list-infrastructure-configurations
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "infrastructureConfigurationSummaryList": [
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",
      "name": "MyExampleInfrastructure",
      "description": "An example that will retain instances of failed builds",
      "dateCreated": "2020-02-19T19:11:51.858Z",
      "tags": {}
    },
    {
      "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-configuration/infrastructureconfiguration-name-a1b2c3d45678",
      "name": "infrastructureConfiguration-name-a1b2c3d45678",
      "dateCreated": "2019-12-16T18:19:01.038Z",
      "tags": {
        "KeyName": "KeyValue"
      }
    }
  ]
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [ListInfrastructureConfigurations](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un recurso específico

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas de un recurso específico.

```
aws imagebuilder list-tags-for-resource \
  --resource-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline
```

Salida:

```
{
  "tags": {
    "KeyName": "KeyValue"
  }
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## put-component-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-component-policy`.

### AWS CLI

Para aplicar una política de recursos a un componente

El siguiente `put-component-policy` comando aplica una política de recursos a un componente de compilación para permitir el uso compartido de los componentes de compilación

entre cuentas. Le recomendamos que utilice el RAM CLI comando `create-resource-share`. Si usa el CLI comando EC2 Image Builder `put-component-policy`, también debe usarlo para que el RAM CLI recurso esté visible para todos los principales con los que se comparte el recurso.

`promote-resource-share-create-from-policy`

```
aws imagebuilder put-component-policy \
  --component-arn arn:aws:imagebuilder:us-west-2:123456789012:component/
  examplecomponent/2019.12.02/1 \
  --policy '{ "Version": "2012-10-17", "Statement": [ { "Effect":
  "Allow", "Principal": { "AWS": [ "123456789012" ] }, "Action":
  [ "imagebuilder:GetComponent", "imagebuilder:ListComponents" ],
  "Resource": [ "arn:aws:imagebuilder:us-west-2:123456789012:component/
  examplecomponent/2019.12.02/1" ] } ] }'
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "componentArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/
  examplecomponent/2019.12.02/1"
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [PutComponentPolicy](#) la Referencia de AWS CLI comandos.

## put-image-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-image-policy`.

### AWS CLI

Para aplicar una política de recursos a una imagen

El siguiente `put-image-policy` comando aplica una política de recursos a una imagen para permitir el intercambio de imágenes entre cuentas. Le recomendamos que utilice el RAM CLI comando `create-resource-share`. Si usa el CLI comando EC2 Image Builder `put-image-policy`, también debe usar el RAM CLI comando `promote-resource-share-create-from-policy` para que el recurso esté visible para todos los principales con los que se comparte el recurso.

```
aws imagebuilder put-image-policy \  
  --image-arn arn:aws:imagebuilder:us-west-2:123456789012:image/example-  
image/2019.12.02/1 \  
  --policy '{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow",  
"Principal": { "AWS": [ "123456789012" ] }, "Action": [ "imagebuilder:GetImage",  
"imagebuilder:ListImages" ], "Resource": [ "arn:aws:imagebuilder:us-  
west-2:123456789012:image/example-image/2019.12.02/1" ] } ] }'
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "imageArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/example-  
image/2019.12.02/1"  
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [PutImagePolicy](#) la Referencia de AWS CLI comandos.

## put-image-recipe-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-image-recipe-policy`.

AWS CLI

Para aplicar una política de recursos a una receta de imagen

El siguiente `put-image-recipe-policy` comando aplica una política de recursos a una receta de imagen para permitir el intercambio de recetas de imágenes entre cuentas. Le recomendamos que utilice el RAM CLI comando `create-resource-share`. Si usa el CLI comando EC2 Image Builder `put-image-recipe-policy`, también debe usarlo para que el RAM CLI recurso esté visible para todos los principales con los que se comparte el recurso. `promote-resource-share-create-from-policy`

```
aws imagebuilder put-image-recipe-policy \  
  --image-recipe-arn arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/  
example-image-recipe/2019.12.02 \  
  --policy '{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow",  
"Principal": { "AWS": [ "123456789012" ] }, "Action": [ "imagebuilder:PutImageRecipePolicy",  
"imagebuilder:ListImageRecipes" ], "Resource": [ "arn:aws:imagebuilder:us-  
west-2:123456789012:image-recipe/example-image-recipe/2019.12.02" ] } ] }'
```



```
--policy '{ "Version": "2012-10-17", "Statement": [ { "Effect":
"Allow", "Principal": { "AWS": [ "123456789012" ] }, "Action":
[ "imagebuilder:GetImageRecipe", "imagebuilder:ListImageRecipes" ], "Resource":
[ "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/example-image-
recipe/2019.12.02" ] } ] }'
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
example-image-recipe/2019.12.02/1"
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [PutImageRecipePolicy](#) la Referencia de AWS CLI comandos.

## start-image-pipeline-execution

En el siguiente ejemplo de código, se muestra cómo usar `start-image-pipeline-execution`.

AWS CLI

Para iniciar una canalización de imágenes manualmente

En el siguiente `start-image-pipeline-execution` ejemplo, se inicia manualmente una canalización de imágenes.

```
aws imagebuilder start-image-pipeline-execution \
  --image-pipeline-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
}
```

```
"imageBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/mybasicrecipe/2019.12.03/1"
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [StartImagePipelineExecution](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar un recurso

En el siguiente `tag-resource` ejemplo, se agrega y etiqueta un recurso a EC2 Image Builder mediante un JSON archivo.

```
aws imagebuilder tag-resource \
  --cli-input-json file://tag-resource.json
```

Contenidos de `tag-resource.json`:

```
{
  "resourceArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/mywindows2016pipeline",
  "tags": {
    "KeyName": "KeyValue"
  }
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar una etiqueta de un recurso

En el siguiente `untag-resource` ejemplo, se elimina una etiqueta de un recurso mediante un JSON archivo.

```
aws imagebuilder untag-resource \  
  --cli-input-json file://tag-resource.json
```

Contenidos de `untag-resource.json`:

```
{  
  "resourceArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/  
mywindows2016pipeline",  
  "tagKeys": [  
    "KeyName"  
  ]  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-distribution-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-distribution-configuration`.

### AWS CLI

Para actualizar una configuración de distribución

El siguiente `update-distribution-configuration` ejemplo actualiza una configuración de distribución mediante un JSON archivo.

```
aws imagebuilder update-distribution-configuration \  
  --cli-input-json file://update-distribution-configuration.json
```

Contenidos de `update-distribution-configuration.json`:

```
{  
  "distributionConfigurationArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:distribution-configuration/myexampledistribution",  
  "description": "Copies AMI to eu-west-2 and exports to S3",  
  "distributions": [  
    {  
      "region": "us-west-2",  
      "amiDistributionConfiguration": {  
        "name": "Name {{imagebuilder:buildDate}}",  
        "description": "An example image name with parameter references"  
      }  
    },  
    {  
      "region": "eu-west-2",  
      "amiDistributionConfiguration": {  
        "name": "My {{imagebuilder:buildVersion}} image  
{{imagebuilder:buildDate}}"  
      }  
    }  
  ]  
}
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [UpdateDistributionConfiguration](#) la Referencia de AWS CLI comandos.

## update-image-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `update-image-pipeline`.

### AWS CLI

Para actualizar una canalización de imágenes

En el siguiente `update-image-pipeline` ejemplo, se actualiza una canalización de imágenes mediante un JSON archivo.

```
aws imagebuilder update-image-pipeline \  
  --cli-input-json file://update-image-pipeline.json
```

Contenidos de `update-image-pipeline.json`:

```
{  
  "imagePipelineArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/  
mywindows2016pipeline",  
  "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/  
mybasicrecipe/2019.12.03",  
  "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",  
  "distributionConfigurationArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:distribution-configuration/myexampledistribution",  
  "imageTestsConfiguration": {  
    "imageTestsEnabled": true,  
    "timeoutMinutes": 120  
  },  
  "schedule": {  
    "scheduleExpression": "cron(0 0 * * MON)",  
    "pipelineExecutionStartCondition":  
"EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"  
  },  
  "status": "DISABLED"  
}
```

Salida:

```
{  
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [UpdateImagePipeline](#) la Referencia de AWS CLI comandos.

## update-infrastructure-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-infrastructure-configuration`.

### AWS CLI

Para actualizar una configuración de infraestructura

El siguiente `update-infrastructure-configuration` ejemplo actualiza una configuración de infraestructura mediante un JSON archivo.

```
aws imagebuilder update-infrastructure-configuration \  
--cli-input-json file:/update-infrastructure-configuration.json
```

Contenidos de `update-infrastructure-configuration.json`:

```
{  
  "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-  
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",  
  "description": "An example that will terminate instances of failed builds",  
  "instanceTypes": [  
    "m5.large", "m5.2xlarge"  
  ],  
  "instanceProfileName": "EC2InstanceProfileForImageFactory",  
  "securityGroupIds": [  
    "sg-a48c95ef"  
  ],  
  "subnetId": "subnet-a48c95ef",  
  "logging": {  
    "s3Logs": {  
      "s3BucketName": "bucket-name",  
      "s3KeyPrefix": "bucket-path"  
    }  
  },  
  "terminateInstanceOnFailure": true,
```

```
"snsTopicArn": "arn:aws:sns:us-west-2:123456789012:sns-name"
}
```

Salida:

```
{
  "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Para obtener más información, consulte [Configuración y administración de una canalización de imágenes de EC2 Image Builder mediante AWS CLI](#) la Guía del usuario de EC2 Image Builder.

- Para API obtener más información, consulte [UpdateInfrastructureConfiguration](#) la Referencia de AWS CLI comandos.

## Ejemplos de Incident Manager que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el AWS Command Line Interface uso de Incident Manager.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-replication-set**

En el siguiente ejemplo de código, se muestra cómo usar `create-replication-set`.

AWS CLI

Para crear el conjunto de réplicas

En el siguiente `create-replication-set` ejemplo, se crea el conjunto de réplicas que Incident Manager utiliza para replicar y cifrar los datos de su cuenta de Amazon Web Services. En este ejemplo, se utilizan las regiones `us-east-1` y `us-east-2` al crear el conjunto de replicación.

```
aws ssm-incidents create-replication-set \
  --regions '{"us-east-1": {"sseKmsKeyId": "arn:aws:kms:us-
east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"}, "us-east-2":
{"sseKmsKeyId": "arn:aws:kms:us-
east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"}}'
```

Salida:

```
{
  "replicationSetArns": [
    "arn:aws:ssm-incidents::111122223333:replication-set/c4bcb603-4bf9-
bb3f-413c-08df53673b57"
  ]
}
```

Para obtener más información, consulte [Uso del conjunto de replicación de Incident Manager en la Guía del usuario de Incident Manager](#).

- Para API obtener más información, consulte [CreateReplicationSet](#) la Referencia de AWS CLI comandos.

## create-response-plan

En el siguiente ejemplo de código, se muestra cómo usar `create-response-plan`.

AWS CLI

Para crear un plan de respuesta

En el siguiente `create-response-plan` ejemplo, se crea un plan de respuesta con los detalles especificados.

```
aws ssm-incidents create-response-plan \
  --chat-channel '{"chatbotSns": ["arn:aws:sns:us-
east-1:111122223333:Standard_User"]}' \
  --display-name "Example response plan" \
  --incident-template '{"impact": 5, "title": "example-incident"}' \
```



```
--name "example-response" \
--actions '[{"ssmAutomation": {"documentName": "AWSIncidents-
CriticalIncidentRunbookTemplate", "documentVersion": "$DEFAULT",
"roleArn": "arn:aws:iam::111122223333:role/aws-service-role/ssm-
incidents.amazonaws.com/AWSServiceRoleForIncidentManager", "targetAccount":
"RESPONSE_PLAN_OWNER_ACCOUNT"}}]' \
--engagements '["arn:aws:ssm-contacts:us-east-1:111122223333:contact/example"]'
```

Salida:

```
{
  "arn": "arn:aws:ssm-incidents::111122223333:response-plan/example-response"
}
```

Para obtener más información, consulte [Preparación de incidentes](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [CreateResponsePlan](#) la Referencia de AWS CLI comandos.

## create-timeline-event

En el siguiente ejemplo de código, se muestra cómo usar `create-timeline-event`.

AWS CLI

Ejemplo 1: Para crear un evento de cronología personalizado

El siguiente `create-timeline-event` ejemplo crea un evento de cronología personalizado a la hora especificada del incidente especificado.

```
aws ssm-incidents create-timeline-event \
--event-data "\"example timeline event\"" \
--event-time 2022-10-01T20:30:00.000 \
--event-type "Custom Event" \
--incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4EXAMPLE"
```

Salida:

```
{
  "eventId": "c0bcc885-a41d-eb01-b4ab-9d2deEXAMPLE",
```

```
"incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4EXAMPLE"
}
```

Ejemplo 2: Para crear un evento en la cronología con una nota del incidente

En el siguiente `create-timeline-event` ejemplo, se crea un evento de cronología que aparece en el panel «Notas del incidente».

```
aws ssm-incidents create-timeline-event \
  --event-data "\"New Note\"" \
  --event-type "Note" \
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/
Test/6cc46130-ca6c-3b38-68f1-f6abeEXAMPLE" \
  --event-time 2023-06-20T12:06:00.000 \
  --event-references '["resource": "arn:aws:ssm-incidents::111122223333:incident-
record/Test/6cc46130-ca6c-3b38-68f1-f6abeEXAMPLE"]'
```

Salida:

```
{
  "eventId": "a41dc885-c0bc-b4ab-eb01-de9d2EXAMPLE",
  "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4EXAMPLE"
}
```

Para obtener más información, consulte los [detalles del incidente](#) en la Guía del usuario del administrador de incidentes.

- Para API obtener más información, consulte [CreateTimelineEvent](#) la Referencia de AWS CLI comandos.

## **delete-incident-record**

En el siguiente ejemplo de código, se muestra cómo usar `delete-incident-record`.

### AWS CLI

Para eliminar un registro de incidentes

En el siguiente `delete-incident-record` ejemplo, se elimina el registro de incidentes especificado.

```
aws ssm-incidents delete-incident-record \  
  --arn "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-  
Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte el [seguimiento de incidentes](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [DeleteIncidentRecord](#) la Referencia de AWS CLI comandos.

## delete-replication-set

En el siguiente ejemplo de código, se muestra cómo usar delete-replication-set.

### AWS CLI

Para eliminar el conjunto de réplicas

En el siguiente delete-replication-set ejemplo, se elimina el conjunto de replicación de su cuenta de Amazon Web Services. Al eliminar el conjunto de replicación, también se eliminan todos los datos de Incident Manager. Esto no se puede deshacer.

```
aws ssm-incidents delete-replication-set \  
  --arn "arn:aws:ssm-incidents::111122223333:replication-set/c4bcb603-4bf9-  
bb3f-413c-08df53673b57"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso del conjunto de replicación de Incident Manager](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [DeleteReplicationSet](#) la Referencia de AWS CLI comandos.

## delete-resource-policy

En el siguiente ejemplo de código, se muestra cómo usar delete-resource-policy.

## AWS CLI

Para eliminar una política de recursos

En el siguiente `delete-resource-policy` ejemplo, se elimina una política de recursos de un plan de respuesta. Esto revocará el acceso del director o la organización con la que se compartió el plan de respuesta.

```
aws ssm-incidents delete-resource-policy \  
  --policy-id "be8b57191f0371f1c6827341aa3f0a03" \  
  --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con contactos y planes de respuesta compartidos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [DeleteResourcePolicy](#) la Referencia de AWS CLI comandos.

## `delete-response-plan`

En el siguiente ejemplo de código, se muestra cómo usar `delete-response-plan`.

## AWS CLI

Para eliminar un plan de respuesta

En el siguiente `delete-response-plan` ejemplo, se elimina el plan de respuesta especificado.

```
aws ssm-incidents delete-response-plan \  
  --arn "arn:aws:ssm-incidents::111122223333:response-plan/example-response"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Preparación de incidentes](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [DeleteResponsePlan](#) la Referencia de AWS CLI comandos.

## delete-timeline-event

En el siguiente ejemplo de código, se muestra cómo usar `delete-timeline-event`.

### AWS CLI

Para eliminar un evento de la línea de tiempo

En el siguiente `delete-timeline-event` ejemplo, se elimina un evento de cronología personalizado del registro de incidentes especificado.

```
aws ssm-incidents delete-timeline-event \  
  --event-id "c0bcc885-a41d-eb01-b4ab-9d2de193643c" \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [detalles del incidente](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [DeleteTimelineEvent](#) la Referencia de AWS CLI comandos.

## get-incident-record

En el siguiente ejemplo de código, se muestra cómo usar `get-incident-record`.

### AWS CLI

Para obtener un registro de incidentes

En el siguiente `get-incident-record` ejemplo, se obtienen detalles sobre el registro de incidentes especificado.

```
aws ssm-incidents get-incident-record \  
  --arn "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-  
Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Salida:

```
{  
  "incidentRecord": {
```

```

    "arn": "arn:aws:ssm-incidents::111122223333:incident-record/Example-
Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308",
    "automationExecutions": [],
    "creationTime": "2021-05-21T18:16:57.579000+00:00",
    "dedupeString": "c4bcc812-85e7-938d-2b78-17181176ee1a",
    "impact": 5,
    "incidentRecordSource": {
      "createdBy": "arn:aws:iam::111122223333:user/draliatp",
      "invokedBy": "arn:aws:iam::111122223333:user/draliatp",
      "source": "aws.ssm-incidents.custom"
    },
    "lastModifiedBy": "arn:aws:iam::111122223333:user/draliatp",
    "lastModifiedTime": "2021-05-21T18:16:59.149000+00:00",
    "notificationTargets": [],
    "status": "OPEN",
    "title": "Example-Incident"
  }
}

```

Para obtener más información, consulte los [detalles del incidente](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [GetIncidentRecord](#) la Referencia de AWS CLI comandos.

## get-replication-set

En el siguiente ejemplo de código, se muestra cómo usar `get-replication-set`.

### AWS CLI

Para obtener el conjunto de replicación

En el siguiente `get-replication-set` ejemplo, se obtienen los detalles del conjunto de replicación que Incident Manager utiliza para replicar y cifrar los datos de su cuenta de Amazon Web Services.

```

aws ssm-incidents get-replication-set \
  --arn "arn:aws:ssm-incidents::111122223333:replication-set/c4bcb603-4bf9-
bb3f-413c-08df53673b57"

```

Salida:

```
{
  "replicationSet": {
    "createdBy": "arn:aws:sts::111122223333:assumed-role/Admin/username",
    "createdTime": "2021-05-14T17:57:22.010000+00:00",
    "deletionProtected": false,
    "lastModifiedBy": "arn:aws:sts::111122223333:assumed-role/Admin/username",
    "lastModifiedTime": "2021-05-14T17:57:22.010000+00:00",
    "regionMap": {
      "us-east-1": {
        "sseKmsKeyId": "DefaultKey",
        "status": "ACTIVE"
      },
      "us-east-2": {
        "sseKmsKeyId": "DefaultKey",
        "status": "ACTIVE",
        "statusMessage": "Tagging inaccessible"
      }
    },
    "status": "ACTIVE"
  }
}
```

Para obtener más información, consulte [Uso del conjunto de replicación de Incident Manager](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [GetReplicationSet](#) la Referencia de AWS CLI comandos.

## get-resource-policies

En el siguiente ejemplo de código, se muestra cómo usar `get-resource-policies`.

### AWS CLI

Para enumerar las políticas de recursos de un plan de respuesta

En el siguiente `command-name` ejemplo, se enumeran las políticas de recursos asociadas al plan de respuesta especificado.

```
aws ssm-incidents get-resource-policies \
  --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

**Salida:**

```
{
  "resourcePolicies": [
    {
      "policyDocument": "{\n\"Version\": \"2012-10-17\", \"Statement\": [\n{\n\"Sid\": \"d901b37a-dbb0-458a-8842-75575c464219-external-principals\", \"Effect\": \"Allow\", \"Principal\": {\n\"AWS\": \"arn:aws:iam::222233334444:root\"}, \"Action\": [\n\"ssm-incidents:GetResponsePlan\", \"ssm-incidents:StartIncident\", \"ssm-incidents:UpdateIncidentRecord\", \"ssm-incidents:GetIncidentRecord\", \"ssm-incidents:CreateTimelineEvent\", \"ssm-incidents:UpdateTimelineEvent\", \"ssm-incidents:GetTimelineEvent\", \"ssm-incidents:ListTimelineEvents\", \"ssm-incidents:UpdateRelatedItems\", \"ssm-incidents:ListRelatedItems\"], \"Resource\": [\n\"arn:aws:ssm-incidents:*:111122223333:response-plan/Example-Response-Plan\", \"arn:aws:ssm-incidents:*:111122223333:incident-record/Example-Response-Plan/*\n\"]}]}",
      "policyId": "be8b57191f0371f1c6827341aa3f0a03",
      "ramResourceShareRegion": "us-east-1"
    }
  ]
}
```

Para obtener más información, consulte [Trabajar con contactos compartidos y planes de respuesta](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [GetResourcePolicies](#) la Referencia de AWS CLI comandos.

**get-response-plan**

En el siguiente ejemplo de código, se muestra cómo usar `get-response-plan`.

**AWS CLI**

Para obtener detalles de un plan de respuesta

En el siguiente `command-name` ejemplo, se obtienen detalles sobre un plan de respuesta específico de tu AWS cuenta.

```
aws ssm-incidents get-response-plan \
  --arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```



Salida:

```
{
  "actions": [
    {
      "ssmAutomation": {
        "documentName": "AWSIncidents-CriticalIncidentRunbookTemplate",
        "documentVersion": "$DEFAULT",
        "roleArn": "arn:aws:iam::111122223333:role/aws-service-role/ssm-
incidents.amazonaws.com/AWSServiceRoleForIncidentManager",
        "targetAccount": "RESPONSE_PLAN_OWNER_ACCOUNT"
      }
    }
  ],
  "arn": "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-
Plan",
  "chatChannel": {
    "chatbotSns": [
      "arn:aws:sns:us-east-1:111122223333:Standard_User"
    ]
  },
  "displayName": "Example response plan",
  "engagements": [
    "arn:aws:ssm-contacts:us-east-1:111122223333:contact/example"
  ],
  "incidentTemplate": {
    "impact": 5,
    "title": "Example-Incident"
  },
  "name": "Example-Response-Plan"
}
```

Para obtener más información, consulte [Preparación ante incidentes](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [GetResponsePlan](#) la Referencia de AWS CLI comandos.

## get-timeline-event

En el siguiente ejemplo de código, se muestra cómo usar `get-timeline-event`.

## AWS CLI

Para obtener detalles de un evento de la cronología

En el siguiente `get-timeline-event` ejemplo, se devuelven los detalles del evento de la cronología especificado.

```
aws ssm-incidents get-timeline-event \  
  --event-id 20bcc812-8a94-4cd7-520c-0ff742111424 \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Salida:

```
{  
  "event": {  
    "eventData": "\"Incident Started\"",  
    "eventId": "20bcc812-8a94-4cd7-520c-0ff742111424",  
    "eventTime": "2021-05-21T18:16:57+00:00",  
    "eventType": "Custom Event",  
    "eventUpdatedTime": "2021-05-21T18:16:59.944000+00:00",  
    "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"  
  }  
}
```

Para obtener más información, consulte los [detalles del incidente](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [GetTimelineEvent](#) la Referencia de AWS CLI comandos.

## **list-incident-records**

En el siguiente ejemplo de código, se muestra cómo usar `list-incident-records`.

## AWS CLI

Para enumerar los registros de incidentes

En el siguiente `command-name` ejemplo, se enumeran los registros de incidentes de su cuenta de Amazon Web Services.

```
aws ssm-incidents list-incident-records
```

Salida:

```
{
  "incidentRecordSummaries": [
    {
      "arn": "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308",
      "creationTime": "2021-05-21T18:16:57.579000+00:00",
      "impact": 5,
      "incidentRecordSource": {
        "createdBy": "arn:aws:iam::111122223333:user/draliatp",
        "invokedBy": "arn:aws:iam::111122223333:user/draliatp",
        "source": "aws.ssm-incidents.custom"
      },
      "status": "OPEN",
      "title": "Example-Incident"
    }
  ]
}
```

Para obtener más información, consulte la [lista de incidentes](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListIncidentRecords](#) la Referencia de AWS CLI comandos.

## list-related-items

En el siguiente ejemplo de código, se muestra cómo usar `list-related-items`.

AWS CLI

Para enumerar artículos relacionados

En el siguiente `list-related-items` ejemplo, se enumeran los elementos relacionados con el incidente especificado.

```
aws ssm-incidents list-related-items \
```

```
--incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Salida:

```
{  
  "relatedItems": [  
    {  
      "identifier": {  
        "type": "OTHER",  
        "value": {  
          "url": "https://console.aws.amazon.com/systems-manager/opsitems/  
oi-8ef82158e190/workbench?region=us-east-1"  
        }  
      },  
      "title": "Example related item"  
    },  
    {  
      "identifier": {  
        "type": "PARENT",  
        "value": {  
          "arn": "arn:aws:ssm:us-east-1:111122223333:opsitem/  
oi-8084126392ac"  
        }  
      },  
      "title": "parentItem"  
    }  
  ]  
}
```

Para obtener más información, consulte los [detalles del incidente](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListRelatedItems](#) la Referencia de AWS CLI comandos.

## list-replication-sets

En el siguiente ejemplo de código, se muestra cómo usar `list-replication-sets`.

AWS CLI

Para enumerar el conjunto de réplicas

En el siguiente `list-replication-set` ejemplo, se muestra el conjunto de réplicas que Incident Manager utiliza para replicar y cifrar los datos de su AWS cuenta.

```
aws ssm-incidents list-replication-sets
```

Salida:

```
{
  "replicationSetArns": [
    "arn:aws:ssm-incidents::111122223333:replication-set/c4bcb603-4bf9-
    bb3f-413c-08df53673b57"
  ]
}
```

Para obtener más información, consulte [Uso del conjunto de replicación de Incident Manager](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListReplicationSets](#) la Referencia de AWS CLI comandos.

## list-response-plans

En el siguiente ejemplo de código, se muestra cómo usar `list-response-plans`.

AWS CLI

Para ver una lista de los planes de respuesta disponibles

En el siguiente `list-response-plans` ejemplo, se enumeran los planes de respuesta disponibles en su cuenta de Amazon Web Services.

```
aws ssm-incidents list-response-plans
```

Salida:

```
{
  "responsePlanSummaries": [
    {
      "arn": "arn:aws:ssm-incidents::111122223333:response-plan/Example-
      Response-Plan",
      "displayName": "Example response plan",
    }
  ]
}
```

```
        "name": "Example-Response-Plan"
      }
    ]
  }
```

Para obtener más información, consulte [Preparación ante incidentes](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListResponsePlans](#) la Referencia de AWS CLI comandos.

## **list-tags-for-resource**

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un plan de respuesta

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas asociadas al plan de respuesta especificado.

```
aws ssm-incidents list-tags-for-resource \
  --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

Salida:

```
{
  "tags": {
    "group1": "1"
  }
}
```

Para obtener más información, consulte [Etiquetado](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## **list-timeline-events**

En el siguiente ejemplo de código, se muestra cómo usar `list-timeline-events`.

## AWS CLI

Para enumerar los eventos cronológicos de un incidente

En el siguiente command-name ejemplo, se enumeran los eventos cronológicos del incidente especificado.

```
aws ssm-incidents list-timeline-events \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Salida:

```
{  
  "eventSummaries": [  
    {  
      "eventId": "8cbcc889-35e1-a42d-2429-d6f100799915",  
      "eventTime": "2021-05-21T22:36:13.766000+00:00",  
      "eventType": "SSM Incident Record Update",  
      "eventUpdatedTime": "2021-05-21T22:36:13.766000+00:00",  
      "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-  
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"  
    },  
    {  
      "eventId": "a2bcc825-aab5-1787-c605-f9bb2640d85b",  
      "eventTime": "2021-05-21T18:58:46.443000+00:00",  
      "eventType": "SSM Incident Record Update",  
      "eventUpdatedTime": "2021-05-21T18:58:46.443000+00:00",  
      "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-  
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"  
    },  
    {  
      "eventId": "5abcc812-89c0-b0a8-9437-1c74223d4685",  
      "eventTime": "2021-05-21T18:16:59.149000+00:00",  
      "eventType": "SSM Incident Record Update",  
      "eventUpdatedTime": "2021-05-21T18:16:59.149000+00:00",  
      "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-  
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"  
    },  
    {  
      "eventId": "06bcc812-8820-405e-4065-8d2b14d29b92",  
      "eventTime": "2021-05-21T18:16:58+00:00",  
      "eventType": "SSM Automation Execution Start Failure for Incident",
```

```

    "eventUpdatedTime": "2021-05-21T18:16:58.689000+00:00",
    "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
  },
  {
    "eventId": "20bcc812-8a94-4cd7-520c-0ff742111424",
    "eventTime": "2021-05-21T18:16:57+00:00",
    "eventType": "Custom Event",
    "eventUpdatedTime": "2021-05-21T18:16:59.944000+00:00",
    "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
  },
  {
    "eventId": "c0bcc885-a41d-eb01-b4ab-9d2de193643c",
    "eventTime": "2020-10-01T20:30:00+00:00",
    "eventType": "Custom Event",
    "eventUpdatedTime": "2021-05-21T22:28:26.299000+00:00",
    "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
  }
]
}

```

Para obtener más información, consulte los [detalles del incidente](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListTimelineEvents](#) la Referencia de AWS CLI comandos.

## put-resource-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-resource-policy`.

### AWS CLI

Para compartir un plan de respuesta y los incidentes

En el siguiente `command-name` ejemplo, se agrega una política de recursos al `Example-Response-Plan` que comparte el plan de respuesta y los incidentes asociados con el principal especificado.

```
aws ssm-incidents put-resource-policy \
```



```
--resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan" \
--policy "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
\"ExampleResourcePolciy\",\"Effect\":\"Allow\",\"Principal\":{\"AWS\":
\"arn:aws:iam::222233334444:root\"},\"Action\":[\"ssm-incidents:GetResponsePlan
\",\"ssm-incidents:StartIncident\",\"ssm-incidents:UpdateIncidentRecord
\",\"ssm-incidents:GetIncidentRecord\",\"ssm-incidents:CreateTimelineEvent
\",\"ssm-incidents:UpdateTimelineEvent\",\"ssm-incidents:GetTimelineEvent
\",\"ssm-incidents:ListTimelineEvents\",\"ssm-incidents:UpdateRelatedItems
\",\"ssm-incidents:ListRelatedItems\"],\"Resource\":[\"arn:aws:ssm-
incidents*:111122223333:response-plan/Example-Response-Plan\",\"arn:aws:ssm-
incidents*:111122223333:incident-record/Example-Response-Plan/*\"]}]}"
```

Salida:

```
{
  "policyId": "be8b57191f0371f1c6827341aa3f0a03"
}
```

Para obtener más información, consulte [Trabajar con contactos y planes de respuesta compartidos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [PutResourcePolicy](#) la Referencia de AWS CLI comandos.

## start-incident

En el siguiente ejemplo de código, se muestra cómo usar start-incident.

AWS CLI

Para iniciar un incidente

En el siguiente start-incident ejemplo, se inicia un incidente mediante el plan de respuesta especificado.

```
aws ssm-incidents start-incident \
--response-plan-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

Salida:

```
{
  "incidentRecordArn": "arn:aws:ssm-incidents::682428703967:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
}
```

Para obtener más información, consulte [la creación de incidentes](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [StartIncident](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

### AWS CLI

Para etiquetar un plan de respuesta

El siguiente tag-resource ejemplo etiqueta un plan de respuesta especificado con el par clave-valor proporcionado.

```
aws ssm-incidents tag-resource \
  --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-
Response-Plan" \
  --tags '{"group1":"1"}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar untag-resource.

### AWS CLI

Para eliminar etiquetas de un plan de respuesta

En el siguiente `untag-resource` ejemplo, se eliminan las etiquetas especificadas del plan de respuesta.

```
aws ssm-incidents untag-resource \  
  --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-  
Response-Plan" \  
  --tag-keys ["group1"]
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-deletion-protection

En el siguiente ejemplo de código, se muestra cómo usar `update-deletion-protection`.

### AWS CLI

Para actualizar la protección contra la eliminación de conjuntos de replicación

En el siguiente `update-deletion-protection` ejemplo, se actualiza la protección contra la eliminación de su cuenta para evitar que se elimine la última región del conjunto de replications.

```
aws ssm-incidents update-deletion-protection \  
  --arn "arn:aws:ssm-incidents::111122223333:replication-set/  
a2bcc5c9-0f53-8047-7fef-c20749989b40" \  
  --deletion-protected
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso del conjunto de replicación de Incident Manager](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [UpdateDeletionProtection](#) la Referencia de AWS CLI comandos.

## update-incident-record

En el siguiente ejemplo de código, se muestra cómo usar `update-incident-record`.

## AWS CLI

Para actualizar un registro de incidentes

El siguiente command-name ejemplo resuelve el incidente especificado.

```
aws ssm-incidents update-incident-record \  
  --arn "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308" \  
  --status "RESOLVED"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [detalles del incidente](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [UpdateIncidentRecord](#) la Referencia de AWS CLI comandos.

## update-related-items

En el siguiente ejemplo de código, se muestra cómo usar update-related-items.

## AWS CLI

Para actualizar un elemento relacionado con un incidente

En el siguiente update-related-item ejemplo, se elimina un elemento relacionado del registro de incidentes especificado.

```
aws ssm-incidents update-related-items \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308" \  
  --related-items-update '{"itemToRemove": {"type": "OTHER", "value": {"url": "https://console.aws.amazon.com/systems-manager/opsitems/oi-8ef82158e190/workbench?region=us-east-1"}}}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [detalles del incidente](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [UpdateRelatedItems](#) la Referencia de AWS CLI comandos.

## update-replication-set

En el siguiente ejemplo de código, se muestra cómo usar `update-replication-set`.

### AWS CLI

Para actualizar un conjunto de réplicas

En el siguiente `command-name` ejemplo, se elimina la región `us-east-2` del conjunto de réplicas.

```
aws ssm-incidents update-replication-set \  
  --arn "arn:aws:ssm-incidents::111122223333:replication-set/  
a2bcc5c9-0f53-8047-7fef-c20749989b40" \  
  --actions '[{"deleteRegionAction": {"regionName": "us-east-2"}}]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso del conjunto de replicación de Incident Manager](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [UpdateReplicationSet](#) la Referencia de AWS CLI comandos.

## update-response-plan

En el siguiente ejemplo de código, se muestra cómo usar `update-response-plan`.

### AWS CLI

Para actualizar un plan de respuesta

El siguiente `update-response-plan` ejemplo elimina un canal de chat del plan de respuesta especificado.

```
aws ssm-incidents update-response-plan \  
  --arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"  
 \  
  --chat-channel '{"empty":{}}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Preparación ante incidentes](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [UpdateResponsePlan](#) la Referencia de AWS CLI comandos.

## update-timeline-event

En el siguiente ejemplo de código, se muestra cómo usar `update-timeline-event`.

### AWS CLI

Para actualizar un evento de la cronología

En el siguiente `update-timeline-event` ejemplo, se actualiza la hora a la que se produjo el evento.

```
aws ssm-incidents update-timeline-event \  
  --event-id 20bcc812-8a94-4cd7-520c-0ff742111424 \  
  --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/  
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308" \  
  --event-time "2021-05-21T18:10:57+00:00"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte los [detalles del incidente](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [UpdateTimelineEvent](#) la Referencia de AWS CLI comandos.

## Ejemplos de contactos de Incident Manager que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante el AWS Command Line Interface uso de Incident Manager Contacts.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **accept-page**

En el siguiente ejemplo de código, se muestra cómo usar `accept-page`.

#### AWS CLI

Para aceptar una página durante una interacción

En el siguiente `accept-page` ejemplo, se utiliza un código de aceptación enviado al canal de contacto para aceptar una página.

```
aws ssm-contacts accept-page \  
  --page-id "arn:aws:ssm-contacts:us-east-2:682428703967:page/  
akuam/94ea0c7b-56d9-46c3-b84a-a37c8b067ad3" \  
  --accept-type READ \  
  --accept-code 425440
```

Este comando no produce ningún resultado

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [AcceptPage](#) la Referencia de AWS CLI comandos.

### **activate-contact-channel**

En el siguiente ejemplo de código, se muestra cómo usar `activate-contact-channel`.

#### AWS CLI

Activa el canal de contacto de un contacto

En el siguiente `activate-contact-channel` ejemplo, se activa un canal de contacto y se puede utilizar como parte de un incidente.

```
aws ssm-contacts activate-contact-channel \  
  --contact-channel-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact-  
channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d" \  
  --activation-code "466136"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ActivateContactChannel](#) la Referencia de AWS CLI comandos.

## command-name

En el siguiente ejemplo de código, se muestra cómo usar command-name.

### AWS CLI

Para eliminar un contacto

En el siguiente command-name ejemplo, se elimina un contacto. Ya no se podrá acceder al contacto desde ningún plan de escalación que se refiera a él.

```
aws ssm-contacts delete-contact \  
  --contact-id "arn:aws:ssm-contacts:us-east-1:682428703967:contact/alejr"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [CommandName](#) la Referencia de AWS CLI comandos.

## create-contact-channel

En el siguiente ejemplo de código, se muestra cómo usar create-contact-channel.

### AWS CLI

Para crear un canal de contactos



Creará un canal de contacto del tipo SMS Akua Mansa. Los canales de contacto se pueden crear del tipo SMSEMAIL, o. VOICE

```
aws ssm-contacts create-contact-channel \  
  --contact-id "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam" \  
  --name "akuas sms-test" \  
  --type SMS \  
  --delivery-address '{"SimpleAddress": "+15005550199"}'
```

Salida:

```
{  
  "ContactChannelArn": "arn:aws:ssm-contacts:us-east-1:111122223333:contact-  
channel/akuam/02f506b9-ea5d-4764-af89-2daa793ff024"  
}
```

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [CreateContactChannel](#) la Referencia de AWS CLI comandos.

## create-contact

En el siguiente ejemplo de código, se muestra cómo usar create-contact.

### AWS CLI

Para crear un contacto

El siguiente create-contact ejemplo crea un contacto en su entorno con un plan en blanco. El plan se puede actualizar después de crear los canales de contacto. Utilice el create-contact-channel comando con el resultado ARN de este comando. Una vez que haya creado los canales de contacto para este contacto, utilice update-contact para actualizar el plan.

```
aws ssm-contacts create-contact \  
  --alias "akuam" \  
  --display-name "Akua Mansa" \  
  --type PERSONAL \  
  --plan '{"Stages": []}'
```

Salida:

```
{
  "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam"
}
```

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [CreateContact](#) la Referencia de AWS CLI comandos.

## **deactivate-contact-channel**

En el siguiente ejemplo de código, se muestra cómo usar `deactivate-contact-channel`.

### AWS CLI

Para desactivar un canal de contacto

El siguiente `deactivate-contact-channel` ejemplo desactiva un canal de contacto. La desactivación de un canal de contacto significa que el canal de contacto dejará de estar paginado durante un incidente. También puede reactivar un canal de contacto en cualquier momento mediante el comando `activate-contact-channel`

```
aws ssm-contacts deactivate-contact-channel \
  --contact-channel-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact-
channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [DeactivateContactChannel](#) la Referencia de AWS CLI comandos.

## **delete-contact-channel**

En el siguiente ejemplo de código, se muestra cómo usar `delete-contact-channel`.

### AWS CLI

Para eliminar un canal de contactos

El siguiente `delete-contact-channel` ejemplo elimina un canal de contactos. Al eliminar un canal de contacto, se garantiza que el canal de contacto no se llamará durante un incidente.

```
aws ssm-contacts delete-contact-channel \  
  --contact-channel-id "arn:aws:ssm-contacts:us-east-1:111122223333:contact-  
channel/akuam/13149bad-52ee-45ea-ae1e-45857f78f9b2"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [DeleteContactChannel](#) la Referencia de AWS CLI comandos.

## **delete-contact**

En el siguiente ejemplo de código, se muestra cómo usar `delete-contact`.

### AWS CLI

Para eliminar un contacto

En el siguiente `delete-contact` ejemplo, se elimina un contacto. Ya no se podrá acceder al contacto desde ningún plan de escalación que se refiera a él.

```
aws ssm-contacts delete-contact \  
  --contact-id "arn:aws:ssm-contacts:us-east-1:111122223333:contact/alejr"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [DeleteContact](#) la Referencia de AWS CLI comandos.

## **describe-engagement**

En el siguiente ejemplo de código, se muestra cómo usar `describe-engagement`.

### AWS CLI

Para describir los detalles de un enfrentamiento

En el siguiente `describe-engagement` ejemplo, se enumeran los detalles de una interacción con un contacto o un plan de escalación. El asunto y el contenido se envían a los canales de contacto.

```
aws ssm-contacts describe-engagement \  
  --engagement-id "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/  
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356"
```

Salida:

```
{  
  "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/  
example_escalation",  
  "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/  
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356",  
  "Sender": "cli",  
  "Subject": "cli-test",  
  "Content": "Testing engagements via CLI",  
  "PublicSubject": "cli-test",  
  "PublicContent": "Testing engagements va CLI",  
  "StartTime": "2021-05-18T18:25:41.151000+00:00"  
}
```

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [DescribeEngagement](#) la Referencia de AWS CLI comandos.

## **describe-page**

En el siguiente ejemplo de código, se muestra cómo usar `describe-page`.

AWS CLI

Para incluir los detalles de una página en un canal de contacto

En el siguiente `describe-page` ejemplo, se muestran los detalles de una página para un canal de contactos. La página incluirá el asunto y el contenido proporcionados.

```
aws ssm-contacts describe-page \  

```

```
--page-id "arn:aws:ssm-contacts:us-east-2:111122223333:page/akuam/ad0052bd-
e606-498a-861b-25726292eb93"
```

Salida:

```
{
  "PageArn": "arn:aws:ssm-contacts:us-east-2:111122223333:page/akuam/ad0052bd-
e606-498a-861b-25726292eb93",
  "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/
akuam/78a29753-3674-4ac5-9f83-0468563567f0",
  "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam",
  "Sender": "cli",
  "Subject": "cli-test",
  "Content": "Testing engagements via CLI",
  "PublicSubject": "cli-test",
  "PublicContent": "Testing engagements va CLI",
  "SentTime": "2021-05-18T18:43:29.301000+00:00",
  "ReadTime": "2021-05-18T18:43:55.708000+00:00",
  "DeliveryTime": "2021-05-18T18:43:55.265000+00:00"
}
```

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [DescribePage](#) la Referencia de AWS CLI comandos.

## get-contact-channel

En el siguiente ejemplo de código, se muestra cómo usar `get-contact-channel`.

AWS CLI

Para enumerar los detalles de un canal de contacto

El siguiente `get-contact-channel` ejemplo muestra los detalles de un canal de contacto.

```
aws ssm-contacts get-contact-channel \
  --contact-channel-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact-
channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d"
```

Salida:

```
{
  "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam",
  "ContactChannelArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
  "Name": "akuas sms",
  "Type": "SMS",
  "DeliveryAddress": {
    "SimpleAddress": "+15005550199"
  },
  "ActivationStatus": "ACTIVATED"
}
```

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [GetContactChannel](#) en la Referencia de AWS CLI comandos.

## get-contact-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-contact-policy`.

### AWS CLI

Para enumerar las políticas de recursos de un contacto

En el siguiente `get-contact-policy` ejemplo, se enumeran las políticas de recursos asociadas al contacto especificado.

```
aws ssm-contacts get-contact-policy \
  --contact-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam"
```

Salida:

```
{
  "ContactArn": "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam",
  "Policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\": \"SharePolicyForDocumentationDralia\", \"Effect\":\"Allow\", \"Principal\": {\"AWS\": \"222233334444\"}, \"Action\": [\"ssm-contacts:GetContact\", \"ssm-contacts:StartEngagement\", \"ssm-contacts:DescribeEngagement\", \"ssm-contacts:ListPagesByEngagement\", \"ssm-contacts:StopEngagement\"], \"Resource\": [\"arn:aws:ssm-contacts:*:111122223333:contact/akuam\", \"arn:aws:ssm-contacts:*:111122223333:engagement/akuam/*\"]}]}"
```

```
}

```

Para obtener más información, consulte [Trabajar con contactos compartidos y planes de respuesta](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [GetContactPolicy](#) la Referencia de AWS CLI comandos.

## get-contact

En el siguiente ejemplo de código, se muestra cómo usar `get-contact`.

### AWS CLI

Ejemplo 1: Para describir un plan de contactos

El siguiente `get-contact` ejemplo describe un contacto.

```
aws ssm-contacts get-contact \
  --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam"
```

Salida:

```
{
  "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam",
  "Alias": "akuam",
  "DisplayName": "Akua Mansa",
  "Type": "PERSONAL",
  "Plan": {
    "Stages": [
      {
        "DurationInMinutes": 5,
        "Targets": [
          {
            "ChannelTargetInfo": {
              "ContactChannelId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/beb25840-5ac8-4644-95cc-7a8de390fa65",
              "RetryIntervalInMinutes": 1
            }
          }
        ]
      }
    ],
  },
  {
```

```

        "DurationInMinutes": 5,
        "Targets": [
            {
                "ChannelTargetInfo": {
                    "ContactChannelId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/49f3c24d-5f9f-4638-ae25-3f49e04229ad",
                    "RetryIntervalInMinutes": 1
                }
            }
        ],
    },
    {
        "DurationInMinutes": 5,
        "Targets": [
            {
                "ChannelTargetInfo": {
                    "ContactChannelId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/77d4f447-f619-4954-afff-85551e369c2a",
                    "RetryIntervalInMinutes": 1
                }
            }
        ]
    }
]
}
}

```

Ejemplo 2: Para describir un plan de escalación

El siguiente `get-contact` ejemplo describe un plan de escalamiento.

```

aws ssm-contacts get-contact \
--contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation"

```

Salida:

```

{
  "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation",
  "Alias": "example_escalation",
  "DisplayName": "Example Escalation",
  "Type": "ESCALATION",

```



```
"Plan": {
  "Stages": [
    {
      "DurationInMinutes": 5,
      "Targets": [
        {
          "ContactTargetInfo": {
            "ContactId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact/akuam",
            "IsEssential": true
          }
        }
      ]
    },
    {
      "DurationInMinutes": 5,
      "Targets": [
        {
          "ContactTargetInfo": {
            "ContactId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact/alejrr",
            "IsEssential": false
          }
        }
      ]
    },
    {
      "DurationInMinutes": 0,
      "Targets": [
        {
          "ContactTargetInfo": {
            "ContactId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact/anasi",
            "IsEssential": false
          }
        }
      ]
    }
  ]
}
```

Para obtener más información, consulte [Contactos](#) en la guía del usuario de Incident Manager.

- Para API obtener más información, consulte [GetContact](#) la Referencia de AWS CLI comandos.

## list-contact-channels

En el siguiente ejemplo de código, se muestra cómo usar `list-contact-channels`.

### AWS CLI

Para enumerar los canales de contacto de un contacto

El siguiente `list-contact-channels` ejemplo muestra los canales de contacto disponibles del contacto especificado.

```
aws ssm-contacts list-contact-channels \  
  --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam"
```

Salida:

```
{  
  [  
    {  
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/  
akuam",  
      "Name": "akuas email",  
      "Type": "EMAIL",  
      "DeliveryAddress": {  
        "SimpleAddress": "akuam@example.com"  
      },  
      "ActivationStatus": "NOT_ACTIVATED"  
    },  
    {  
      "ContactChannelArn": "arn:aws:ssm-contacts:us-  
east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",  
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/  
akuam",  
      "Name": "akuas sms",  
      "Type": "SMS",  
      "DeliveryAddress": {  
        "SimpleAddress": "+15005550100"  
      },  
      "ActivationStatus": "ACTIVATED"  
    }  
  ]  
}
```

```
]
}
```

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListContactChannels](#) la Referencia de AWS CLI comandos.

## list-contacts

En el siguiente ejemplo de código, se muestra cómo usar `list-contacts`.

### AWS CLI

Para enumerar todos los planes y contactos de escalamiento

En el siguiente `list-contacts` ejemplo, se enumeran los contactos y los planes de escalamiento de su cuenta.

```
aws ssm-contacts list-contacts
```

Salida:

```
{
  "Contacts": [
    {
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
      akuam",
      "Alias": "akuam",
      "DisplayName": "Akua Mansa",
      "Type": "PERSONAL"
    },
    {
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
      alejr",
      "Alias": "alejr",
      "DisplayName": "Alejandro Rosalez",
      "Type": "PERSONAL"
    },
    {
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
      anasi",
      "Alias": "anasi",
```

```

        "DisplayName": "Ana Carolina Silva",
        "Type": "PERSONAL"
    },
    {
        "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation",
        "Alias": "example_escalation",
        "DisplayName": "Example Escalation",
        "Type": "ESCALATION"
    }
]
}

```

Para obtener más información, consulte [Contactos](#) en la guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListContacts](#) la Referencia de AWS CLI comandos.

## list-engagements

En el siguiente ejemplo de código, se muestra cómo usar `list-engagements`.

### AWS CLI

Para ver una lista de todos los enfrentamientos

En el siguiente `list-engagements` ejemplo, se enumeran las interacciones con los planes de escalamiento y los contactos. También puede enumerar las interacciones de un solo incidente.

```
aws ssm-contacts list-engagements
```

Salida:

```

{
  "Engagements": [
    {
      "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/91792571-0b53-4821-9f73-d25d13d9e529",
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
      "Sender": "cli",
      "StartTime": "2021-05-18T20:37:50.300000+00:00"
    },
    {

```

```

    "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/78a29753-3674-4ac5-9f83-0468563567f0",
    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
    "Sender": "cli",
    "StartTime": "2021-05-18T18:40:26.666000+00:00"
  },
  {
    "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356",
    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation",
    "Sender": "cli",
    "StartTime": "2021-05-18T18:25:41.151000+00:00"
  },
  {
    "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/607ced0e-e8fa-4ea7-8958-a237b8803f8f",
    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
    "Sender": "cli",
    "StartTime": "2021-05-18T18:20:58.093000+00:00"
  }
]
}

```

Para obtener más información, consulte [los contactos](#) en la guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListEngagements](#) la Referencia de AWS CLI comandos.

## list-page-receipts

En el siguiente ejemplo de código, se muestra cómo usar `list-page-receipts`.

### AWS CLI

Para enumerar las páginas de recepción

En el siguiente `command-name` ejemplo se indica si un contacto ha recibido o no una página.

```
aws ssm-contacts list-page-receipts \
```

```
--page-id "arn:aws:ssm-contacts:us-east-2:111122223333:page/
akuam/94ea0c7b-56d9-46c3-b84a-a37c8b067ad3"
```

Salida:

```
{
  "Receipts": [
    {
      "ContactChannelArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
      "ReceiptType": "DELIVERED",
      "ReceiptInfo": "425440",
      "ReceiptTime": "2021-05-18T20:42:57.485000+00:00"
    },
    {
      "ContactChannelArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
      "ReceiptType": "READ",
      "ReceiptInfo": "425440",
      "ReceiptTime": "2021-05-18T20:42:57.907000+00:00"
    },
    {
      "ContactChannelArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
      "ReceiptType": "SENT",
      "ReceiptInfo": "SM6656c19132f1465f9c9c1123a5dde7c9",
      "ReceiptTime": "2021-05-18T20:40:52.962000+00:00"
    }
  ]
}
```

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListPageReceipts](#) la Referencia de AWS CLI comandos.

## list-pages-by-contact

En el siguiente ejemplo de código, se muestra cómo usar `list-pages-by-contact`.

AWS CLI

Para enumerar páginas por contacto

En el siguiente `list-pages-by-contact` ejemplo, se muestran todas las páginas del contacto especificado.

```
aws ssm-contacts list-pages-by-contact \
  --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam"
```

Salida:

```
{
  "Pages": [
    {
      "PageArn": "arn:aws:ssm-contacts:us-east-2:111122223333:page/akuam/
ad0052bd-e606-498a-861b-25726292eb93",
      "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/78a29753-3674-4ac5-9f83-0468563567f0",
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
      "Sender": "cli",
      "SentTime": "2021-05-18T18:43:29.301000+00:00",
      "DeliveryTime": "2021-05-18T18:43:55.265000+00:00",
      "ReadTime": "2021-05-18T18:43:55.708000+00:00"
    }
  ]
}
```

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListPagesByContact](#) la Referencia de AWS CLI comandos.

## **list-pages-by-engagement**

En el siguiente ejemplo de código, se muestra cómo usar `list-pages-by-engagement`.

AWS CLI

Para ver una lista de páginas para contactar con los canales creados a partir de una interacción.

En el siguiente `list-pages-by-engagement` ejemplo, se enumeran las páginas que aparecieron durante la participación en el plan de participación definido.

```
aws ssm-contacts list-pages-by-engagement \
```

```
--engagement-id "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/akuam/78a29753-3674-4ac5-9f83-0468563567f0"
```

Salida:

```
{
  "Pages": [
    {
      "PageArn": "arn:aws:ssm-contacts:us-east-2:111122223333:page/akuam/ad0052bd-e606-498a-861b-25726292eb93",
      "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/akuam/78a29753-3674-4ac5-9f83-0468563567f0",
      "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam",
      "Sender": "cli",
      "SentTime": "2021-05-18T18:40:27.245000+00:00"
    }
  ]
}
```

Para obtener más información, consulte [los contactos](#) en la guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListPagesByEngagement](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

AWS CLI

Para enumerar las etiquetas de un contacto

El siguiente `list-tags-for-resource` ejemplo muestra las etiquetas del contacto especificado.

```
aws ssm-contacts list-tags-for-resource \
  --resource-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam"
```

Salida:

```
{
```



```

    "Tags": [
      {
        "Key": "group1",
        "Value": "1"
      }
    ]
  }

```

Para obtener más información, consulte [Etiquetado](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## put-contact-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-contact-policy`.

### AWS CLI

Para compartir un contacto y compromisos

En el siguiente `put-contact-policy` ejemplo, se agrega una política de recursos al contacto Akua que comparte el contacto y las interacciones relacionadas con el director.

```

aws ssm-contacts put-contact-policy \
  --contact-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam" \
  --policy "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
  \"ExampleResourcePolicy\",\"Action\":[\"ssm-contacts:GetContact\",\"ssm-
  contacts:StartEngagement\",\"ssm-contacts:DescribeEngagement\",\"ssm-
  contacts:ListPagesByEngagement\",\"ssm-contacts:StopEngagement\"],
  \"Principal\":{\"AWS\":\"222233334444\"},\"Effect\":\"Allow\",\"Resource
  \":[\"arn:aws:ssm-contacts:*:111122223333:contact/akuam\",\"arn:aws:ssm-
  contacts:*:111122223333:engagement/akuam/*\"]}]}"

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con contactos compartidos y planes de respuesta](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [PutContactPolicy](#) la Referencia de AWS CLI comandos.

## send-activation-code

En el siguiente ejemplo de código, se muestra cómo usar `send-activation-code`.

### AWS CLI

Para enviar un código de activación

El siguiente `send-activation-code` ejemplo envía un código de activación y un mensaje al canal de contacto especificado.

```
aws ssm-contacts send-activation-code \  
  --contact-channel-id "arn:aws:ssm-contacts:us-east-1:111122223333:contact-  
channel/akuam/8ddae2d1-12c8-4e45-b852-c8587266c400"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [SendActivationCode](#) la Referencia de AWS CLI comandos.

## start-engagement

En el siguiente ejemplo de código, se muestra cómo usar `start-engagement`.

### AWS CLI

Ejemplo 1: Para paginar los canales de contacto de un contacto

Las siguientes `start-engagement` páginas son los canales de contacto de los contactos. El remitente, el asunto, el asunto público y el contenido público están libres de campos. Incident Manager envía el asunto y el contenido a los canales proporcionados VOICE o de EMAIL contacto. Incident Manager envía el asunto público y el contenido público a los canales de SMS contacto proporcionados. El remitente se utiliza para rastrear quién inició la interacción.

```
aws ssm-contacts start-engagement \  
  --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam" \  
  --sender "cli" \  
  --subject "cli-test" \  
  --content "Testing engagements via CLI" \  
  \
```

```
--public-subject "cli-test" \
--public-content "Testing engagements va CLI"
```

Salida:

```
{
  "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/
akuam/607ced0e-e8fa-4ea7-8958-a237b8803f8f"
}
```

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

Ejemplo 2: Para localizar a un contacto en el plan de escalación proporcionado.

Lo siguiente `start-engagement` involucra a los contactos a través de un plan de escalación. Cada contacto se localiza de acuerdo con su plan de participación.

```
aws ssm-contacts start-engagement \
  --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation" \
  --sender "cli" \
  --subject "cli-test" \
  --content "Testing engagements via CLI" \
  --public-subject "cli-test" \
  --public-content "Testing engagements va CLI"
```

Salida:

```
{
  "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356"
}
```

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [StartEngagement](#) la Referencia de AWS CLI comandos.

## stop-engagement

En el siguiente ejemplo de código, se muestra cómo usar `stop-engagement`.

## AWS CLI

Para detener un enfrentamiento

El siguiente stop-engagement ejemplo impide que una interacción llegue a más contactos y canales de contacto.

```
aws ssm-contacts stop-engagement \  
  --engagement-id "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/  
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [StopEngagement](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

## AWS CLI

Para etiquetar un contacto

El siguiente tag-resource ejemplo etiqueta un contacto especificado con el par clave-valor proporcionado.

```
aws ssm-contacts tag-resource \  
  --resource-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam" \  
  --tags '[{"Key":"group1","Value":"1"}]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar untag-resource.

## AWS CLI

Para eliminar etiquetas de un contacto

En el siguiente `untag-resource` ejemplo, se quita la etiqueta `group1` del contacto especificado.

```
aws ssm-contacts untag-resource \  
  --resource-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam" \  
  --tag-keys "group1"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-contact-channel

En el siguiente ejemplo de código, se muestra cómo usar `update-contact-channel`.

## AWS CLI

Para actualizar un canal de contacto

El siguiente `update-contact-channel` ejemplo actualiza el nombre y la dirección de entrega de un canal de contacto.

```
aws ssm-contacts update-contact-channel \  
  --contact-channel-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact-  
channel/akuam/49f3c24d-5f9f-4638-ae25-3f49e04229ad" \  
  --name "akuas voice channel" \  
  --delivery-address '{"SimpleAddress": "+15005550198"}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [UpdateContactChannel](#) la Referencia de AWS CLI comandos.

## update-contact

En el siguiente ejemplo de código, se muestra cómo usar `update-contact`.

### AWS CLI

Para actualizar el plan de compromiso de contacto

El siguiente `update-contact` ejemplo actualiza el plan de participación del contacto Akua para incluir los tres tipos de canales de contactos. Esto se hace después de crear los canales de contacto para Akua.

```
aws ssm-contacts update-contact \  
  --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam" \  
  --plan '{"Stages": [{"DurationInMinutes": 5, "Targets": [{"ChannelTargetInfo":  
  {"ContactChannelId": "arn:aws:ssm-contacts:us-east-2:111122223333:contact-  
channel/akuam/beb25840-5ac8-4644-95cc-7a8de390fa65", "RetryIntervalInMinutes":  
1 }]}], {"DurationInMinutes": 5, "Targets": [{"ChannelTargetInfo":  
{"ContactChannelId": "arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/  
akuam/49f3c24d-5f9f-4638-ae25-3f49e04229ad", "RetryIntervalInMinutes": 1}]},  
{"DurationInMinutes": 5, "Targets": [{"ChannelTargetInfo": {"ContactChannelId":  
"arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/akuam/77d4f447-  
f619-4954-afff-85551e369c2a", "RetryIntervalInMinutes": 1 }]}]}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Contactos](#) en la Guía del usuario de Incident Manager.

- Para API obtener más información, consulte [UpdateContact](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon Inspector que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con Amazon Inspector.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **add-attributes-to-findings**

En el siguiente ejemplo de código, se muestra cómo usar `add-attributes-to-findings`.

#### AWS CLI

Para añadir atributos a los hallazgos

El siguiente `add-attribute-to-finding` comando asigna un atributo con la clave de `Example` y el valor de `example` al hallazgo con el ARN `dearn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-811VIE0D/run/0-Z02cjjug/finding/0-T8yM9mEU`:

```
aws inspector add-attributes-to-findings --finding-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-811VIE0D/run/0-Z02cjjug/finding/0-T8yM9mEU --attributes key=Example,value=example
```

Salida:

```
{
  "failedItems": {}
}
```

Para obtener más información, consulta los hallazgos de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [AddAttributesToFindings](#) la Referencia de AWS CLI comandos.

### **create-assessment-target**

En el siguiente ejemplo de código, se muestra cómo usar `create-assessment-target`.

#### AWS CLI

Para crear un objetivo de evaluación

El siguiente `create-assessment-target` comando crea un objetivo de evaluación denominado `ExampleAssessmentTarget` mediante el grupo de recursos con el signo ARN `dearn:aws:inspector:us-west-2:123456789012:resourcegroup/0-AB6DMKnv`:

```
aws inspector create-assessment-target --assessment-target-name ExampleAssessmentTarget --resource-group-arn arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-AB6DMKnv
```

Salida:

```
{
  "assessmentTargetArn": "arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX"
}
```

Para obtener más información, consulte los objetivos de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [CreateAssessmentTarget](#) la Referencia de AWS CLI comandos.

## **create-assessment-template**

En el siguiente ejemplo de código, se muestra cómo usar `create-assessment-template`.

### AWS CLI

Para crear una plantilla de evaluación

El siguiente `create-assessment-template` comando crea una plantilla de evaluación llamada objetivo `ExampleAssessmentTemplate` de la evaluación con ARN la `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX` siguiente información:

```
aws inspector create-assessment-template --assessment-target-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX --assessment-template-name ExampleAssessmentTemplate --duration-in-seconds 180 --rules-package-arns arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p --user-attributes-for-findings key=ExampleTag,value=examplevalue
```

Salida:



```
{
  "assessmentTemplateArn": "arn:aws:inspector:us-west-2:123456789012:target/0-
nvgVhaxX/template/0-it5r2S4T"
}
```

Para obtener más información, consulte las plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [CreateAssessmentTemplate](#) la Referencia de AWS CLI comandos.

## create-resource-group

En el siguiente ejemplo de código, se muestra cómo usar `create-resource-group`.

### AWS CLI

Para crear un grupo de recursos

El siguiente `create-resource-group` comando crea un grupo de recursos con la clave de etiqueta `Name` y el valor de `example`:

```
aws inspector create-resource-group --resource-group-tags key=Name,value=example
```

Salida:

```
{
  "resourceGroupArn": "arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-
AB6DMKnv"
}
```

Para obtener más información, consulte los objetivos de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [CreateResourceGroup](#) la Referencia de AWS CLI comandos.

## delete-assessment-run

En el siguiente ejemplo de código, se muestra cómo usar `delete-assessment-run`.

## AWS CLI

Para eliminar una ejecución de evaluación

El siguiente `delete-assessment-run` comando elimina la evaluación ejecutada con el ARN `dearn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-11LMTAVe`:

```
aws inspector delete-assessment-run --assessment-run-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-11LMTAVe
```

Para obtener más información, consulte las plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [DeleteAssessmentRun](#) la Referencia de AWS CLI comandos.

## `delete-assessment-target`

En el siguiente ejemplo de código, se muestra cómo usar `delete-assessment-target`.

### AWS CLI

Para eliminar un objetivo de evaluación

El siguiente `delete-assessment-target` comando elimina el objetivo de la evaluación con el signo ARN `dearn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq`:

```
aws inspector delete-assessment-target --assessment-target-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq
```

Para obtener más información, consulte los objetivos de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [DeleteAssessmentTarget](#) la Referencia de AWS CLI comandos.

## `delete-assessment-template`

En el siguiente ejemplo de código, se muestra cómo usar `delete-assessment-template`.

## AWS CLI

Para eliminar una plantilla de evaluación,

El siguiente `delete-assessment-template` comando elimina la plantilla de evaluación con la ARN `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T`:

```
aws inspector delete-assessment-template --assessment-template-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T
```

Para obtener más información, consulte las plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [DeleteAssessmentTemplate](#) la Referencia de AWS CLI comandos.

## describe-assessment-runs

En el siguiente ejemplo de código, se muestra cómo usar `describe-assessment-runs`.

## AWS CLI

Para describir las ejecuciones de evaluación

El siguiente `describe-assessment-run` comando describe una evaluación ejecutada con la ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE`:

```
aws inspector describe-assessment-runs --assessment-run-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE
```

Salida:

```
{
  "assessmentRuns": [
    {
      "arn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE",
      "assessmentTemplateArn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw",
      "completedAt": 1458680301.4,

```

```
"createdAt": 1458680170.035,
"dataCollected": true,
"durationInSeconds": 3600,
"name": "Run 1 for ExampleAssessmentTemplate",
"notifications": [],
"rulesPackageArns": [
  "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-X1KXtawP"
],
"startedAt": 1458680170.161,
"state": "COMPLETED",
"stateChangedAt": 1458680301.4,
"stateChanges": [
  {
    "state": "CREATED",
    "stateChangedAt": 1458680170.035
  },
  {
    "state": "START_DATA_COLLECTION_PENDING",
    "stateChangedAt": 1458680170.065
  },
  {
    "state": "START_DATA_COLLECTION_IN_PROGRESS",
    "stateChangedAt": 1458680170.096
  },
  {
    "state": "COLLECTING_DATA",
    "stateChangedAt": 1458680170.161
  },
  {
    "state": "STOP_DATA_COLLECTION_PENDING",
    "stateChangedAt": 1458680239.883
  },
  {
    "state": "DATA_COLLECTED",
    "stateChangedAt": 1458680299.847
  },
  {
    "state": "EVALUATING_RULES",
    "stateChangedAt": 1458680300.099
  },
  {
    "state": "COMPLETED",
    "stateChangedAt": 1458680301.4
  }
]
```

```

        ],
        "userAttributesForFindings": []
    }
],
"failedItems": {}
}

```

Para obtener más información, consulte las plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [DescribeAssessmentRuns](#) la Referencia de AWS CLI comandos.

## describe-assessment-targets

En el siguiente ejemplo de código, se muestra cómo usar `describe-assessment-targets`.

### AWS CLI

Para describir los objetivos de la evaluación

El siguiente `describe-assessment-targets` comando describe el objetivo de la evaluación con el ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq`:

```
aws inspector describe-assessment-targets --assessment-target-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq
```

Salida:

```

{
  "assessmentTargets": [
    {
      "arn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq",
      "createdAt": 1458074191.459,
      "name": "ExampleAssessmentTarget",
      "resourceGroupArn": "arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-PyGXopAI",
      "updatedAt": 1458074191.459
    }
  ],
  "failedItems": {}
}

```

Para obtener más información, consulte los objetivos de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [DescribeAssessmentTargets](#) la Referencia de AWS CLI comandos.

## describe-assessment-templates

En el siguiente ejemplo de código, se muestra cómo usar `describe-assessment-templates`.

### AWS CLI

Para describir las plantillas de evaluación

El siguiente `describe-assessment-templates` comando describe la plantilla de evaluación con ARN los siguientes valores `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw`:

```
aws inspector describe-assessment-templates --assessment-template-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw
```

Salida:

```
{
  "assessmentTemplates": [
    {
      "arn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw",
      "assessmentTargetArn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq",
      "createdAt": 1458074191.844,
      "durationInSeconds": 3600,
      "name": "ExampleAssessmentTemplate",
      "rulesPackageArns": [
        "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-X1KXtawP"
      ],
      "userAttributesForFindings": []
    }
  ],
  "failedItems": {}
}
```

Para obtener más información, consulte las plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [DescribeAssessmentTemplates](#) la Referencia de AWS CLI comandos.

## **describe-cross-account-access-role**

En el siguiente ejemplo de código, se muestra cómo usar `describe-cross-account-access-role`.

### AWS CLI

Para describir la función de acceso entre cuentas

El siguiente `describe-cross-account-access-role` comando describe la IAM función que permite a Amazon Inspector acceder a tu AWS cuenta:

```
aws inspector describe-cross-account-access-role
```

Salida:

```
{
  "registeredAt": 1458069182.826,
  "roleArn": "arn:aws:iam::123456789012:role/inspector",
  "valid": true
}
```

Para obtener más información, consulta [Cómo configurar Amazon Inspector](#) en la guía Amazon Inspector.

- Para API obtener más información, consulte [DescribeCrossAccountAccessRole](#) la Referencia de AWS CLI comandos.

## **describe-findings**

En el siguiente ejemplo de código, se muestra cómo usar `describe-findings`.

### AWS CLI

Para describir los hallazgos

El describe-findings comando siguiente describe el hallazgo con el valor ARN dearn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE/finding/0-HwPnsDm4:

```
aws inspector describe-findings --finding-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE/finding/0-HwPnsDm4
```

Salida:

```
{
  "failedItems": {},
  "findings": [
    {
      "arn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE/finding/0-HwPnsDm4",
      "assetAttributes": {
        "ipv4Addresses": [],
        "schemaVersion": 1
      },
      "assetType": "ec2-instance",
      "attributes": [],
      "confidence": 10,
      "createdAt": 1458680301.37,
      "description": "Amazon Inspector did not find any potential security issues during this assessment.",
      "indicatorOfCompromise": false,
      "numericSeverity": 0,
      "recommendation": "No remediation needed.",
      "schemaVersion": 1,
      "service": "Inspector",
      "serviceAttributes": {
        "assessmentRunArn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE",
        "rulesPackageArn": "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-X1KXtawP",
        "schemaVersion": 1
      },
      "severity": "Informational",
      "title": "No potential security issues found",
      "updatedAt": 1458680301.37,
      "userAttributes": []
    }
  ]
}
```



```
]
}
```

Para obtener más información, consulta los hallazgos de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [DescribeFindings](#) la Referencia de AWS CLI comandos.

## describe-resource-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-resource-groups`.

### AWS CLI

Para describir los grupos de recursos

El siguiente `describe-resource-groups` comando describe el grupo de recursos con el valor ARN `arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-PyGXopAI`:

```
aws inspector describe-resource-groups --resource-group-arns arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-PyGXopAI
```

Salida:

```
{
  "failedItems": {},
  "resourceGroups": [
    {
      "arn": "arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-
PyGXopAI",
      "createdAt": 1458074191.098,
      "tags": [
        {
          "key": "Name",
          "value": "example"
        }
      ]
    }
  ]
}
```

Para obtener más información, consulte los objetivos de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [DescribeResourceGroups](#) la Referencia de AWS CLI comandos.

## describe-rules-packages

En el siguiente ejemplo de código, se muestra cómo usar `describe-rules-packages`.

### AWS CLI

Para describir los paquetes de reglas

El siguiente `describe-rules-packages` comando describe el paquete de reglas con el valor ARN `arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p`:

```
aws inspector describe-rules-packages --rules-package-arns arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p
```

Salida:

```
{
  "failedItems": {},
  "rulesPackages": [
    {
      "arn": "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p",
      "description": "The rules in this package help verify whether the EC2 instances in your application are exposed to Common Vulnerabilities and Exposures (CVEs). Attacks can exploit unpatched vulnerabilities to compromise the confidentiality, integrity, or availability of your service or data. The CVE system provides a reference for publicly known information security vulnerabilities and exposures. For more information, see [https://cve.mitre.org/](https://cve.mitre.org/). If a particular CVE appears in one of the produced Findings at the end of a completed Inspector assessment, you can search [https://cve.mitre.org/](https://cve.mitre.org/) using the CVE's ID (for example, \"CVE-2009-0021\") to find detailed information about this CVE, its severity, and how to mitigate it. ",
      "name": "Common Vulnerabilities and Exposures",
      "provider": "Amazon Web Services, Inc.",
      "version": "1.1"
    }
  ]
}
```

```
    }
  ]
}
```

Para obtener más información, consulta Amazon Inspector Rules Packages and Rules en la guía Amazon Inspector.

- Para API obtener más información, consulte [DescribeRulesPackages](#) la Referencia de AWS CLI comandos.

## get-telemetry-metadata

En el siguiente ejemplo de código, se muestra cómo usar `get-telemetry-metadata`.

### AWS CLI

Para obtener los metadatos de telemetría

El siguiente `get-telemetry-metadata` comando genera información sobre los datos que se recopilan para la evaluación que se ejecuta con: ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE`

```
aws inspector get-telemetry-metadata --assessment-run-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE
```

Salida:

```
{
  "telemetryMetadata": [
    {
      "count": 2,
      "dataSize": 345,
      "messageType": "InspectorDuplicateProcess"
    },
    {
      "count": 3,
      "dataSize": 255,
      "messageType": "InspectorTimeEventMsg"
    },
    {
      "count": 4,
      "dataSize": 1082,
```

```
    "messageType": "InspectorNetworkInterface"
  },
  {
    "count": 2,
    "dataSize": 349,
    "messageType": "InspectorDnsEntry"
  },
  {
    "count": 11,
    "dataSize": 2514,
    "messageType": "InspectorDirectoryInfoMsg"
  },
  {
    "count": 1,
    "dataSize": 179,
    "messageType": "InspectorTcpV6ListeningPort"
  },
  {
    "count": 101,
    "dataSize": 10949,
    "messageType": "InspectorTerminal"
  },
  {
    "count": 26,
    "dataSize": 5916,
    "messageType": "InspectorUser"
  },
  {
    "count": 282,
    "dataSize": 32148,
    "messageType": "InspectorDynamicallyLoadedCodeModule"
  },
  {
    "count": 18,
    "dataSize": 10172,
    "messageType": "InspectorCreateProcess"
  },
  {
    "count": 3,
    "dataSize": 8001,
    "messageType": "InspectorProcessPerformance"
  },
  {
    "count": 1,
```

```
    "dataSize": 360,  
    "messageType": "InspectorOperatingSystem"  
  },  
  {  
    "count": 6,  
    "dataSize": 546,  
    "messageType": "InspectorStopProcess"  
  },  
  {  
    "count": 1,  
    "dataSize": 1553,  
    "messageType": "InspectorInstanceMetaData"  
  },  
  {  
    "count": 2,  
    "dataSize": 434,  
    "messageType": "InspectorTcpV4Connection"  
  },  
  {  
    "count": 474,  
    "dataSize": 2960322,  
    "messageType": "InspectorPackageInfo"  
  },  
  {  
    "count": 3,  
    "dataSize": 2235,  
    "messageType": "InspectorSystemPerformance"  
  },  
  {  
    "count": 105,  
    "dataSize": 46048,  
    "messageType": "InspectorCodeModule"  
  },  
  {  
    "count": 1,  
    "dataSize": 182,  
    "messageType": "InspectorUdpV6ListeningPort"  
  },  
  {  
    "count": 2,  
    "dataSize": 371,  
    "messageType": "InspectorUdpV4ListeningPort"  
  },  
  {
```

```
    "count": 18,  
    "dataSize": 8362,  
    "messageType": "InspectorKernelModule"  
  },  
  {  
    "count": 29,  
    "dataSize": 48788,  
    "messageType": "InspectorConfigurationInfo"  
  },  
  {  
    "count": 1,  
    "dataSize": 79,  
    "messageType": "InspectorMonitoringStart"  
  },  
  {  
    "count": 5,  
    "dataSize": 0,  
    "messageType": "InspectorSplitMsgBegin"  
  },  
  {  
    "count": 51,  
    "dataSize": 4593,  
    "messageType": "InspectorGroup"  
  },  
  {  
    "count": 1,  
    "dataSize": 184,  
    "messageType": "InspectorTcpV4ListeningPort"  
  },  
  {  
    "count": 1159,  
    "dataSize": 3146579,  
    "messageType": "Total"  
  },  
  {  
    "count": 5,  
    "dataSize": 0,  
    "messageType": "InspectorSplitMsgEnd"  
  },  
  {  
    "count": 1,  
    "dataSize": 612,  
    "messageType": "InspectorLoadImageInProgress"  
  }  
}
```

```
]
}
```

- Para API obtener más información, consulte [GetTelemetryMetadata](#) la Referencia de AWS CLI comandos.

## list-assessment-run-agents

En el siguiente ejemplo de código, se muestra cómo usar `list-assessment-run-agents`.

### AWS CLI

Para ver una lista de los agentes de evaluación, ejecute

El siguiente `list-assessment-run-agents` comando muestra los agentes de la evaluación ejecutada con los especificadosARN.

```
aws inspector list-assessment-run-agents \
  --assessment-run-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
  template/0-4r1V2mAw/run/0-MKkpXXPE
```

Salida:

```
{
  "assessmentRunAgents": [
    {
      "agentHealth": "HEALTHY",
      "agentHealthCode": "HEALTHY",
      "agentId": "i-49113b93",
      "assessmentRunArn": "arn:aws:inspector:us-
west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE",
      "telemetryMetadata": [
        {
          "count": 2,
          "dataSize": 345,
          "messageType": "InspectorDuplicateProcess"
        },
        {
          "count": 3,
          "dataSize": 255,
          "messageType": "InspectorTimeEventMsg"
        }
      ]
    }
  ]
}
```

```
{
  "count": 4,
  "dataSize": 1082,
  "messageType": "InspectorNetworkInterface"
},
{
  "count": 2,
  "dataSize": 349,
  "messageType": "InspectorDnsEntry"
},
{
  "count": 11,
  "dataSize": 2514,
  "messageType": "InspectorDirectoryInfoMsg"
},
{
  "count": 1,
  "dataSize": 179,
  "messageType": "InspectorTcpV6ListeningPort"
},
{
  "count": 101,
  "dataSize": 10949,
  "messageType": "InspectorTerminal"
},
{
  "count": 26,
  "dataSize": 5916,
  "messageType": "InspectorUser"
},
{
  "count": 282,
  "dataSize": 32148,
  "messageType": "InspectorDynamicallyLoadedCodeModule"
},
{
  "count": 18,
  "dataSize": 10172,
  "messageType": "InspectorCreateProcess"
},
{
  "count": 3,
  "dataSize": 8001,
  "messageType": "InspectorProcessPerformance"
}
```



```
    },  
    {  
      "count": 1,  
      "dataSize": 360,  
      "messageType": "InspectorOperatingSystem"  
    },  
    {  
      "count": 6,  
      "dataSize": 546,  
      "messageType": "InspectorStopProcess"  
    },  
    {  
      "count": 1,  
      "dataSize": 1553,  
      "messageType": "InspectorInstanceMetaData"  
    },  
    {  
      "count": 2,  
      "dataSize": 434,  
      "messageType": "InspectorTcpV4Connection"  
    },  
    {  
      "count": 474,  
      "dataSize": 2960322,  
      "messageType": "InspectorPackageInfo"  
    },  
    {  
      "count": 3,  
      "dataSize": 2235,  
      "messageType": "InspectorSystemPerformance"  
    },  
    {  
      "count": 105,  
      "dataSize": 46048,  
      "messageType": "InspectorCodeModule"  
    },  
    {  
      "count": 1,  
      "dataSize": 182,  
      "messageType": "InspectorUdpV6ListeningPort"  
    },  
    {  
      "count": 2,  
      "dataSize": 371,
```

```
    "messageType": "InspectorUdpV4ListeningPort"
  },
  {
    "count": 18,
    "dataSize": 8362,
    "messageType": "InspectorKernelModule"
  },
  {
    "count": 29,
    "dataSize": 48788,
    "messageType": "InspectorConfigurationInfo"
  },
  {
    "count": 1,
    "dataSize": 79,
    "messageType": "InspectorMonitoringStart"
  },
  {
    "count": 5,
    "dataSize": 0,
    "messageType": "InspectorSplitMsgBegin"
  },
  {
    "count": 51,
    "dataSize": 4593,
    "messageType": "InspectorGroup"
  },
  {
    "count": 1,
    "dataSize": 184,
    "messageType": "InspectorTcpV4ListeningPort"
  },
  {
    "count": 1159,
    "dataSize": 3146579,
    "messageType": "Total"
  },
  {
    "count": 5,
    "dataSize": 0,
    "messageType": "InspectorSplitMsgEnd"
  },
  {
    "count": 1,
```

```

        "dataSize": 612,
        "messageType": "InspectorLoadImageInProgress"
      }
    ]
  }
}

```

Para obtener más información, consulte [AWS Agentes](#) en la Guía del usuario de Amazon Inspector.

- Para API obtener más información, consulte [ListAssessmentRunAgents](#) la Referencia de AWS CLI comandos.

## list-assessment-runs

En el siguiente ejemplo de código, se muestra cómo usar `list-assessment-runs`.

### AWS CLI

Para enumerar las ejecuciones de evaluación

El siguiente `list-assessment-runs` comando muestra todas las ejecuciones de evaluación existentes.

```
aws inspector list-assessment-runs
```

Salida:

```

{
  "assessmentRunArns": [
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-MKkpXXPE",
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-v5D6fI3v"
  ]
}

```

Para obtener más información, consulte las [plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector](#) en la Guía del usuario de Amazon Inspector.

- Para API obtener más información, consulte [ListAssessmentRuns](#) la Referencia de AWS CLI comandos.

## **list-assessment-targets**

En el siguiente ejemplo de código, se muestra cómo usar `list-assessment-targets`.

AWS CLI

Para enumerar los objetivos de la evaluación

El siguiente `list-assessment-targets` comando muestra todos los objetivos de evaluación existentes:

```
aws inspector list-assessment-targets
```

Salida:

```
{
  "assessmentTargetArns": [
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq"
  ]
}
```

Para obtener más información, consulte los objetivos de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [ListAssessmentTargets](#) la Referencia de AWS CLI comandos.

## **list-assessment-templates**

En el siguiente ejemplo de código, se muestra cómo usar `list-assessment-templates`.

AWS CLI

Para enumerar las plantillas de evaluación

El siguiente `list-assessment-templates` comando muestra todas las plantillas de evaluación existentes:

```
aws inspector list-assessment-templates
```

Salida:

```
{
  "assessmentTemplateArns": [
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw",
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-
Uza6ihLh"
  ]
}
```

Para obtener más información, consulte las plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [ListAssessmentTemplates](#) la Referencia de AWS CLI comandos.

## list-event-subscriptions

En el siguiente ejemplo de código, se muestra cómo usar `list-event-subscriptions`.

AWS CLI

Para ver una lista de las suscripciones a eventos

El siguiente `list-event-subscriptions` comando muestra todas las suscripciones a eventos de la plantilla de evaluación con el signo ARN de `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0`:

```
aws inspector list-event-subscriptions --resource-arn arn:aws:inspector:us-
west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0
```

Salida:

```
{
  "subscriptions": [
    {
```

```

    "eventSubscriptions": [
      {
        "event": "ASSESSMENT_RUN_COMPLETED",
        "subscribedAt": 1459455440.867
      }
    ],
    "resourceArn": "arn:aws:inspector:us-west-2:123456789012:target/0-
nvgVhaxX/template/0-7sbz2Kz0",
    "topicArn": "arn:aws:sns:us-west-2:123456789012:exampletopic"
  }
]
}

```

Para obtener más información, consulte las plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [ListEventSubscriptions](#) la Referencia de AWS CLI comandos.

## list-findings

En el siguiente ejemplo de código, se muestra cómo usar `list-findings`.

### AWS CLI

Para enumerar los hallazgos

El siguiente `list-findings` comando muestra todos los hallazgos generados:

```
aws inspector list-findings
```

Salida:

```

{
  "findingArns": [
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-MKkpXXPE/finding/0-HwPnsDm4",
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-v5D6fI3v/finding/0-tyvmqBLy"
  ]
}

```

Para obtener más información, consulta los hallazgos de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [ListFindings](#) la Referencia de AWS CLI comandos.

## **list-rules-packages**

En el siguiente ejemplo de código, se muestra cómo usar `list-rules-packages`.

### AWS CLI

Para enumerar los paquetes de reglas

El siguiente `list-rules-packages` comando muestra todos los paquetes de reglas del Inspector disponibles:

```
aws inspector list-rules-packages
```

Salida:

```
{
  "rulesPackageArns": [
    "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p",
    "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-H5hpSawc",
    "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-JJ0tZiqQ",
    "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-vg5GGHSD"
  ]
}
```

Para obtener más información, consulta Amazon Inspector Rules Packages and Rules en la guía Amazon Inspector.

- Para API obtener más información, consulte [ListRulesPackages](#) la Referencia de AWS CLI comandos.

## **list-tags-for-resource**

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas del recurso

El siguiente `list-tags-for-resource` comando muestra todas las etiquetas asociadas a la plantilla de evaluación con el signo ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-gcwFliYu`:

```
aws inspector list-tags-for-resource --resource-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-gcwFliYu
```

Salida:

```
{
  "tags": [
    {
      "key": "Name",
      "value": "Example"
    }
  ]
}
```

Para obtener más información, consulte las plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## preview-agents

En el siguiente ejemplo de código, se muestra cómo usar `preview-agents`.

AWS CLI

Para previsualizar los agentes

El siguiente `preview-agents` comando muestra una vista previa de los agentes instalados en las EC2 instancias que forman parte del objetivo de la evaluación con la `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq` siguiente ARN información:

```
aws inspector preview-agents --preview-agents-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq
```

Salida:



```
{
  "agentPreviews": [
    {
      "agentId": "i-49113b93"
    }
  ]
}
```

Para obtener más información, consulte los objetivos de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [PreviewAgents](#) la Referencia de AWS CLI comandos.

## register-cross-account-access-role

En el siguiente ejemplo de código, se muestra cómo usar `register-cross-account-access-role`.

### AWS CLI

Para registrar el rol de acceso entre cuentas

El siguiente `register-cross-account-access-role` comando registra la IAM función con ARN la `arn:aws:iam::123456789012:role/inspector` que Amazon Inspector utiliza para enumerar EC2 las instancias al inicio de la ejecución de la evaluación o cuando se llama al comando `preview-agents`:

```
aws inspector register-cross-account-access-role --role-arn arn:aws:iam::123456789012:role/inspector
```

Para obtener más información, consulta [Cómo configurar Amazon Inspector](#) en la guía Amazon Inspector.

- Para API obtener más información, consulte [RegisterCrossAccountAccessRole](#) la Referencia de AWS CLI comandos.

## remove-attributes-from-findings

En el siguiente ejemplo de código, se muestra cómo usar `remove-attributes-from-findings`.

## AWS CLI

Para eliminar atributos de los hallazgos

El siguiente `remove-attributes-from-finding` comando elimina el atributo con la clave de `Example` y el valor de `example` de de la búsqueda con el ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-811VIE0D/run/0-Z02cjjug/finding/0-T8yM9mEU`:

```
aws inspector remove-attributes-from-findings --finding-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-811VIE0D/run/0-Z02cjjug/finding/0-T8yM9mEU --attribute-keys key=Example,value=example
```

Salida:

```
{
  "failedItems": {}
}
```

Para obtener más información, consulta los hallazgos de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [RemoveAttributesFromFindings](#) la Referencia de AWS CLI comandos.

## set-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `set-tags-for-resource`.

## AWS CLI

Para establecer etiquetas para un recurso

El siguiente `set-tags-for-resource` comando establece la etiqueta con la clave de `Example` y el valor de `example` en la plantilla de evaluación con la ARN `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0`:

```
aws inspector set-tags-for-resource --resource-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0 --tags key=Example,value=example
```

Para obtener más información, consulte las plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [SetTagsForResource](#) la Referencia de AWS CLI comandos.

## start-assessment-run

En el siguiente ejemplo de código, se muestra cómo usar `start-assessment-run`.

### AWS CLI

Para iniciar una evaluación, ejecute

El siguiente `start-assessment-run` comando inicia la ejecución de evaluación denominada `examplerun` mediante la plantilla de evaluación con el signo ARN `dearn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T`:

```
aws inspector start-assessment-run --assessment-run-name examplerun --assessment-template-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T
```

Salida:

```
{
  "assessmentRunArn": "arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-j0o0oxyY"
}
```

Para obtener más información, consulte las plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [StartAssessmentRun](#) la Referencia de AWS CLI comandos.

## stop-assessment-run

En el siguiente ejemplo de código, se muestra cómo usar `stop-assessment-run`.

## AWS CLI

Para detener una ejecución de evaluación

El siguiente `stop-assessment-run` comando detiene la ejecución de la evaluación con el comando ARN `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-j0oroxyY`:

```
aws inspector stop-assessment-run --assessment-run-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-j0oroxyY
```

Para obtener más información, consulte las plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [StopAssessmentRun](#) la Referencia de AWS CLI comandos.

## subscribe-to-event

En el siguiente ejemplo de código, se muestra cómo usar `subscribe-to-event`.

### AWS CLI

Para suscribirse a un evento

El siguiente ejemplo habilita el proceso de enviar SNS notificaciones de Amazon sobre el `ASSESSMENT_RUN_COMPLETED` evento al tema con ARN `arn:aws:sns:us-west-2:123456789012:exampletopic`

```
aws inspector subscribe-to-event \  
  --event ASSESSMENT_RUN_COMPLETED \  
  --resource-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0 \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:exampletopic
```

Este comando no genera ninguna salida.

Para obtener más información, consulte las [plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector](#) en la guía de Amazon Inspector.

- Para API obtener más información, consulte [SubscribeToEvent](#) la Referencia de AWS CLI comandos.

## unsubscribe-from-event

En el siguiente ejemplo de código, se muestra cómo usar `unsubscribe-from-event`.

### AWS CLI

Para cancelar la suscripción a un evento

El siguiente `unsubscribe-from-event` comando deshabilita el proceso de envío de SNS notificaciones de Amazon sobre el `ASSESSMENT_RUN_COMPLETED` evento al tema con el signo ARN `dearn:aws:sns:us-west-2:123456789012:exampletopic`:

```
aws inspector unsubscribe-from-event --event ASSESSMENT_RUN_COMPLETED --resource-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0 --topic arn:aws:sns:us-west-2:123456789012:exampletopic
```

Para obtener más información, consulte las plantillas de evaluación y las ejecuciones de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [UnsubscribeFromEvent](#) la Referencia de AWS CLI comandos.

## update-assessment-target

En el siguiente ejemplo de código, se muestra cómo usar `update-assessment-target`.

### AWS CLI

Para actualizar un objetivo de evaluación

El siguiente `update-assessment-target` comando actualiza el objetivo ARN de `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX` la evaluación con el nombre de `Example` y el grupo de recursos con el ARN `dearn:aws:inspector:us-west-2:123456789012:resourcegroup/0-yNbgL5Pt`:

```
aws inspector update-assessment-target --assessment-target-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX --assessment-target-name Example --resource-group-arn arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-yNbgL5Pt
```

Para obtener más información, consulte los objetivos de evaluación de Amazon Inspector en la guía de Amazon Inspector.

- Para API obtener más información, consulte [UpdateAssessmentTarget](#) la Referencia de AWS CLI comandos.

## AWS IoT ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS IoT.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **accept-certificate-transfer**

En el siguiente ejemplo de código, se muestra cómo usar `accept-certificate-transfer`.

AWS CLI

Para aceptar un certificado de dispositivo transferido desde una AWS cuenta diferente

En el siguiente `accept-certificate-transfer` ejemplo, se acepta un certificado de dispositivo transferido desde otra AWS cuenta. El certificado se identifica por su ID.

```
aws iot accept-certificate-transfer \  
  --certificate-  
  id 488b6a7f2acdeb00a77384e63c4e40b18bEXAMPLEe57b7272ba44c45e3448142
```

Este comando no produce ninguna salida.

Para obtener más información, consulte [Transferir un certificado a otra cuenta](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [AcceptCertificateTransfer](#) la Referencia de AWS CLI comandos.

## add-thing-to-billing-group

En el siguiente ejemplo de código, se muestra cómo usar `add-thing-to-billing-group`.

### AWS CLI

Ejemplo 1: Para añadir algo por su nombre a un grupo de facturación

En el siguiente `add-thing-to-billing-group` ejemplo, se agrega el elemento denominado `MyLightBulb` al grupo de facturación denominado `GroupOne`.

```
aws iot add-thing-to-billing-group \  
  --billing-group-name GroupOne \  
  --thing-name MyLightBulb
```

Este comando no genera ninguna salida.

Ejemplo 2: Para añadir una cosa ARN a un grupo de facturación

En el siguiente `add-thing-to-billing-group` ejemplo, se agrega un elemento con un valor especificado ARN a un grupo de facturación con el valor especificado `ARN`. Especificar un ARN es útil si trabaja con varias AWS regiones o cuentas. Te ayudará a asegurarte de que estás añadiendo la región y la cuenta correctas.

```
aws iot add-thing-to-thing-group \  
  --billing-group-arn "arn:aws:iot:us-west-2:123456789012:billinggroup/GroupOne" \  
  --thing-arn "arn:aws:iot:us-west-2:123456789012:thing/MyOtherLightBulb"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Grupos de facturación](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [AddThingToBillingGroup](#) la Referencia de AWS CLI comandos.

## add-thing-to-thing-group

En el siguiente ejemplo de código, se muestra cómo usar `add-thing-to-thing-group`.

### AWS CLI

Para añadir algo a un grupo

En el siguiente `add-thing-to-thing-group` ejemplo, se agrega el elemento especificado al grupo de elementos especificado.

```
aws iot add-thing-to-thing-group \  
  --thing-name MyLightBulb \  
  --thing-group-name LightBulbs
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [AddThingToThingGroup](#) la Referencia de AWS CLI comandos.

## associate-targets-with-job

En el siguiente ejemplo de código, se muestra cómo usar `associate-targets-with-job`.

### AWS CLI

Para asociar un grupo de cosas a una tarea continua

En el siguiente `associate-targets-with-job` ejemplo, se asocia el grupo de cosas especificado al trabajo continuo especificado.

```
aws iot associate-targets-with-job \  
  --targets "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs" \  
  --job-id "example-job-04"
```

Salida:

```
{  
  "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-04",
```



```
"jobId": "example-job-04",  
"description": "example continuous job"  
}
```

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [AssociateTargetsWithJob](#) la Referencia de AWS CLI comandos.

## attach-policy

En el siguiente ejemplo de código, se muestra cómo usar `attach-policy`.

### AWS CLI

Ejemplo 1: Para adjuntar una política a un grupo de cosas

En el siguiente `attach-policy` ejemplo, se adjunta la política especificada a un grupo de cosas identificado por su ARN.

```
aws iot attach-policy \  
  --target "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs" \  
  --policy-name "UpdateDeviceCertPolicy"
```

Este comando no produce ninguna salida.

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

Ejemplo 2: Para adjuntar una política a un certificado

En el siguiente `attach-policy` ejemplo, se adjunta la política `UpdateDeviceCertPolicy` al principal especificado en un certificado.

```
aws iot attach-policy \  
  --policy-name UpdateDeviceCertPolicy \  
  --target "arn:aws:iot:us-west-2:123456789012:cert/4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e"
```

Este comando no produce ninguna salida.

Para obtener más información, consulte [Adjuntar una política de AWS IoT a un certificado de dispositivo](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [AttachPolicy](#) la Referencia de AWS CLI comandos.

## attach-security-profile

En el siguiente ejemplo de código, se muestra cómo usar `attach-security-profile`.

### AWS CLI

Para asociar un perfil de seguridad a todos los dispositivos no registrados

El siguiente `attach-security-profile` ejemplo asocia el perfil de seguridad de AWS IoT Device Defender nombrado `Testprofile` con todos los dispositivos no registrados de la `us-west-2` región de esta AWS cuenta.

```
aws iot attach-security-profile \  
  --security-profile-name Testprofile \  
  --security-profile-target-arn "arn:aws:iot:us-west-2:123456789012:all/  
unregistered-things"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [AttachSecurityProfile](#) la Referencia de AWS CLI comandos.

## attach-thing-principal

En el siguiente ejemplo de código, se muestra cómo usar `attach-thing-principal`.

### AWS CLI

Para adjuntar un certificado a tu cosa

En el siguiente `attach-thing-principal` ejemplo, se adjunta un certificado a la `MyTemperatureSensor` cosa. El certificado se identifica mediante un ARN. Puede encontrar el certificado ARN para obtener un certificado en la consola de AWS IoT.

```
aws iot attach-thing-principal \  
  --thing-name MyTemperatureSensor \  
  --principal arn:aws:iot:us-  
west-2:123456789012:cert/2e1eb273792174ec2b9bf4e9b37e6c6c692345499506002a35159767055278e8
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de objetos con el registro](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [AttachThingPrincipal](#) la Referencia de AWS CLI comandos.

## cancel-audit-mitigation-actions-task

En el siguiente ejemplo de código, se muestra cómo usar `cancel-audit-mitigation-actions-task`.

### AWS CLI

Para cancelar una tarea de auditoría (acciones de mitigación)

El siguiente `cancel-audit-mitigations-action-task` ejemplo cancela la aplicación de las acciones de mitigación para la tarea especificada. No puede cancelar las tareas que ya se han completado.

```
aws iot cancel-audit-mitigation-actions-task  
  --task-id "myActionsTaskId"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [CancelAuditMitigationActionsTask \(Comandos de acción de mitigación\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CancelAuditMitigationActionsTask](#) la Referencia de AWS CLI comandos.

## cancel-audit-task

En el siguiente ejemplo de código, se muestra cómo usar `cancel-audit-task`.

## AWS CLI

Para cancelar una tarea de auditoría

El siguiente `cancel-audit-task` ejemplo cancela una tarea de auditoría con el identificador de tarea especificado. No puede cancelar una tarea que esté completa.

```
aws iot cancel-audit-task \  
  --task-id a3aea009955e501a31b764abe1bebd3d
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CancelAuditTask](#) la Referencia de AWS CLI comandos.

## `cancel-certificate-transfer`

En el siguiente ejemplo de código, se muestra cómo usar `cancel-certificate-transfer`.

## AWS CLI

Para cancelar la transferencia de un certificado a una AWS cuenta diferente

En el siguiente `cancel-certificate-transfer` ejemplo, se cancela la transferencia de la transferencia de certificado especificada. El certificado se identifica mediante un identificador de certificado. Puedes encontrar el ID de un certificado en la consola de AWS IoT.

```
aws iot cancel-certificate-transfer \  
  --certificate-  
  id f0f33678c7c9a046e5cc87b2b1a58dfa0beec26db78add5e605d630e05c7fc8
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Transferir un certificado a otra cuenta](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [CancelCertificateTransfer](#) la Referencia de AWS CLI comandos.

## cancel-job-execution

En el siguiente ejemplo de código, se muestra cómo usar `cancel-job-execution`.

### AWS CLI

Para cancelar la ejecución de un trabajo en un dispositivo

El siguiente `cancel-job-execution` ejemplo cancela la ejecución del trabajo especificado en un dispositivo. Si el trabajo no está en ese QUEUED estado, debe añadir el `--force` parámetro.

```
aws iot cancel-job-execution \  
  --job-id "example-job-03" \  
  --thing-name "MyRPi"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CancelJobExecution](#) la Referencia de AWS CLI comandos.

## cancel-job

En el siguiente ejemplo de código, se muestra cómo usar `cancel-job`.

### AWS CLI

Para cancelar un trabajo

En el `cancel-job` ejemplo siguiente se cancela el trabajo especificado.

```
aws iot cancel-job \  
  --job-id "example-job-03"
```

Salida:

```
{  
  "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-03",  
  "jobId": "example-job-03",  
  "description": "example job test"
```

```
}
```

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CancelJob](#) la Referencia de AWS CLI comandos.

## **clear-default-authorizer**

En el siguiente ejemplo de código, se muestra cómo usar `clear-default-authorizer`.

### AWS CLI

Para borrar el autorizador predeterminado

El siguiente `clear-default-authorizer` ejemplo borra el autorizador personalizado predeterminado configurado actualmente. Tras ejecutar este comando, no hay ningún autorizador predeterminado. Al utilizar un autorizador personalizado, debe especificarlo por su nombre en los encabezados de las HTTP solicitudes.

```
aws iot clear-default-authorizer
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [ClearDefaultAuthorizer](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [ClearDefaultAuthorizer](#) la Referencia de AWS CLI comandos.

## **confirm-topic-rule-destination**

En el siguiente ejemplo de código, se muestra cómo usar `confirm-topic-rule-destination`.

### AWS CLI

Para confirmar el destino de una regla de tema

El siguiente `confirm-topic-rule-destination` ejemplo confirma el destino de una regla temática con un token de confirmación recibido en un HTTP punto final.

```
aws iot confirm-topic-rule-destination \
```

```
--confirmation-token "AYADeIcmtq-
ZkxfpiWIQqHWM5ucAXwABABVhd3MtY3J5cHRvLXB1YmXpYy1rZXkAREFyY1E0UmLGeDg0V21BZWZ1VjZtZWFRVUJJUkt
aywpPqg8YEsa1LD4B40aJ2s1wEHKMybiF1Ro0ZzYisI0IvsLzQY5UmCkq3tV-3f7-
nKfosgIAAAAAADAAAEEAAAAAAAAAAAAAAAAAAAAAai9RMgy-
V19V9m6Iw2xfbw_____wAAAAEAAAAAAAAAAAAAAAAAAEAAAB1hw4SokgUcxiJ3gT06n50NLJVpzyQR1UmPIj5sShqXEQGc0
iufgrzTePl8RZY0Wr006Aj9DiVzJZx-1iD6Pu-
G6PUw1ka07Knzs2B4AD0qfrHUF4pYRTvyUgBnMGUCMQC8ZRmhKqntd_c6Kgrow3bMUDbvNqo2qZr8Z8Jm2rzgseR01LA
PIetJ803Z4ILILF8xXlcdPGP-PV1d0XFemyL8g"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Confirmación del destino de una regla temática](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ConfirmTopicRuleDestination](#) la Referencia de AWS CLI comandos.

## create-audit-suppression

En el siguiente ejemplo de código, se muestra cómo usar `create-audit-suppression`.

### AWS CLI

Para crear una supresión de los resultados de una auditoría

En el siguiente `create-audit-suppression` ejemplo, se suprime un hallazgo de auditoría para una política denominada `virtualMachinePolicy` «» que se ha marcado por ser demasiado permisiva.

```
aws iot create-audit-suppression \
  --check-name IOT_POLICY_OVERLY_PERMISSIVE_CHECK \
  --resource-identifier
policyVersionIdentifier={"policyName"="virtualMachinePolicy","policyVersionId"="1"}
\
  --no-suppress-indefinitely \
  --expiration-date 2020-10-20
```

Este comando no genera ninguna salida.

Para obtener más información, consulte las [supresiones de detección de auditorías](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte la Referencia [CreateAuditSuppression](#) de AWS CLI comandos.

## create-authorizer

En el siguiente ejemplo de código, se muestra cómo usar `create-authorizer`.

### AWS CLI

Para crear un autorizador personalizado

En el siguiente `create-authorizer` ejemplo, se crea un autorizador personalizado que utiliza la función Lambda especificada como parte de un servicio de autenticación personalizado.

```
aws iot create-authorizer \
  --authorizer-name "CustomAuthorizer" \
  --authorizer-function-arn "arn:aws:lambda:us-
west-2:123456789012:function:CustomAuthorizerFunction" \
  --token-key-name "MyAuthToken" \
  --status ACTIVE \
  --token-signing-public-keys FIRST_KEY="-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEA1uJ0B4lQPgG/1M6ZfIwo
Z+7ENxAio9q6QD4FFqjGZsvjtYwjoe1RKK0U8Eq9xb503kRSmYIwTzwm/f4Gf0Y
ZUloJ+t3PUUwHrmbYTAgrCUgRFygjfgVwGCPs5ZAX4Eyqt5cr+AIHIiUDbxSa7p
zw0BKPeic0asNJpqT8PkBbRaKyleJh5oo81NDHHmVtbBm5A5YiJjqYXLaVAowKzZ
+GqsNvAQ9Jy1wI2VrEa10fL8f1DB/BJLm7zjpfPOHDJQgID0XnZwAlNnZc0hCwIx
50g2LW20y9R/dmqtDmJiVP97Z4GykxPvw1YHrUXY0iW1R3AR/Ac1NhCTGZMwVDB1
1QIDAQAB
-----END PUBLIC KEY-----"
```

Salida:

```
{
  "authorizerName": "CustomAuthorizer",
  "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer2"
}
```

Para obtener más información, consulte [CreateAuthorizer](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [CreateAuthorizer](#) la Referencia de AWS CLI comandos.



## create-billing-group

En el siguiente ejemplo de código, se muestra cómo usar `create-billing-group`.

### AWS CLI

Para crear un grupo de facturación

En el siguiente `create-billing-group` ejemplo, se crea un grupo de facturación sencillo denominado `GroupOne`.

```
aws iot create-billing-group \  
  --billing-group-name GroupOne
```

Salida:

```
{  
  "billingGroupName": "GroupOne",  
  "billingGroupArn": "arn:aws:iot:us-west-2:123456789012:billinggroup/GroupOne",  
  "billingGroupId": "103de383-114b-4f51-8266-18f209ef5562"  
}
```

Para obtener más información, consulte [Grupos de facturación](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateBillingGroup](#) la Referencia de AWS CLI comandos.

## create-certificate-from-csr

En el siguiente ejemplo de código, se muestra cómo usar `create-certificate-from-csr`.

### AWS CLI

Para crear un certificado de dispositivo a partir de una solicitud de firma de certificado (CSR)

En el siguiente `create-certificate-from-csr` ejemplo, se crea un certificado de dispositivo a partir de un CSR. Puede usar el `openssl` comando para crear un CSR.

```
aws iot create-certificate-from-csr \  
  \
```

```
--certificate-signing-request=file://certificate.csr
```

Salida:

```
{
  "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/
c0c57bbc8baaf4631a9a0345c957657f5e710473e3ddbbee1428d216d54d53ac9",
  "certificateId":
"c0c57bbc8baaf4631a9a0345c957657f5e710473e3ddbbee1428d216d54d53ac9",
  "certificatePem": "<certificate-text>"
}
```

Para obtener más información, consulte [CreateCertificateFromCSR](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [CreateCertificateFromCsrl](#) la Referencia de AWS CLI comandos.

## create-custom-metric

En el siguiente ejemplo de código, se muestra cómo usar `create-custom-metric`.

AWS CLI

Para crear una métrica personalizada publicada por sus dispositivos en Device Defender

En el siguiente `create-custom-metric` ejemplo, se crea una métrica personalizada que mide el porcentaje de batería.

```
aws iot create-custom-metric \
  --metric-name "batteryPercentage" \
  --metric-type "number" \
  --display-name "Remaining battery percentage." \
  --region us-east-1 \
  --client-request-token "02ccb92b-33e8-4dfa-a0c1-35b181ed26b0"
```

Salida:

```
{
  "metricName": "batteryPercentage",
  "metricArn": "arn:aws:iot:us-east-1:1234564789012:custommetric/
batteryPercentage"
```

```
}
```

Para obtener más información, consulte [Métricas personalizadas](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [CreateCustomMetric](#) la Referencia de AWS CLI comandos.

## create-dimension

En el siguiente ejemplo de código, se muestra cómo usar `create-dimension`.

### AWS CLI

Para crear una dimensión

A continuación, se `create-dimension` crea una dimensión con un único filtro de tema denominado `TopicFilterForAuthMessages`.

```
aws iot create-dimension \  
  --name TopicFilterForAuthMessages \  
  --type TOPIC_FILTER \  
  --string-values device/+/auth
```

Salida:

```
{  
  "name": "TopicFilterForAuthMessages",  
  "arn": "arn:aws:iot:eu-west-2:123456789012:dimension/TopicFilterForAuthMessages"  
}
```

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateDimension](#) la Referencia de AWS CLI comandos.

## create-domain-configuration

En el siguiente ejemplo de código, se muestra cómo usar `create-domain-configuration`.

## AWS CLI

Para crear una configuración de dominio

El siguiente `create-domain-configuration` ejemplo crea una configuración AWS de dominio administrado con un tipo de servicio de DATA.

```
aws iot create-domain-configuration \  
  --domain-configuration-name "additionalDataDomain" \  
  --service-type "DATA"
```

Salida:

```
{  
  "domainConfigurationName": "additionalDataDomain",  
  "domainConfigurationArn": "arn:aws:iot:us-  
west-2:123456789012:domainconfiguration/additionalDataDomain/dikMh"  
}
```

Para obtener más información, consulte Terminales [configurables en](#) la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateDomainConfiguration](#) la Referencia de AWS CLI comandos.

## `create-dynamic-thing-group`

En el siguiente ejemplo de código, se muestra cómo usar `create-dynamic-thing-group`.

## AWS CLI

Para crear un grupo de cosas dinámico

El siguiente `create-dynamic-thing-group` ejemplo crea un grupo de cosas dinámico que contiene cualquier cosa con un atributo de temperatura superior a 60 grados. Debe habilitar la indexación de flotas de AWS IoT antes de poder usar grupos de cosas dinámicos.

```
aws iot create-dynamic-thing-group \  
  --thing-group-name "RoomTooWarm" \  
  --query-string "attributes.temperature>60"
```

Salida:

```
{
  "thingGroupName": "RoomTooWarm",
  "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RoomTooWarm",
  "thingGroupId": "9d52492a-fc87-43f4-b6e2-e571d2ffcad1",
  "indexName": "AWS_Things",
  "queryString": "attributes.temperature>60",
  "queryVersion": "2017-09-30"
}
```

Para obtener más información, consulte [Grupos de cosas dinámicos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateDynamicThingGroup](#) la Referencia de AWS CLI comandos.

## create-job

En el siguiente ejemplo de código, se muestra cómo usar `create-job`.

### AWS CLI

Ejemplo 1: Para crear un trabajo

El siguiente `create-job` ejemplo crea un trabajo de AWS IoT sencillo que envía un JSON documento al MyRaspberryPi dispositivo.

```
aws iot create-job \
  --job-id "example-job-01" \
  --targets "arn:aws:iot:us-west-2:123456789012:thing/MyRaspberryPi" \
  --document file://example-job.json \
  --description "example job test" \
  --target-selection SNAPSHOT
```

Salida:

```
{
  "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-01",
  "jobId": "example-job-01",
  "description": "example job test"
}
```

```
}
```

## Ejemplo 2: Para crear un trabajo continuo

El siguiente `create-job` ejemplo crea un trabajo que continúa ejecutándose después de que las cosas especificadas como objetivos hayan completado el trabajo. En este ejemplo, el objetivo es un grupo de cosas, por lo que cuando se añaden nuevos dispositivos al grupo, el trabajo continuo se ejecuta en esas cosas nuevas.

```
aws iot create-job --job-id «example-job-04" --targets «arn:aws:iot:us-west-2:123456789012:thinggroup/» --document file: //example-job.json --description «ejemplo de trabajo continuo» --target-selection DeadBulbs CONTINUOUS
```

Salida:

```
{
  "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-04",
  "jobId": "example-job-04",
  "description": "example continuous job"
}
```

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateJob](#) la Referencia de AWS CLI comandos.

## **create-keys-and-certificate**

En el siguiente ejemplo de código, se muestra cómo usar `create-keys-and-certificate`.

AWS CLI

Para crear un RSA key pair y emitir un certificado X.509

A continuación, se `create-keys-and-certificate` crea un par de RSA claves de 2048 bits y se emite un certificado X.509 con la clave pública emitida. Como esta es la única vez que el AWS IoT proporciona la clave privada para este certificado, asegúrese de guardarla en un lugar seguro.

```
aws iot create-keys-and-certificate \
  --certificate-pem-outfile "myTest.cert.pem" \
```

```
--public-key-outfile "myTest.public.key" \
--private-key-outfile "myTest.private.key"
```

Salida:

```
{
  "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/9894ba17925e663f1d29c23af4582b8e3b7619c31f3fbd93adcb51ae54b83dc2",
  "certificateId":
  "9894ba17925e663f1d29c23af4582b8e3b7619c31f3fbd93adcb51ae54b83dc2",
  "certificatePem": "
-----BEGIN CERTIFICATE-----
MIICiTCCEXAMPLE6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMaKGA1UEBhMC
VVMxCzAJBgNVBAGEXAMPLEAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSEXAMPLE2x1MRIwEAYDVQQDEwLUZXN0Q21sYWxhZAd
BgkqhkiG9w0BCQEWEG5vb251QGftYEXAMPLEb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMaKGA1UEBhMCEXAMPLEJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDAEXAMPLEsTC0lBTSBDb25z
b2x1MRIwEAYDVQQDEwLUZXN0Q21sYWxhZAdBgkqhkiG9w0BCQEXAMPLE251QGft
YXpvbi5jb20wZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+aEXAMPLE
EXAMPLEfEvYsWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZEXAMPLEELG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZncvQAEXAMPLEWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUHVvXyUntneD9+h8Mg9qEXAMPLEEyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDEXAMPLEBHjJnyp3780D8uTs7fLvJx79LjStB
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----\n",
  "keyPair": {
    "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQ0AQ8AMIIBCGKCAQEAEXAMPLE1nnyJwKSMHw4h\nMMEXAMPLEuuN/
dMAS3fyce8DW/4+EXAMPLEYjmoF/YVF/gHr99VEEXAMPLE5VF13\n59VK7cEXAMPLE67GK+y+jikqX0gHh/
xJTwo
+sGpWEXAMPLEDz18x0d2ka4tCzuWEXAMPLEEahJbYkCPUBSU8opVkr7qkEXAMPLE1DR6sx2Hocli00Ltu6Fkw91swQWEX
\GB3ZPrNh0PzQYvjUSTzEccyNCx2EXAMPLEvp9mQ0UXP6p1fgxwKRX2fEXAMPLEDa
\nhJLXkX3rHU2xbxJSq7D+XEXAMPLEcw+LyFhI5mgFR188eGdsAEXAMPLElnI9EesG\nnFQIDAQAB\n-----
END PUBLIC KEY-----\n",
    "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----\nkey omitted for security
reasons\n-----END RSA PRIVATE KEY-----\n"
  }
}
```

Para obtener más información, consulte [Crear y registrar un certificado de dispositivo de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte la Referencia [CreateKeysAndCertificate](#) de AWS CLI comandos.

## create-mitigation-action

En el siguiente ejemplo de código, se muestra cómo usar `create-mitigation-action`.

### AWS CLI

Para crear una acción de mitigación

El siguiente `create-mitigation-action` ejemplo define una acción de mitigación denominada `AddThingsToQuarantineGroup1Action` que, cuando se aplica, mueve las cosas al grupo de cosas denominado `QuarantineGroup1`. Esta acción anula los grupos de cosas dinámicos.

```
aws iot create-mitigation-action --cli-input-json file::params.json
```

Contenidos de `params.json`:

```
{
  "actionName": "AddThingsToQuarantineGroup1Action",
  "actionParams": {
    "addThingsToThingGroupParams": {
      "thingGroupNames": [
        "QuarantineGroup1"
      ],
      "overrideDynamicGroups": true
    }
  },
  "roleArn": "arn:aws:iam::123456789012:role/service-role/MoveThingsToQuarantineGroupRole"
}
```

Salida:

```
{
  "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/AddThingsToQuarantineGroup1Action",
  "actionId": "992e9a63-a899-439a-aa50-4e20c52367e1"
}
```



Para obtener más información, consulte [CreateMitigationAction \(Comandos de acción de mitigación\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateMitigationAction](#) la Referencia de AWS CLI comandos.

## create-ota-update

En el siguiente ejemplo de código, se muestra cómo usar `create-ota-update`.

### AWS CLI

Para crear una OTA actualización para usarla con Amazon Free RTOS

El siguiente `create-ota-update` ejemplo crea una AWS IoT OTAUpdate en un grupo de cosas o grupos objetivo. Esto forma parte de una RTOS over-the-air actualización gratuita de Amazon que permite implementar nuevas imágenes de firmware en un solo dispositivo o en un grupo de dispositivos.

```
aws iot create-ota-update \  
  --cli-input-json file://create-ota-update.json
```

Contenidos de `create-ota-update.json`:

```
{  
  "otaUpdateId": "ota12345",  
  "description": "A critical update needed right away.",  
  "targets": [  
    "device1",  
    "device2",  
    "device3",  
    "device4"  
  ],  
  "targetSelection": "SNAPSHOT",  
  "awsJobExecutionsRolloutConfig": {  
    "maximumPerMinute": 10  
  },  
  "files": [  
    {  
      "fileName": "firmware.bin",  
      "fileLocation": {  
        "stream": {
```

```

        "streamId": "004",
        "fileId":123
      }
    },
    "codeSigning": {
      "awsSignerJobId": "48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
    }
  }
]
"roleArn": "arn:aws:iam:123456789012:role/service-role/my_ota_role"
}

```

Salida:

```

{
  "otaUpdateId": "ota12345",
  "awsIotJobId": "job54321",
  "otaUpdateArn": "arn:aws:iot:us-west-2:123456789012:otaupdate/itsaupdate",
  "awsIotJobArn": "arn:aws:iot:us-west-2:123456789012:job/itsajob",
  "otaUpdateStatus": "CREATE_IN_PROGRESS"
}

```

Para obtener más información, consulte [C reateOTAUpdate](#) en la APIReferencia de AWS IoT.

- Para API obtener más información, consulte [CreateOtaUpdate](#)la Referencia de AWS CLI comandos.

## create-policy-version

En el siguiente ejemplo de código, se muestra cómo usar `create-policy-version`.

### AWS CLI

Para actualizar una política con una nueva versión

En el siguiente `create-policy-version` ejemplo, se actualiza la definición de una política y se crea una nueva versión de la política. Este ejemplo también convierte a la nueva versión en la predeterminada.

```

aws iot create-policy-version \
  --policy-name UpdateDeviceCertPolicy \
  --policy-document file://policy.json \

```

**--set-as-default**

Contenidos de `policy.json`:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:UpdateCertificate",
      "Resource": "*"
    }
  ]
}
```

Salida:

```
{
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/UpdateDeviceCertPolicy",
  "policyDocument": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\": \"Allow\", \"Action\": \"iot:UpdateCertificate\", \"Resource\": \"*\" } ] }",
  "policyVersionId": "2",
  "isDefaultVersion": true
}
```

Para obtener más información, consulte [Políticas de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreatePolicyVersion](#) la Referencia de AWS CLI comandos.

## **create-policy**

En el siguiente ejemplo de código, se muestra cómo usar `create-policy`.

### AWS CLI

Para crear una política de AWS IoT

El siguiente `create-policy` ejemplo crea una política de AWS IoT denominada `TemperatureSensorPolicy`. El `policy.json` archivo contiene declaraciones que permiten acciones políticas de AWS IoT.

```
aws iot create-policy \  
  --policy-name TemperatureSensorPolicy \  
  --policy-document file://policy.json
```

Contenidos de `policy.json`:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "iot:Publish",  
        "iot:Receive"  
      ],  
      "Resource": [  
        "arn:aws:iot:us-west-2:123456789012:topic/topic_1",  
        "arn:aws:iot:us-west-2:123456789012:topic/topic_2"  
      ]  
    },  
    {  
      "Effect": "Allow",  
      "Action": [  
        "iot:Subscribe"  
      ],  
      "Resource": [  
        "arn:aws:iot:us-west-2:123456789012:topicfilter/topic_1",  
        "arn:aws:iot:us-west-2:123456789012:topicfilter/topic_2"  
      ]  
    },  
    {  
      "Effect": "Allow",  
      "Action": [  
        "iot:Connect"  
      ],  
      "Resource": [  
        "arn:aws:iot:us-west-2:123456789012:client/basicPubSub"  
      ]  
    }  
  ]  
}
```

**Salida:**

```
{
  "policyName": "TemperatureSensorPolicy",
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TemperatureSensorPolicy",
  "policyDocument": "{
    \"Version\": \"2012-10-17\",
    \"Statement\": [
      {
        \"Effect\": \"Allow\",
        \"Action\": [
          \"iot:Publish\",
          \"iot:Receive\"
        ],
        \"Resource\": [
          \"arn:aws:iot:us-west-2:123456789012:topic/topic_1\",
          \"arn:aws:iot:us-west-2:123456789012:topic/topic_2\"
        ]
      },
      {
        \"Effect\": \"Allow\",
        \"Action\": [
          \"iot:Subscribe\"
        ],
        \"Resource\": [
          \"arn:aws:iot:us-west-2:123456789012:topicfilter/topic_1\",
          \"arn:aws:iot:us-west-2:123456789012:topicfilter/topic_2\"
        ]
      },
      {
        \"Effect\": \"Allow\",
        \"Action\": [
          \"iot:Connect\"
        ],
        \"Resource\": [
          \"arn:aws:iot:us-west-2:123456789012:client/basicPubSub\"
        ]
      }
    ]
  }",
  "policyVersionId": "1"
}
```



```
R0PAQH/BAQDAgeAMA0GCSqGSIb3\ndQEBcWUAA4IBAQAQgix2k6nVqbZFKq97/fZBzLGS0dyz5rT/
E41cDIRX+1j
EPW41\nw0D+2sXheCZLZZnSkvIiP74IToNeXDrjdcaodeGFVHIElRjhMIq+4ZebPbRLtidF
\nRc2hfcTAlqq9Z6v
5Vk6BeM1tu0RqH1wPoVUccLPya8EjNCbnJZUmGd0frN/Y9pho\n5ikV+HPeZhG/k6dhE2GsQJyKfVHL/
uBgKSily
1bRyWU1r6qcpWBNBHjUoD7Hg0wD
\nnzMh4XRb2FQDsqFalkCSYmeL8IVC49sgPD90typ5uteGMTy62usAAUQdq/f
ZvrWg\n0kFpwMVnGKVKT7Kg0kK0LzKWOBB2Jm4/gmrJ\n-----END CERTIFICATE-----\n",
    "keyPair": {
        "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCg
KCAQEAwYSiPeJLSi6k8J4/msjq
\nUwCbfzer0iCQ2b5a2I5AtB08M2nmN06a1pNN0tvb1M1bhDlx10F2W4oYKYN
pun8\n2pFpvf8KY8xPZ8ufsZDx1R+Fp8M+8iuZvEtGoC0/enEQUlp1pqJzlnWNBilc54tA
\nGPoshrnYKxSpuxGn
v79fKF63/NirTgBjuMRtChNlimEXAMPLE3PcWYZVz/3ly4b9\nNPPRqdFlcPT24Sn68ZYiieaurC
+Z1VG1B6uoZU
D2MLAE3jrwKd7tGp0c4E8i9H40\n2D8ip0lWMC1utq/
lWAUhaKxDDgIZKty7D/648aRCpotluJT1UuqBWUSnA/h9
Ssop\neQIDAQAB\n-----END PUBLIC KEY-----\n",
        "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIBAAKCAQEAwYSiPeJLSi6k8J4/
msjqUwCbfzer0iCQ2b5a2I5AtB08M2n
\nmN06a1pNN0tvb1M1bhDlx10F2W4oYKYNpun82pFpvf8KY8xPZ8ufsZ
Dx1R+Fp8M+\n8iuZvEtGoC0/enEQUlp1pqJzlnWNBilc54tAgPoshrnYKxSpuxGnv79fKF63/Nir
\nTgBjuMRtCh
NlimB7E9X8d3PcWYZVz/3ly4b9NPPRqdFlcPT24Sn68ZYiieaurC+Z
\n1VG1B6uoZUD2MLAE3jrwKd7tGp0c4E8i
9H402D8ip0lWMC1utq/lWAUhaKxDDgIZ\nKty7D/648aRCpotluJT1UuqBWUSnA/
h9SsopeQIDAQABAoIBAEAybn
QUtx9T2/nK\nntZ2pA4iugecxI4dz+DmT0XVxs5VJmrx/
nBSq6ejXExEpSIM04RY7LE3ZdJcnd56\nF7tQkkY7yR
VzfxHeXFU1kr0IPuxWebN0rRoPZr+1RSer+ww2aBC525+88pVuR6tM
\nm3pgkrR2ycCj9Fd0UoQxdjHBHaM5PDmJ
9aSxCKdg3nReepeGwsR2TQA+m2vVxWk7\nnou0+91eTOP+/QfP7P8Zj0Ik02XivlRcVDyN/
E4QXPKuIkM/8vS8VK+
E9pATQ0MtB\n2lw8R/YU5AJd6jLEXAMPLEGU2UzRzInNWILtkPPPqgqXXhx0f+mxByjcMalVJk0L
\nh0G2R0UCgY
EA+R0cHNHy/XbsP7Fih0hEh+6Q2QxQ2ncBUPYbBazrR8Hn+7SCICQK
\nVyYfd8Ajfq3e7RsKVL5S1MBp7S1idxak
bIn28fKfPn62DaemGCIOyDgLPf+eUxBx
\nngzbCiBZga8brfurza43UZjKZLpg3hq721+FeAiXi1Nma4Yr9YWEHEN
```

```

8CgYEAxuwT\npzdwWmsiFzfsAw0sy9ySDA/xr5WRWzJyAqUsjsks6rxNzWebpufnYHcmtW7pLdqM
\nkboHwN2pXa
kmZvrk2nKkEMq5brBYGDxuxDe+V369Bianx8aZFyIsckA70wXW1w1h
\nngRC5rQ4X0gp3+Jmw7eA08LRYDjaN846+
Qbt02KcCgYAWS0UL51bijQR0ZwI0dz27\nFQVuCAYsp748aurcRTACCj8jbnK/
QbqTNlxWsaH7ssBjZKo2D5sAqY
BRtASW0Dab\naHXsDhVm2Jye+ESLoHMaCLoyCkT3118yqXIcEDStM07f01Ryag164EiJvSIrMfny\nNL/
fXVjCSH
/udCxdzPt+7QKBgQC+LAD7rxd4J9538hTqpc4XK9vxRbrMXEH55XH
\nHbMa2x0NZXpmeTgEQBukyohCVceyRhK9
i0e6irZTjVXgh0eoTpC8VXkzcnzouTiQ
\neFQQSGfnp7Ioe6UIz23715pKduszSNkMSKrG924ktv7CyDBF1gBQI5g
aDoHnddJBJ\nPRTIZQKBgA8MASXtTxQntRwXXzR92U0vAighiuRkB/mx9jQpUcK1qiqHbkAMqgNF
\nPFCBYIUbFT
iYKKKeJNbyJQvjfsJcKAnaFJ+RnTxk0Q6Wjm20peJ/ii4QiDdnigoE\nvd1c5cFQewWb4/
zqAtPdinkPLN94ileI
79XQdc7R1J0jpgTimL+V\n-----END RSA PRIVATE KEY-----\n"
    },
    "expiration": 1595955066.0
  }
}

```

Para obtener más información, consulte [Aprovisionamiento por usuario de confianza](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [CreateProvisioningClaim](#) la Referencia de AWS CLI comandos.

## create-provisioning-template-version

En el siguiente ejemplo de código, se muestra cómo usar `create-provisioning-template-version`.

### AWS CLI

Para crear una versión de plantilla de aprovisionamiento

En el siguiente ejemplo, se crea una versión para la plantilla de aprovisionamiento especificada. El cuerpo de la nueva versión se incluye en el archivo `template.json`.

```

aws iot create-provisioning-template-version \
  --template-name widget-template \
  --template-body file://template.json

```



Contenidos de `template.json`:

```
{
  "Parameters" : {
    "DeviceLocation": {
      "Type": "String"
    }
  },
  "Mappings": {
    "LocationTable": {
      "Seattle": {
        "LocationUrl": "https://example.aws"
      }
    }
  },
  "Resources" : {
    "thing" : {
      "Type" : "AWS::IoT::Thing",
      "Properties" : {
        "AttributePayload" : {
          "version" : "v1",
          "serialNumber" : "serialNumber"
        },
        "ThingName" : {"Fn::Join":["",["ThingPrefix_",
{"Ref":"SerialNumber"}]]},
        "ThingTypeName" : {"Fn::Join":["",["ThingTypePrefix_",
{"Ref":"SerialNumber"}]]},
        "ThingGroups" : ["widgets", "WA"],
        "BillingGroup": "BillingGroup"
      },
      "OverrideSettings" : {
        "AttributePayload" : "MERGE",
        "ThingTypeName" : "REPLACE",
        "ThingGroups" : "DO_NOTHING"
      }
    },
    "certificate" : {
      "Type" : "AWS::IoT::Certificate",
      "Properties" : {
        "CertificateId": {"Ref": "AWS::IoT::Certificate::Id"},
        "Status" : "Active"
      }
    },
    "policy" : {
```

```

    "Type" : "AWS::IoT::Policy",
    "Properties" : {
      "PolicyDocument" : {
        "Version": "2012-10-17",
        "Statement": [{
          "Effect": "Allow",
          "Action":["iot:Publish"],
          "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/foo/
bar"]
        }]
      }
    }
  },
  "DeviceConfiguration": {
    "FallbackUrl": "https://www.example.com/test-site",
    "LocationUrl": {
      "Fn::FindInMap": ["LocationTable",{"Ref": "DeviceLocation"},
"LocationUrl"]}
    }
  }
}

```

Salida:

```

{
  "templateArn": "arn:aws:iot:us-east-1:123456789012:provisioningtemplate/widget-
template",
  "templateName": "widget-template",
  "versionId": 2,
  "isDefaultVersion": false
}

```

Para obtener más información, consulte [AWS IoT Secure Tunneling](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte la Referencia [CreateProvisioningTemplateVersion](#) de AWS CLI comandos.

## create-provisioning-template

En el siguiente ejemplo de código, se muestra cómo usar create-provisioning-template.

## AWS CLI

Para crear una plantilla de aprovisionamiento

En el siguiente create-provisioning-template ejemplo, se crea una plantilla de aprovisionamiento tal como se define en el archivo. `template.json`

```
aws iot create-provisioning-template \  
  --template-name widget-template \  
  --description "A provisioning template for widgets" \  
  --provisioning-role-arn arn:aws:iam::123456789012:role/Provision_role \  
  --template-body file://template.json
```

Contenidos de `template.json`:

```
{  
  "Parameters" : {  
    "DeviceLocation": {  
      "Type": "String"  
    }  
  },  
  "Mappings": {  
    "LocationTable": {  
      "Seattle": {  
        "LocationUrl": "https://example.aws"  
      }  
    }  
  },  
  "Resources" : {  
    "thing" : {  
      "Type" : "AWS::IoT::Thing",  
      "Properties" : {  
        "AttributePayload" : {  
          "version" : "v1",  
          "serialNumber" : "serialNumber"  
        },  
        "ThingName" : {"Fn::Join":["",["ThingPrefix_",  
{"Ref":"SerialNumber"}]]},  
        "ThingTypeName" : {"Fn::Join":["",["ThingTypePrefix_",  
{"Ref":"SerialNumber"}]]},  
        "ThingGroups" : ["widgets", "WA"],  
        "BillingGroup": "BillingGroup"  
      },  
    },  
  },  
}
```

```

    "OverrideSettings" : {
      "AttributePayload" : "MERGE",
      "ThingTypeName" : "REPLACE",
      "ThingGroups" : "DO_NOTHING"
    }
  },
  "certificate" : {
    "Type" : "AWS::IoT::Certificate",
    "Properties" : {
      "CertificateId": {"Ref": "AWS::IoT::Certificate::Id"},
      "Status" : "Active"
    }
  },
  "policy" : {
    "Type" : "AWS::IoT::Policy",
    "Properties" : {
      "PolicyDocument" : {
        "Version": "2012-10-17",
        "Statement": [{
          "Effect": "Allow",
          "Action":["iot:Publish"],
          "Resource": ["arn:aws:iot:us-east-1:504350838278:topic/foo/
bar"]
        }]
      }
    }
  },
  "DeviceConfiguration": {
    "FallbackUrl": "https://www.example.com/test-site",
    "LocationUrl": {
      "Fn::FindInMap": ["LocationTable",{"Ref": "DeviceLocation"},
"LocationUrl"]}
  }
}

```

Salida:

```

{
  "templateArn": "arn:aws:iot:us-east-1:123456789012:provisioningtemplate/widget-
template",
  "templateName": "widget-template",

```

```
"defaultVersionId": 1
}
```

Para obtener más información, consulte [AWS IoT Secure Tunneling](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte la Referencia [CreateProvisioningTemplate](#) de AWS CLI comandos.

## create-role-alias

En el siguiente ejemplo de código, se muestra cómo usar `create-role-alias`.

### AWS CLI

Para crear un alias de rol

En el siguiente `create-role-alias` ejemplo, se crea un alias de rol llamado `LightBulbRole` para el rol especificado.

```
aws iot create-role-alias \
  --role-alias LightBulbRole \
  --role-arn arn:aws:iam::123456789012:role/lightbulbrole-001
```

Salida:

```
{
  "roleAlias": "LightBulbRole",
  "roleAliasArn": "arn:aws:iot:us-west-2:123456789012:rolealias/LightBulbRole"
}
```

Para obtener más información, consulte [CreateRoleAlias](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [CreateRoleAlias](#) la Referencia de AWS CLI comandos.

## create-scheduled-audit

En el siguiente ejemplo de código, se muestra cómo usar `create-scheduled-audit`.

## AWS CLI

Para crear una auditoría programada

En el siguiente `create-scheduled-audit` ejemplo, se crea una auditoría programada que se ejecuta semanalmente, los miércoles, para comprobar si los certificados de CA o los certificados de los dispositivos van a caducar.

```
aws iot create-scheduled-audit \  
  --scheduled-audit-name WednesdayCertCheck \  
  --frequency WEEKLY \  
  --day-of-week WED \  
  --target-check-  
names CA_CERTIFICATE_EXPIRING_CHECK DEVICE_CERTIFICATE_EXPIRING_CHECK
```

Salida:

```
{  
  "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/  
WednesdayCertCheck"  
}
```

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateScheduledAudit](#) la Referencia de AWS CLI comandos.

## **create-security-profile**

En el siguiente ejemplo de código, se muestra cómo usar `create-security-profile`.

## AWS CLI

Para crear un perfil de seguridad

El siguiente `create-security-profile` ejemplo crea un perfil de seguridad que comprueba si el ancho de banda móvil supera un umbral o si se producen más de 10 errores de autorización en un período de cinco minutos.

```
aws iot create-security-profile \  

```

```
--security-profile-name PossibleIssue \
--security-profile-description "Check to see if authorization fails 10 times in
5 minutes or if cellular bandwidth exceeds 128" \
--behaviors "[{"name":"CellularBandwidth","metric":"aws:message-byte-size",
"criteria":{"comparisonOperator":"greater-than","value":{"count":128},
"consecutiveDatapointsToAlarm":1,"consecutiveDatapointsToClear":1}},{"name
":"Authorization","metric":"aws:num-authorization-failures","criteria":
{"comparisonOperator":"less-than","value":{"count":10},"durationSeconds
":300,"consecutiveDatapointsToAlarm":1,"consecutiveDatapointsToClear":1}}]"
```

Salida:

```
{
  "securityProfileName": "PossibleIssue",
  "securityProfileArn": "arn:aws:iot:us-west-2:123456789012:securityprofile/
PossibleIssue"
}
```

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateSecurityProfile](#) la Referencia de AWS CLI comandos.

## create-stream

En el siguiente ejemplo de código, se muestra cómo usar create-stream.

### AWS CLI

Para crear una secuencia para entregar uno o más archivos grandes en trozos MQTT

En el siguiente create-stream ejemplo, se crea una secuencia para entregar uno o más archivos grandes por partes. MQTT Una transmisión transporta bytes de datos en fragmentos o bloques empaquetados como MQTT mensajes desde una fuente como S3. Puede tener uno o más archivos asociados con un flujo.

```
aws iot create-stream \
--cli-input-json file://create-stream.json
```

Contenidos de create-stream.json:

```
{
  "streamId": "stream12345",
  "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
  "files": [
    {
      "fileId": 123,
      "s3Location": {
        "bucket": "codesign-ota-bucket",
        "key": "48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
      }
    }
  ],
  "roleArn": "arn:aws:iam:123456789012:role/service-role/my_ota_stream_role"
}
```

Salida:

```
{
  "streamId": "stream12345",
  "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream12345",
  "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
  "streamVersion": "1"
}
```

Para obtener más información, consulte [CreateStream](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [CreateStream](#) la Referencia de AWS CLI comandos.

## create-thing-group

En el siguiente ejemplo de código, se muestra cómo usar `create-thing-group`.

AWS CLI

Ejemplo 1: Para crear un grupo de cosas

En el siguiente `create-thing-group` ejemplo, se crea un grupo de cosas denominado `LightBulbs` con una descripción y dos atributos.

```
aws iot create-thing-group \
  --thing-group-name LightBulbs \
```



```
--thing-group-properties "thingGroupDescription=\"Generic bulb group\",  
attributePayload={attributes={Manufacturer=AnyCompany,wattage=60}}"
```

Salida:

```
{  
  "thingGroupName": "LightBulbs",  
  "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs",  
  "thingGroupId": "9198bf9f-1e76-4a88-8e8c-e7140142c331"  
}
```

Ejemplo 2: Para crear un grupo de cosas que forme parte de un grupo principal

A continuación se `create-thing-group` crea un grupo de cosas denominado `HalogenBulbs` que tiene un nombre para el grupo de cosas principal `LightBulbs`.

```
aws iot create-thing-group \  
  --thing-group-name HalogenBulbs \  
  --parent-group-name LightBulbs
```

Salida:

```
{  
  "thingGroupName": "HalogenBulbs",  
  "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/HalogenBulbs",  
  "thingGroupId": "f4ec6b84-b42b-499d-9ce1-4dbd4d4f6f6e"  
}
```

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateThingGroup](#) la Referencia de AWS CLI comandos.

## create-thing-type

En el siguiente ejemplo de código, se muestra cómo usar `create-thing-type`.

AWS CLI

Para definir un tipo de cosa

El siguiente `create-thing-type` ejemplo define un tipo de cosa y los atributos asociados.

```
aws iot create-thing-type \  
  --thing-type-name "LightBulb" \  
  --thing-type-properties "thingTypeDescription=light bulb type,  
  searchableAttributes=wattage,model"
```

Salida:

```
{  
  "thingTypeName": "LightBulb",  
  "thingTypeArn": "arn:aws:iot:us-west-2:123456789012:thingtype/LightBulb",  
  "thingTypeId": "ce3573b0-0a3c-45a7-ac93-4e0ce14cd190"  
}
```

Para obtener más información, consulte [Tipos de cosas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateThingType](#) la Referencia de AWS CLI comandos.

## `create-thing`

En el siguiente ejemplo de código, se muestra cómo usar `create-thing`.

### AWS CLI

Ejemplo 1: Para crear un registro de cosas en el registro

El siguiente `create-thing` ejemplo crea una entrada para un dispositivo en el registro de cosas de AWS IoT.

```
aws iot create-thing \  
  --thing-name SampleIoTThing
```

Salida:

```
{  
  "thingName": "SampleIoTThing",  
  "thingArn": "arn:aws:iot:us-west-2: 123456789012:thing/SampleIoTThing",  
  "thingId": " EXAMPLE1-90ab-cdef-fedc-ba987EXAMPLE "  
}
```

```
}
```

Ejemplo 2: Para definir una cosa que está asociada a un tipo de cosa

En el siguiente `create-thing` ejemplo, se crea una cosa que tenga el tipo de cosa especificado y sus atributos.

```
aws iot create-thing \  
  --thing-name "MyLightBulb" \  
  --thing-type-name "LightBulb" \  
  --attribute-payload '{"attributes": {"wattage": "75", "model": "123"}}'
```

Salida:

```
{  
  "thingName": "MyLightBulb",  
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyLightBulb",  
  "thingId": "40da2e73-c6af-406e-b415-15acae538797"  
}
```

Para obtener más información, consulte [Cómo administrar cosas con el registro y los tipos de cosas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateThing](#) la Referencia de AWS CLI comandos.

## **create-topic-rule-destination**

En el siguiente ejemplo de código, se muestra cómo usar `create-topic-rule-destination`.

AWS CLI

Para crear un tema, un destino de regla

En el siguiente `create-topic-rule-destination` ejemplo, se crea un destino de regla temática para un HTTP punto final.

```
aws iot create-topic-rule-destination \  
  --destination-configuration httpUrlConfiguration={confirmationUrl=https://  
example.com}
```

Salida:

```
{
  "topicRuleDestination": {
    "arn": "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "status": "IN_PROGRESS",
    "statusReason": "Awaiting confirmation. Confirmation message sent on
2020-07-09T22:47:54.154Z; no response received from the endpoint.",
    "httpUrlProperties": {
      "confirmationUrl": "https://example.com"
    }
  }
}
```

Para obtener más información, consulte [Crear un destino de regla temática](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateTopicRuleDestination](#) la Referencia de AWS CLI comandos.

## create-topic-rule

En el siguiente ejemplo de código, se muestra cómo usar `create-topic-rule`.

### AWS CLI

Para crear una regla que envíe una SNS alerta de Amazon

En el siguiente `create-topic-rule` ejemplo, se crea una regla que envía un SNS mensaje de Amazon cuando las lecturas del nivel de humedad del suelo, tal y como se encuentran en la sombra de un dispositivo, son bajas.

```
aws iot create-topic-rule \
  --rule-name "LowMoistureRule" \
  --topic-rule-payload file://plant-rule.json
```

El ejemplo requiere que se guarde el siguiente JSON código en un archivo llamado `plant-rule.json`:

```
{
  "sql": "SELECT * FROM '$aws/things/MyRPi/shadow/update/accepted' WHERE
state.reported.moisture = 'low'\n",
```

```
"description": "Sends an alert whenever soil moisture level readings are too
low.",
"ruleDisabled": false,
"awsIotSqlVersion": "2016-03-23",
"actions": [{
  "sns": {
    "targetArn": "arn:aws:sns:us-
west-2:123456789012:MyRPiLowMoistureTopic",
    "roleArn": "arn:aws:iam::123456789012:role/service-role/
MyRPiLowMoistureTopicRole",
    "messageFormat": "RAW"
  }
}]
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de una regla de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateTopicRule](#) la Referencia de AWS CLI comandos.

## **delete-account-audit-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `delete-account-audit-configuration`.

### AWS CLI

Para deshabilitar todas las comprobaciones de auditoría de su AWS cuenta

El siguiente `delete-account-audit-configuration` ejemplo restaura la configuración predeterminada de AWS IoT Device Defender para esta cuenta, deshabilita todas las comprobaciones de auditoría y borra los datos de configuración. También elimina cualquier auditoría programada para esta cuenta. Utilice este comando con precaución.

```
aws iot delete-account-audit-configuration \
  --delete-scheduled-audits
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteAccountAuditConfiguration](#) la Referencia de AWS CLI comandos.

## **delete-audit-suppression**

En el siguiente ejemplo de código, se muestra cómo usar `delete-audit-suppression`.

### AWS CLI

Para eliminar una supresión de un hallazgo de auditoría

El siguiente `delete-audit-suppression` ejemplo elimina la supresión de un hallazgo de auditoría para `DEVICE __ CERTIFICATE EXPIRING _CHECK`.

```
aws iot delete-audit-suppression \  
  --check-name DEVICE_CERTIFICATE_EXPIRING_CHECK \  
  --resource-identifier deviceCertificateId="c7691e<shortened>"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte las [supresiones de detección de auditorías](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte la Referencia [DeleteAuditSuppression](#) de AWS CLI comandos.

## **delete-authorizer**

En el siguiente ejemplo de código, se muestra cómo usar `delete-authorizer`.

### AWS CLI

Para eliminar un autorizador personalizado

En el siguiente `delete-authorizer` ejemplo, se elimina el autorizador denominado `CustomAuthorizer`. Un autorizador personalizado debe estar en ese `INACTIVE` estado para poder eliminarlo.

```
aws iot delete-authorizer \  
  --authorizer-name CustomAuthorizer
```

```
--authorizer-name CustomAuthorizer
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteAuthorizer](#) la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteAuthorizer](#) la Referencia de AWS CLI comandos.

## **delete-billing-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-billing-group`.

AWS CLI

Para eliminar un grupo de facturación

En el siguiente `delete-billing-group` ejemplo, se elimina el grupo de facturación especificado. Puede eliminar un grupo de facturación incluso si contiene uno o más elementos.

```
aws iot delete-billing-group \  
  --billing-group-name BillingGroupTwo
```

Este comando no produce ninguna salida.

Para obtener más información, consulte [Grupos de facturación](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteBillingGroup](#) la Referencia de AWS CLI comandos.

## **delete-ca-certificate**

En el siguiente ejemplo de código, se muestra cómo usar `delete-ca-certificate`.

AWS CLI

Para eliminar un certificado de CA

En el siguiente `delete-ca-certificate` ejemplo, se elimina el certificado de CA con el identificador de certificado especificado.

```
aws iot delete-ca-certificate \  
  --certificate-  
id f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteCACertificate](#) en la API Referencia de AWS IoT.

- Para API obtener más información, consulte [DeleteCaCertificate](#) la Referencia de AWS CLI comandos.

## **delete-certificate**

En el siguiente ejemplo de código, se muestra cómo usar `delete-certificate`.

### AWS CLI

Para eliminar un certificado de dispositivo

En el siguiente `delete-certificate` ejemplo, se elimina el certificado del dispositivo con el ID especificado.

```
aws iot delete-certificate \  
  --certificate-  
id c0c57bbc8baaf4631a9a0345c957657f5e710473e3ddbbee1428d216d54d53ac9
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteCertificate](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [DeleteCertificate](#) la Referencia de AWS CLI comandos.

## **delete-custom-metric**

En el siguiente ejemplo de código, se muestra cómo usar `delete-custom-metric`.

### AWS CLI

Para eliminar una métrica personalizada

En el siguiente `delete-custom-metric` ejemplo, se elimina una métrica personalizada.



```
aws iot delete-custom-metric \  
  --metric-name batteryPercentage \  
  --region us-east-1
```

Salida:

```
HTTP 200
```

Para obtener más información, consulte [Métricas personalizadas](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [DeleteCustomMetric](#) la Referencia de AWS CLI comandos.

## delete-dimension

En el siguiente ejemplo de código, se muestra cómo usar `delete-dimension`.

AWS CLI

Para eliminar una dimensión

En el `delete-dimension` ejemplo siguiente se elimina una dimensión llamada `TopicFilterForAuthMessages`.

```
aws iot delete-dimension \  
  --name TopicFilterForAuthMessages
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteDimension](#) la Referencia de AWS CLI comandos.

## delete-domain-configuration

En el siguiente ejemplo de código, se muestra cómo usar `delete-domain-configuration`.

## AWS CLI

Para eliminar una configuración de dominio

En el siguiente `delete-domain-configuration` ejemplo, se elimina una configuración de dominio denominada `additionalDataDomain` de su AWS cuenta.

```
aws iot delete-domain-configuration \  
  --domain-configuration-name "additionalDataDomain" \  
  --domain-configuration-status "OK"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Terminales [configurables en](#) la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteDomainConfiguration](#) la Referencia de AWS CLI comandos.

## `delete-dynamic-thing-group`

En el siguiente ejemplo de código, se muestra cómo usar `delete-dynamic-thing-group`.

## AWS CLI

Para eliminar un grupo de cosas dinámico

En el siguiente `delete-dynamic-thing-group` ejemplo, se elimina el grupo de cosas dinámico especificado.

```
aws iot delete-dynamic-thing-group \  
  --thing-group-name "RoomTooWarm"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Grupos de cosas dinámicos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteDynamicThingGroup](#) la Referencia de AWS CLI comandos.

## delete-job-execution

En el siguiente ejemplo de código, se muestra cómo usar `delete-job-execution`.

### AWS CLI

Para eliminar la ejecución de un trabajo

En el siguiente `delete-job-execution` ejemplo, se elimina la ejecución del trabajo especificado en un dispositivo. Se utiliza `describe-job-execution` para obtener el número de ejecución.

```
aws iot delete-job-execution
  --job-id "example-job-02"
  --thing-name "MyRaspberryPi"
  --execution-number 1
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteJobExecution](#) la Referencia de AWS CLI comandos.

## delete-job

En el siguiente ejemplo de código, se muestra cómo usar `delete-job`.

### AWS CLI

Eliminación de un trabajo

En el siguiente `delete-job` ejemplo, se elimina el trabajo especificado. Al especificar la `--force` opción, el trabajo se elimina incluso si el estado es `IN_PROGRESS`.

```
aws iot delete-job \
  --job-id "example-job-04" \
  --force
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteJob](#) la Referencia de AWS CLI comandos.

## **delete-mitigation-action**

En el siguiente ejemplo de código, se muestra cómo usar `delete-mitigation-action`.

### AWS CLI

Para eliminar una acción de mitigación

En el siguiente `delete-mitigation-action` ejemplo, se elimina la acción de mitigación especificada.

```
aws iot delete-mitigation-action \  
  --action-name AddThingsToQuarantineGroup1Action
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteMitigationAction \(Comandos de acción de mitigación\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteMitigationAction](#) la Referencia de AWS CLI comandos.

## **delete-ota-update**

En el siguiente ejemplo de código, se muestra cómo usar `delete-ota-update`.

### AWS CLI

Para eliminar una OTA actualización

En el siguiente `delete-ota-update` ejemplo, se elimina la OTA actualización especificada.

```
aws iot delete-ota-update \  
  --ota-update-id ota12345 \  
  --delete-stream \  
  --force-delete-aws-job
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteOTAUpdate](#) en la API Referencia de AWS IoT.

- Para API obtener más información, consulte [DeleteOtaUpdate](#) la Referencia de AWS CLI comandos.

## **delete-policy-version**

En el siguiente ejemplo de código, se muestra cómo usar `delete-policy-version`.

### AWS CLI

Para eliminar una versión de la política

En el siguiente `delete-policy-version` ejemplo, se elimina la versión 2 de la política especificada de su AWS cuenta.

```
aws iot delete-policy-version \  
  --policy-name UpdateDeviceCertPolicy \  
  --policy-version-id 2
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Políticas de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeletePolicyVersion](#) la Referencia de AWS CLI comandos.

## **delete-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-policy`.

### AWS CLI

Eliminación de una política

En el siguiente `delete-policy` ejemplo, se elimina la política especificada de su AWS cuenta.

```
aws iot delete-policy --policy-name UpdateDeviceCertPolicy
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Políticas de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeletePolicy](#) la Referencia de AWS CLI comandos.

## **delete-provisioning-template-version**

En el siguiente ejemplo de código, se muestra cómo usar `delete-provisioning-template-version`.

### AWS CLI

Para eliminar una versión de plantilla de aprovisionamiento

En el siguiente `delete-provisioning-template-version` ejemplo, se elimina la versión 2 de la plantilla de aprovisionamiento especificada.

```
aws iot delete-provisioning-template-version \  
  --version-id 2 \  
  --template-name "widget-template"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [AWS IoT Secure Tunneling](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte la Referencia [DeleteProvisioningTemplateVersion](#) de AWS CLI comandos.

## **delete-provisioning-template**

En el siguiente ejemplo de código, se muestra cómo usar `delete-provisioning-template`.

### AWS CLI

Para eliminar una plantilla de aprovisionamiento

En el siguiente `delete-provisioning-template` ejemplo, se elimina la plantilla de aprovisionamiento especificada.

```
aws iot delete-provisioning-template \  
  --template-name widget-template
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [AWS IoT Secure Tunneling](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte la Referencia [DeleteProvisioningTemplate](#) de AWS CLI comandos.

## **delete-registration-code**

En el siguiente ejemplo de código, se muestra cómo usar `delete-registration-code`.

### AWS CLI

Para eliminar su código de registro

El siguiente `delete-registration-code` ejemplo elimina el código de registro específico de una cuenta de AWS IoT.

```
aws iot delete-registration-code
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Use su propio certificado](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteRegistrationCode](#) la Referencia de AWS CLI comandos.

## **delete-role-alias**

En el siguiente ejemplo de código, se muestra cómo usar `delete-role-alias`.

### AWS CLI

Para eliminar un alias de rol de AWS IoT

En el siguiente `delete-role-alias` ejemplo, se elimina un alias de rol de AWS IoT denominado `LightBulbRole`.

```
aws iot delete-role-alias \  
  --role-alias LightBulbRole
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Autorización de llamadas directas a AWS los servicios](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteRoleAlias](#) la Referencia de AWS CLI comandos.

## **delete-scheduled-audit**

En el siguiente ejemplo de código, se muestra cómo usar `delete-scheduled-audit`.

### AWS CLI

Para eliminar una auditoría programada

En el siguiente `delete-scheduled-audit` ejemplo, se elimina la auditoría programada de AWS IoT Device Defender denominada `AWSIoTDeviceDefenderDailyAudit`.

```
aws iot delete-scheduled-audit \  
  --scheduled-audit-name AWSIoTDeviceDefenderDailyAudit
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteScheduledAudit](#) la Referencia de AWS CLI comandos.

## **delete-security-profile**

En el siguiente ejemplo de código, se muestra cómo usar `delete-security-profile`.

### AWS CLI

Para eliminar un perfil de seguridad

En el siguiente `delete-security-profile` ejemplo, se elimina un perfil de seguridad denominado `PossibleIssue`.

```
aws iot delete-security-profile \  
  --security-profile-name PossibleIssue
```



```
--security-profile-name PossibleIssue
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteSecurityProfile](#) la Referencia de AWS CLI comandos.

## delete-stream

En el siguiente ejemplo de código, se muestra cómo usar `delete-stream`.

AWS CLI

Para eliminar una transmisión

En el siguiente `delete-stream` ejemplo, se elimina la transmisión especificada.

```
aws iot delete-stream \  
  --stream-id stream12345
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteStream](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [DeleteStream](#) la Referencia de AWS CLI comandos.

## delete-thing-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-thing-group`.

AWS CLI

Para eliminar un grupo de cosas

En el siguiente `delete-thing-group` ejemplo, se elimina el grupo de cosas especificado. No se puede eliminar un grupo de cosas si contiene grupos de cosas secundarios.

```
aws iot delete-thing-group \  
  --thing-group-name thing-group-name
```

```
--thing-group-name DefectiveBulbs
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteThingGroup](#) la Referencia de AWS CLI comandos.

## delete-thing-type

En el siguiente ejemplo de código, se muestra cómo usar delete-thing-type.

AWS CLI

Ejemplo 1: Para eliminar un tipo de cosa

En el siguiente delete-thing-type ejemplo, se elimina un tipo de cosa obsoleto.

```
aws iot delete-thing-type \  
  --thing-type-name "obsoleteThingType"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Tipos de cosas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteThingType](#) la Referencia de AWS CLI comandos.

## delete-thing

En el siguiente ejemplo de código, se muestra cómo usar delete-thing.

AWS CLI

Para mostrar información detallada sobre una cosa

En el siguiente delete-thing ejemplo, se elimina un elemento del registro de AWS IoT de tu AWS cuenta.

¿Fue para borrar algo --thing-name» FourthBulb

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de objetos con el registro](#) en la Guía para desarrolladores de AWS IoT.

- Para obtener API más información, consulte la Referencia de comandos. [DeleteThing](#) AWS CLI

## **delete-topic-rule-destination**

En el siguiente ejemplo de código, se muestra cómo usar `delete-topic-rule-destination`.

### AWS CLI

Para eliminar el destino de una regla de tema

En el siguiente `delete-topic-rule-destination` ejemplo, se elimina el destino de la regla temática especificada.

```
aws iot delete-topic-rule-destination \  
  --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar el destino de una regla temática](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteTopicRuleDestination](#) la Referencia de AWS CLI comandos.

## **delete-topic-rule**

En el siguiente ejemplo de código, se muestra cómo usar `delete-topic-rule`.

### AWS CLI

Para eliminar una regla

En el siguiente `delete-topic-rule` ejemplo, se elimina la regla especificada.

```
aws iot delete-topic-rule \  
  --rule-name "LowMoistureRule"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar una regla](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteTopicRule](#) la Referencia de AWS CLI comandos.

## delete-v2-logging-level

En el siguiente ejemplo de código, se muestra cómo usar `delete-v2-logging-level`.

### AWS CLI

Para eliminar el nivel de registro de un grupo de cosas

En el siguiente `delete-v2-logging-level` ejemplo, se elimina el nivel de registro del grupo de cosas especificado.

```
aws iot delete-v2-logging-level \  
  --target-type THING_GROUP \  
  --target-name LightBulbs
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteV2 LoggingLevel](#) en la Referencia de comandos.AWS CLI

## deprecate-thing-type

En el siguiente ejemplo de código, se muestra cómo usar `deprecate-thing-type`.

### AWS CLI

Ejemplo 1: Para desaprobar un tipo de cosa

En el siguiente `deprecate-thing-type` ejemplo, se desaprueba un tipo de cosa para que los usuarios no puedan asociarle ningún elemento nuevo.

```
aws iot deprecate-thing-type \  
  --thing-type-name "obsoleteThingType"
```

Este comando no genera ninguna salida.

Ejemplo 2: Para revertir la obsolescencia de un tipo de cosa

El siguiente `deprecate-thing-type` ejemplo anula la obsolescencia de un tipo de cosa, lo que permite a los usuarios volver a asociarle cosas nuevas.

```
aws iot deprecate-thing-type \  
  --thing-type-name "obsoleteThingType" \  
  --undo-deprecate
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Tipos de cosas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeprecateThingType](#) la Referencia de AWS CLI comandos.

## **describe-account-audit-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `describe-account-audit-configuration`.

AWS CLI

Para ver los ajustes de configuración de auditoría actuales

En el siguiente `describe-account-audit-configuration` ejemplo, se enumeran los ajustes actuales de la configuración de auditoría de AWS IoT Device Defender.

```
aws iot describe-account-audit-configuration
```

Salida:

```
{  
  "roleArn": "arn:aws:iam::123456789012:role/service-role/  
AWSIoTDeviceDefenderAudit_1551201085996",  
  "auditNotificationTargetConfigurations": {  
    "SNS": {  
      "targetArn": "arn:aws:sns:us-west-2:123456789012:ddaudits",
```

```
        "roleArn": "arn:aws:iam::123456789012:role/service-role/
AWSIoTDeviceDefenderAudit",
        "enabled": true
    }
},
"auditCheckConfigurations": {
    "AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK": {
        "enabled": true
    },
    "CA_CERTIFICATE_EXPIRING_CHECK": {
        "enabled": true
    },
    "CONFLICTING_CLIENT_IDS_CHECK": {
        "enabled": true
    },
    "DEVICE_CERTIFICATE_EXPIRING_CHECK": {
        "enabled": true
    },
    "DEVICE_CERTIFICATE_SHARED_CHECK": {
        "enabled": true
    },
    "IOT_POLICY_OVERLY_PERMISSIVE_CHECK": {
        "enabled": true
    },
    "LOGGING_DISABLED_CHECK": {
        "enabled": true
    },
    "REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK": {
        "enabled": true
    },
    "REVOKED_DEVICE_CERTIFICATE_STILL_ACTIVE_CHECK": {
        "enabled": true
    },
    "UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK": {
        "enabled": true
    }
}
}
```

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeAccountAuditConfiguration](#) la Referencia de AWS CLI comandos.

## describe-audit-finding

En el siguiente ejemplo de código, se muestra cómo usar `describe-audit-finding`.

### AWS CLI

Para enumerar los detalles de una conclusión de auditoría

En el siguiente `describe-audit-finding` ejemplo, se enumeran los detalles del hallazgo de auditoría de AWS IoT Device Defender especificado. Una auditoría puede producir varios resultados. Utilice el `list-audit-findings` comando para obtener una lista de los resultados de una auditoría y obtener la `findingId`.

```
aws iot describe-audit-finding \  
  --finding-id "ef4826b8-e55a-44b9-b460-5c485355371b"
```

Salida:

```
{  
  "finding": {  
    "findingId": "ef4826b8-e55a-44b9-b460-5c485355371b",  
    "taskId": "873ed69c74a9ec8fa9b8e88e9abc4661",  
    "checkName": "IOT_POLICY_OVERLY_PERMISSIVE_CHECK",  
    "taskStartTime": 1576012045.745,  
    "findingTime": 1576012046.168,  
    "severity": "CRITICAL",  
    "nonCompliantResource": {  
      "resourceType": "IOT_POLICY",  
      "resourceIdentifier": {  
        "policyVersionIdentifier": {  
          "policyName": "smp-ggrass-group_Core-policy",  
          "policyVersionId": "1"  
        }  
      }  
    },  
    "reasonForNonCompliance": "Policy allows broad access to IoT data plane  
actions: [iot:Subscribe, iot:Connect, iot:GetThingShadow, iot>DeleteThingShadow,  
iot:UpdateThingShadow, iot:Publish].",  
    "reasonForNonComplianceCode":  
    "ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS"  
  }  
}
```

Para obtener más información, [consulte Comprobar los resultados de la auditoría \(comandos de auditoría\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeAuditFinding](#) la Referencia de AWS CLI comandos.

## describe-audit-mitigation-actions-task

En el siguiente ejemplo de código, se muestra cómo usar `describe-audit-mitigation-actions-task`.

### AWS CLI

Para mostrar los detalles de una tarea de auditoría (acciones de mitigación)

En el siguiente `describe-audit-mitigation-actions-task` ejemplo, se muestran los detalles de la tarea especificada, en la `ResetPolicyVersionAction` que se aplicó a un hallazgo. Los resultados incluyen cuándo comenzó y finalizó la tarea, cuántos hallazgos se buscaron (y cuál fue el resultado) y la definición de la acción que se aplicó como parte de esta tarea.

```
aws iot describe-audit-mitigation-actions-task \
  --task-id ResetPolicyTask01
```

Salida:

```
{
  "taskStatus": "COMPLETED",
  "startTime": "2019-12-10T15:13:19.457000-08:00",
  "endTime": "2019-12-10T15:13:19.947000-08:00",
  "taskStatistics": {
    "IOT_POLICY_OVERLY_PERMISSIVE_CHECK": {
      "totalFindingsCount": 1,
      "failedFindingsCount": 0,
      "succeededFindingsCount": 1,
      "skippedFindingsCount": 0,
      "canceledFindingsCount": 0
    }
  },
  "target": {
    "findingIds": [
```



```

        "ef4826b8-e55a-44b9-b460-5c485355371b"
    ]
},
"auditCheckToActionsMapping": {
    "IOT_POLICY_OVERLY_PERMISSIVE_CHECK": [
        "ResetPolicyVersionAction"
    ]
},
"actionsDefinition": [
    {
        "name": "ResetPolicyVersionAction",
        "id": "1ea0b415-bef1-4a01-bd13-72fb63c59afb",
        "roleArn": "arn:aws:iam::123456789012:role/service-role/
ReplacePolicyVersionRole",
        "actionParams": {
            "replaceDefaultPolicyVersionParams": {
                "templateName": "BLANK_POLICY"
            }
        }
    }
]
}

```

Para obtener más información, consulte [DescribeAuditMitigationActionsTask \(Comandos de acción de mitigación\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeAuditMitigationActionsTask](#) la Referencia de AWS CLI comandos.

## describe-audit-suppression

En el siguiente ejemplo de código, se muestra cómo usar `describe-audit-suppression`.

### AWS CLI

Para obtener detalles sobre la supresión de un hallazgo de auditoría

En el siguiente `describe-audit-suppression` ejemplo se enumeran los detalles sobre la supresión de un hallazgo de auditoría.

```

aws iot describe-audit-task \
  --task-id "787ed873b69cb4d6cdbae6ddd06996c5"

```

**Salida:**

```
{
  "taskStatus": "COMPLETED",
  "taskType": "SCHEDULED_AUDIT_TASK",
  "taskStartTime": 1596168096.157,
  "taskStatistics": {
    "totalChecks": 1,
    "inProgressChecks": 0,
    "waitingForDataCollectionChecks": 0,
    "compliantChecks": 0,
    "nonCompliantChecks": 1,
    "failedChecks": 0,
    "canceledChecks": 0
  },
  "scheduledAuditName": "AWSIoTDeviceDefenderDailyAudit",
  "auditDetails": {
    "DEVICE_CERTIFICATE_EXPIRING_CHECK": {
      "checkRunStatus": "COMPLETED_NON_COMPLIANT",
      "checkCompliant": false,
      "totalResourcesCount": 195,
      "nonCompliantResourcesCount": 2
    }
  }
}
```

Para obtener más información, consulte las [supresiones de detección de auditorías](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte la Referencia [DescribeAuditSuppression](#) de AWS CLI comandos.

**describe-audit-task**

En el siguiente ejemplo de código, se muestra cómo usar `describe-audit-task`.

**AWS CLI**

Para obtener información sobre una instancia de auditoría

El siguiente `describe-audit-task` ejemplo obtiene información sobre una instancia de una auditoría de AWS IoT Device Defender. Si la auditoría se ha completado, las estadísticas resumidas de la ejecución se incluyen en los resultados.

```
aws iot describe-audit-task \  
--task-id a3aea009955e501a31b764abe1bebd3d
```

Salida:

```
{  
  "taskStatus": "COMPLETED",  
  "taskType": "ON_DEMAND_AUDIT_TASK",  
  "taskStartTime": 1560356923.434,  
  "taskStatistics": {  
    "totalChecks": 3,  
    "inProgressChecks": 0,  
    "waitingForDataCollectionChecks": 0,  
    "compliantChecks": 3,  
    "nonCompliantChecks": 0,  
    "failedChecks": 0,  
    "canceledChecks": 0  
  },  
  "auditDetails": {  
    "CA_CERTIFICATE_EXPIRING_CHECK": {  
      "checkRunStatus": "COMPLETED_COMPLIANT",  
      "checkCompliant": true,  
      "totalResourcesCount": 0,  
      "nonCompliantResourcesCount": 0  
    },  
    "DEVICE_CERTIFICATE_EXPIRING_CHECK": {  
      "checkRunStatus": "COMPLETED_COMPLIANT",  
      "checkCompliant": true,  
      "totalResourcesCount": 6,  
      "nonCompliantResourcesCount": 0  
    },  
    "REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK": {  
      "checkRunStatus": "COMPLETED_COMPLIANT",  
      "checkCompliant": true,  
      "totalResourcesCount": 0,  
      "nonCompliantResourcesCount": 0  
    }  
  }  
}
```

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeAuditTask](#) la Referencia de AWS CLI comandos.

## describe-authorizer

En el siguiente ejemplo de código, se muestra cómo usar `describe-authorizer`.

### AWS CLI

Para obtener información sobre un autorizador personalizado

En el siguiente `describe-authorizer` ejemplo, se muestran los detalles del autorizador personalizado especificado.

```
aws iot describe-authorizer \  
  --authorizer-name CustomAuthorizer
```

Salida:

```
{  
  "authorizerDescription": {  
    "authorizerName": "CustomAuthorizer",  
    "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/  
CustomAuthorizer",  
    "authorizerFunctionArn": "arn:aws:lambda:us-  
west-2:123456789012:function:CustomAuthorizerFunction",  
    "tokenKeyName": "MyAuthToken",  
    "tokenSigningPublicKeys": {  
      "FIRST_KEY": "-----BEGIN PUBLIC KEY-----  
\nMIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIBcGKCAQEA1uJOB4lQPgG/1M6ZfIwo  
\nZ+7ENxAio9q6QD4FFqjGZsvjtYwjoe1RKK0U8Eq9xb503kRSmyIwTzwzm/f4Gf0Y  
\nZUloJ+t3PUUwHrmbYTAgrCUgRFyggfVwGCPs5ZAX4Eyqt5cr+AIHIiUDbxSa7p  
\nzw0BKPeic0asNJpqT8PkBbRaKylJh5oo81NDHmVtbBm5A5YiJjqYXLaVAowKzZ\n+GqsNvAQ9Jy1wI2VrEa10fL8f1DB/BJLm7zjpfPOHDJQgID0XnZwAlNnZc0hCwIx\n50g2LW20y9R/  
dmqtDmJiVP97Z4GyKxPvwLYHrUXY0iW1R3AR/Ac1NhCTGZMwVDB1\nlQIDAQAB\n-----END PUBLIC  
KEY-----"  
    },  
    "status": "ACTIVE",  
    "creationDate": 1571245658.069,  
    "lastModifiedDate": 1571245658.069  
  }  
}
```

Para obtener más información, consulte [DescribeAuthorizer](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [DescribeAuthorizer](#) la Referencia de AWS CLI comandos.

## describe-billing-group

En el siguiente ejemplo de código, se muestra cómo usar `describe-billing-group`.

### AWS CLI

Para obtener información sobre un grupo de facturación

En el siguiente `describe-billing-group` ejemplo, se obtiene información del grupo de facturación especificado.

```
aws iot describe-billing-group --billing-group-name GroupOne
```

Salida:

```
{
  "billingGroupName": "GroupOne",
  "billingGroupId": "103de383-114b-4f51-8266-18f209ef5562",
  "billingGroupArn": "arn:aws:iot:us-west-2:123456789012:billinggroup/GroupOne",
  "version": 1,
  "billingGroupProperties": {},
  "billingGroupMetadata": {
    "creationDate": 1560199355.378
  }
}
```

Para obtener más información, consulte [Grupos de facturación](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeBillingGroup](#) la Referencia de AWS CLI comandos.

## describe-ca-certificate

En el siguiente ejemplo de código, se muestra cómo usar `describe-ca-certificate`.

## AWS CLI

Para obtener detalles sobre un certificado de CA

En el siguiente describe-ca-certificate ejemplo, se muestran los detalles del certificado de CA especificado.

```
aws iot describe-ca-certificate \
  --certificate-
  id f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467
```

Salida:

```
{
  "certificateDescription": {
    "certificateArn": "arn:aws:iot:us-west-2:123456789012:cacert/
f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467",
    "certificateId":
"f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467",
    "status": "INACTIVE",
    "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIICzzCCAbegEXAMPLEJANVEPWX18taPMA0GCSqGSIb3DQEBBQUAMB4xCzAJBgNV
\nBAYTA1VMTQ8wDQYDVQQKDAZBbWF6b24wHhcNMTkwOTI0MjEzMTE1WhcNMjkwOTIx
\nMjEzMTE1WjAeMQswCQYDVQQGEwJVUzEPMA0GA1UECgwGQW1hem9uMIIBIjANBgkq
\nhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAZd3R3ioa1CS0MhFWfBrVGR036EK07UAF
\nVdz9EXAMPLE1VczICbADnATK522kEIB51/18Vz1FtAhQL5V5eybXKnB7QebNer5m
\n4Yibx7shR5oqNzFsrXWxuugN5+w5gEfqNMaw0jhF4Lscu1KG49yuqjcDU19/13ua
\n3B2gxs1Pe7TiWwvUskzxb01F2WCshbEJvqY8fIwtGYCjTeJAgQ9hvZx/69XhKen
\nwV9LJw0QxrsUS0Ty8IHwbB8fRy72VM3u7fJoaU+n04jD5cqaoEPtzoEPUEXAMPLE
\nyVAJpqHwgbYbcUfn7V+AB6yh1+0Fa1rEQGuZDPGyJslxwr5vh8nRewIDAQABoxAw
\nDjAMBgNVHRMBETADAQH/MA0GCSqGSIb3DQEBBQUAA4IBAQA+3a5CV3IJg0nd0AgI
\nBgVMtmYzTvqAngx26aG9/spvCjXckh2SBF+EcB1CFwH1yakwjJL1dR4yarnrfxgI
\nEqP4A0YVimAVoQ5FBwnloHe16+3qtDib1U9DeXBUctS55EcfREXAMPLEYtXdqU5C
\nU9ia4KAjV0dxW1+EFYMwX5eGeb0gDTNHBy1V6B/f0SZiQAwDYp4x3B+gAP+a/bWB
\nu1um0qtBdWe6L6/83L+JhaTByqV25iVJ4c/UZUnG8926wU1DM9zQvEXuEVvzZ7+m\n4PSNqst/
nV0vnLpoG4e0WgcJgANuB33CSWtjWSuYsbhmQQRknGhREXAMPLEZT4fm\nfo0e\n-----END
CERTIFICATE-----\n",
    "ownedBy": "123456789012",
    "creationDate": 1569365372.053,
    "autoRegistrationStatus": "DISABLE",
    "lastModifiedDate": 1569365372.053,
    "customerVersion": 1,
    "generationId": "c5c2eb95-140b-4f49-9393-6aaac85b2a90",
```

```

    "validity": {
      "notBefore": 1569360675.0,
      "notAfter": 1884720675.0
    }
  }
}

```

Para obtener más información, consulte [DescribeCACertificate](#) en la API Referencia de AWS IoT.

- Para API obtener más información, consulte [DescribeCaCertificate](#) la Referencia de AWS CLI comandos.

## describe-certificate

En el siguiente ejemplo de código, se muestra cómo usar `describe-certificate`.

### AWS CLI

Para obtener información sobre un certificado

En el siguiente `describe-certificate` ejemplo, se muestran los detalles del certificado especificado.

```

aws iot describe-certificate \
  --certificate-
  id "4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e"

```

Salida:

```

{
  "certificateDescription": {
    "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e",
    "certificateId":
    "4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e",
    "status": "ACTIVE",
    "certificatePem": "-----BEGIN CERTIFICATE-----
MIICiTEXAMPLEQD6m7oRw0uX0jANBgqhkiG9w0BAQUFADCBiDELMAKGA1UEBhMC
VVMxCzAJBgNVBEXAMPLEMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAScT0lBTsBDEXAMPLElMRIwEAYDVQQDEw1UZXR0Q21sYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFTYXpvbi5EXAMPLEcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAKGA1UEBhMCVVMxCzAJBgNEXAMPLEdBMRAwDgYD

```

```
VQQHEwdTZWF0dGx1MQ8wDQYDVQKQEWZBbWF6b24xFDASBgNVBA5TC01BEXAMPLEz
b2xEXAMPLEYDVQKQEWLUZXN0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8EXAMPLEZIHvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYEXAMPLEpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7EXAMPLEGBzZswY6786m86gpE
Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFEXAMPLEAtCu4
nUhVVxYUnEXAMPLE8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GEXAMPLE10ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----",
    "ownedBy": "123456789012",
    "creationDate": 1541022751.983,
    "lastModifiedDate": 1541022751.983,
    "customerVersion": 1,
    "transferData": {},
    "generationId": "6974fbed-2e61-4114-bc5e-4204cc79b045",
    "validity": {
      "notBefore": 1541022631.0,
      "notAfter": 2524607999.0
    }
  }
}
```

Para obtener más información, consulte [DescribeCertificate](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [DescribeCertificate](#) la Referencia de AWS CLI comandos.

## describe-custom-metric

En el siguiente ejemplo de código, se muestra cómo usar `describe-custom-metric`.

### AWS CLI

Para obtener información sobre una métrica personalizada de Device Defender

En el siguiente `describe-custom-metric` ejemplo, se obtiene información sobre una métrica personalizada denominada `myCustomMetric`.

```
aws iot describe-custom-metric \
  --metric-name myCustomMetric
```

Salida:



```
{
  "metricName": "myCustomMetric",
  "metricArn": "arn:aws:iot:us-east-1:1234564789012:custommetric/myCustomMetric",
  "metricType": "number",
  "displayName": "My custom metric",
  "creationDate": 2020-11-17T23:02:12.879000-09:00,
  "lastModifiedDate": 2020-11-17T23:02:12.879000-09:00
}
```

Para obtener más información, consulte [Métricas personalizadas](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [DescribeCustomMetric](#) la Referencia de AWS CLI comandos.

## **describe-default-authorizer**

En el siguiente ejemplo de código, se muestra cómo usar `describe-default-authorizer`.

### AWS CLI

Para obtener información sobre el autorizador personalizado predeterminado

En el siguiente `describe-default-authorizer` ejemplo, se muestran los detalles del autorizador personalizado predeterminado.

```
aws iot describe-default-authorizer
```

Salida:

```
{
  "authorizerName": "CustomAuthorizer",
  "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer"
}
```

Para obtener más información, consulte [DescribeDefaultAuthorizer](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [DescribeDefaultAuthorizer](#) la Referencia de AWS CLI comandos.

## describe-dimension

En el siguiente ejemplo de código, se muestra cómo usar `describe-dimension`.

### AWS CLI

Para obtener información sobre una dimensión

En el `describe-dimension` ejemplo siguiente se obtiene información sobre una dimensión denominada `TopicFilterForAuthMessages`.

```
aws iot describe-dimension \  
  --name TopicFilterForAuthMessages
```

Salida:

```
{  
  "name": "TopicFilterForAuthMessages",  
  "arn": "arn:aws:iot:eu-west-2:123456789012:dimension/  
TopicFilterForAuthMessages",  
  "type": "TOPIC_FILTER",  
  "stringValues": [  
    "device/+/auth"  
  ],  
  "creationDate": 1578620223.255,  
  "lastModifiedDate": 1578620223.255  
}
```

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeDimension](#) la Referencia de AWS CLI comandos.

## describe-domain-configuration

En el siguiente ejemplo de código, se muestra cómo usar `describe-domain-configuration`.

### AWS CLI

Para describir la configuración de un dominio

En el siguiente `describe-domain-configuration` ejemplo, se muestran detalles sobre la configuración de dominio especificada.

```
aws iot describe-domain-configuration \  
  --domain-configuration-name "additionalDataDomain"
```

Salida:

```
{  
  "domainConfigurationName": "additionalDataDomain",  
  "domainConfigurationArn": "arn:aws:iot:us-  
east-1:758EXAMPLE143:domainconfiguration/additionalDataDomain/norpw",  
  "domainName": "d055exampleed74y71zfd-ats.beta.us-east-1.iot.amazonaws.com",  
  "serverCertificates": [],  
  "domainConfigurationStatus": "ENABLED",  
  "serviceType": "DATA",  
  "domainType": "AWS_MANAGED",  
  "lastStatusChangeDate": 1601923783.774  
}
```

Para obtener más información, consulte Terminales [configurables en](#) la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeDomainConfiguration](#) la Referencia de AWS CLI comandos.

## **describe-endpoint**

En el siguiente ejemplo de código, se muestra cómo usar `describe-endpoint`.

### AWS CLI

Ejemplo 1: Para obtener su AWS punto final actual

El siguiente `describe-endpoint` ejemplo recupera el AWS punto final predeterminado al que se aplican todos los comandos.

```
aws iot describe-endpoint
```

Salida:

```
{
  "endpointAddress": "abc123defghijk.iot.us-west-2.amazonaws.com"
}
```

Para obtener más información, consulte [DescribeEndpoint](#) la Guía para desarrolladores de AWS IoT.

Ejemplo 2: Para obtener su ATS terminal

En el siguiente `describe-endpoint` ejemplo, se recupera el punto final de Amazon Trust Services (ATS).

```
aws iot describe-endpoint \
  --endpoint-type iot:Data-ATS
```

Salida:

```
{
  "endpointAddress": "abc123defghijk-ats.iot.us-west-2.amazonaws.com"
}
```

Para obtener más información, consulte [Certificados X.509 e AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeEndpoint](#) la Referencia de AWS CLI comandos.

## **describe-event-configurations**

En el siguiente ejemplo de código, se muestra cómo usar `describe-event-configurations`.

AWS CLI

Para mostrar qué tipos de eventos se publican

El siguiente `describe-event-configurations` ejemplo muestra la configuración que controla qué eventos se generan cuando se agrega, actualiza o elimina algo.

```
aws iot describe-event-configurations
```

Salida:

```
{
  "eventConfigurations": {
    "CA_CERTIFICATE": {
      "Enabled": false
    },
    "CERTIFICATE": {
      "Enabled": false
    },
    "JOB": {
      "Enabled": false
    },
    "JOB_EXECUTION": {
      "Enabled": false
    },
    "POLICY": {
      "Enabled": false
    },
    "THING": {
      "Enabled": false
    },
    "THING_GROUP": {
      "Enabled": false
    },
    "THING_GROUP_HIERARCHY": {
      "Enabled": false
    },
    "THING_GROUP_MEMBERSHIP": {
      "Enabled": false
    },
    "THING_TYPE": {
      "Enabled": false
    },
    "THING_TYPE_ASSOCIATION": {
      "Enabled": false
    }
  }
}
```

Para obtener más información, consulte [Mensajes de eventos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeEventConfigurations](#) la Referencia de AWS CLI comandos.

## describe-index

En el siguiente ejemplo de código, se muestra cómo usar `describe-index`.

### AWS CLI

Para recuperar el estado actual del índice de cosas

El siguiente `describe-index` ejemplo recupera el estado actual del índice de cosas.

```
aws iot describe-index \  
  --index-name "AWS_Things"
```

Salida:

```
{  
  "indexName": "AWS_Things",  
  "indexStatus": "ACTIVE",  
  "schema": "REGISTRY_AND_SHADOW_AND_CONNECTIVITY_STATUS"  
}
```

Para obtener más información, consulte [Gestión de la indexación de cosas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeIndex](#) la Referencia de AWS CLI comandos.

## describe-job-execution

En el siguiente ejemplo de código, se muestra cómo usar `describe-job-execution`.

### AWS CLI

Para obtener los detalles de ejecución de un trabajo en un dispositivo

En el siguiente `describe-job-execution` ejemplo, se obtienen los detalles de ejecución del trabajo especificado.

```
aws iot describe-job-execution \  
  --job-id "example-job-01" \  
  --thing-name "MyRaspberryPi"
```

Salida:

```
{
  "execution": {
    "jobId": "example-job-01",
    "status": "QUEUED",
    "statusDetails": {},
    "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyRaspberryPi",
    "queuedAt": 1560787023.636,
    "lastUpdatedAt": 1560787023.636,
    "executionNumber": 1,
    "versionNumber": 1
  }
}
```

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeJobExecution](#) la Referencia de AWS CLI comandos.

## describe-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-job`.

### AWS CLI

Para obtener el estado detallado de un trabajo

En el siguiente `describe-job` ejemplo se obtiene el estado detallado del trabajo cuyo identificador es `example-job-01`.

```
aws iot describe-job \
  --job-id "example-job-01"
```

Salida:

```
{
  "job": {
    "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-01",
    "jobId": "example-job-01",
    "targetSelection": "SNAPSHOT",
```

```
"status": "IN_PROGRESS",
"targets": [
  "arn:aws:iot:us-west-2:123456789012:thing/MyRaspberryPi"
],
"description": "example job test",
"presignedUrlConfig": {},
"jobExecutionsRolloutConfig": {},
"createdAt": 1560787022.733,
"lastUpdatedAt": 1560787026.294,
"jobProcessDetails": {
  "numberOfCanceledThings": 0,
  "numberOfSucceededThings": 0,
  "numberOfFailedThings": 0,
  "numberOfRejectedThings": 0,
  "numberOfQueuedThings": 1,
  "numberOfInProgressThings": 0,
  "numberOfRemovedThings": 0,
  "numberOfTimedOutThings": 0
},
"timeoutConfig": {}
}
}
```

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeJob](#) la Referencia de AWS CLI comandos.

## describe-mitigation-action

En el siguiente ejemplo de código, se muestra cómo usar `describe-mitigation-action`.

### AWS CLI

Para ver los detalles de una acción de mitigación definida

En el siguiente `describe-mitigation-action` ejemplo, se muestran los detalles de la acción de mitigación especificada.

```
aws iot describe-mitigation-action \
  --action-name AddThingsToQuarantineGroupAction
```

Salida:



```
{
  "actionName": "AddThingsToQuarantineGroupAction",
  "actionType": "ADD_THINGS_TO_THING_GROUP",
  "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
AddThingsToQuarantineGroupAction",
  "actionId": "2fd2726d-98e1-4abf-b10f-09465ccd6bfa",
  "roleArn": "arn:aws:iam::123456789012:role/service-role/
MoveThingsToQuarantineGroupRole",
  "actionParams": {
    "addThingsToThingGroupParams": {
      "thingGroupNames": [
        "QuarantineGroup1"
      ],
      "overrideDynamicGroups": true
    }
  },
  "creationDate": "2019-12-10T11:09:35.999000-08:00",
  "lastModifiedDate": "2019-12-10T11:09:35.999000-08:00"
}
```

Para obtener más información, consulte [DescribeMitigationAction \(Comandos de acción de mitigación\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeMitigationAction](#) la Referencia de AWS CLI comandos.

## describe-provisioning-template-version

En el siguiente ejemplo de código, se muestra cómo usar `describe-provisioning-template-version`.

### AWS CLI

Para describir una versión de plantilla de aprovisionamiento

En el siguiente `describe-provisioning-template-version` ejemplo, se describe una versión de plantilla de aprovisionamiento.

```
aws iot describe-provisioning-template-version \
  --template-name MyTestProvisioningTemplate \
  --version-id 1
```

## Salida:

```

{
  "versionId": 1,
  "creationDate": 1589308310.574,
  "templateBody": "{
    \"Parameters\":{
      \"SerialNumber\":{
        \"Type\": \"String\"
      },
      \"AWS::IoT::Certificate::Id\":{
        \"Type\": \"String\"
      }
    },
    \"Resources\":{
      \"certificate\":{
        \"Properties\":{
          \"CertificateId\":{
            \"Ref\": \"AWS::IoT::Certificate::Id\"
          },
          \"Status\": \"Active\"
        },
        \"Type\": \"AWS::IoT::Certificate\"
      },
      \"policy\":{
        \"Properties\":{
          \"PolicyName\": \"MyIotPolicy\"
        },
        \"Type\": \"AWS::IoT::Policy\"
      },
      \"thing\":{
        \"OverrideSettings\":{
          \"AttributePayload\": \"MERGE\",
          \"ThingGroups\": \"DO_NOTHING\",
          \"ThingTypeName\": \"REPLACE\"
        },
        \"Properties\":{
          \"AttributePayload\": {},
          \"ThingGroups\": [],
          \"ThingName\":{
            \"Fn::Join\": [
              \"\",
              [
                \"DemoGroup_\",

```

```

        {"Ref": "\SerialNumber\"}
      ]
    ],
    "ThingTypeName": "\VirtualThings\"
  },
  "Type": "\AWS::IoT::Thing\"
}
}
}],
"isDefaultVersion": true
}

```

Para obtener más información, consulte [Aprovisionamiento de dispositivos que no tienen certificados de dispositivo mediante el aprovisionamiento de flotas](#) en la Guía básica para desarrolladores de AWS IoT.

- Para API obtener más información, consulte la Referencia [DescribeProvisioningTemplateVersion](#) de AWS CLI comandos.

## describe-provisioning-template

En el siguiente ejemplo de código, se muestra cómo usar describe-provisioning-template.

### AWS CLI

Para describir una plantilla de aprovisionamiento

El siguiente describe-provisioning-template ejemplo describe una plantilla de aprovisionamiento.

```
aws iot describe-provisioning-template \
  --template-name MyTestProvisioningTemplate
```

Salida:

```
{
  "templateArn": "arn:aws:iot:us-west-2:57EXAMPLE833:provisioningtemplate/
MyTestProvisioningTemplate",
  "templateName": "MyTestProvisioningTemplate",
  "creationDate": 1589308310.574,
  "lastModifiedDate": 1589308345.539,
```

```

"defaultVersionId": 1,
"templateBody": "{
  \"Parameters\":{
    \"SerialNumber\":{
      \"Type\": \"String\"
    },
    \"AWS::IoT::Certificate::Id\":{
      \"Type\": \"String\"
    }
  },
  \"Resources\":{
    \"certificate\":{
      \"Properties\":{
        \"CertificateId\":{
          \"Ref\": \"AWS::IoT::Certificate::Id\"
        },
        \"Status\": \"Active\"
      },
      \"Type\": \"AWS::IoT::Certificate\"
    },
    \"policy\":{
      \"Properties\":{
        \"PolicyName\": \"MyIotPolicy\"
      },
      \"Type\": \"AWS::IoT::Policy\"
    },
    \"thing\":{
      \"OverrideSettings\":{
        \"AttributePayload\": \"MERGE\",
        \"ThingGroups\": \"DO_NOTHING\",
        \"ThingTypeName\": \"REPLACE\"
      },
      \"Properties\":{
        \"AttributePayload\": {},
        \"ThingGroups\": [],
        \"ThingName\":{
          \"Fn::Join\": [
            \"\",
            [
              \"DemoGroup_\",
              {\"Ref\": \"SerialNumber\"}
            ]
          ]
        }
      },
    },
  },
}

```

```

        \ThingTypeName\":"VirtualThings\"
    },
    \Type\":"AWS::IoT::Thing\"
  }
}
}],
"enabled": true,
"provisioningRoleArn": "arn:aws:iam::571032923833:role/service-role/IoT_access"
}

```

Para obtener más información, consulte [Aprovisionamiento de dispositivos que no tienen certificados de dispositivo mediante el aprovisionamiento de flotas](#) en la Guía básica para desarrolladores de AWS IoT.

- Para API obtener más información, consulte la Referencia [DescribeProvisioningTemplate](#) de AWS CLI comandos.

## describe-role-alias

En el siguiente ejemplo de código, se muestra cómo usar `describe-role-alias`.

### AWS CLI

Para obtener información sobre un alias de rol de AWS IoT

En el siguiente `describe-role-alias` ejemplo, se muestran los detalles del alias de rol especificado.

```
aws iot describe-role-alias \
  --role-alias LightBulbRole
```

Salida:

```
{
  "roleAliasDescription": {
    "roleAlias": "LightBulbRole",
    "roleAliasArn": "arn:aws:iot:us-west-2:123456789012:rolealias/
LightBulbRole",
    "roleArn": "arn:aws:iam::123456789012:role/light_bulb_role_001",
    "owner": "123456789012",
    "credentialDurationSeconds": 3600,
  }
}
```

```
    "creationDate": 1570558643.221,  
    "lastModifiedDate": 1570558643.221  
  }  
}
```

Para obtener más información, consulte [DescribeRoleAlias](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [DescribeRoleAlias](#) la Referencia de AWS CLI comandos.

## describe-scheduled-audit

En el siguiente ejemplo de código, se muestra cómo usar `describe-scheduled-audit`.

### AWS CLI

Para obtener información sobre una auditoría programada

En el siguiente `describe-scheduled-audit` ejemplo se obtiene información detallada sobre una auditoría programada de AWS IOT Device Defender denominada `AWSIoTDeviceDefenderDailyAudit`.

```
aws iot describe-scheduled-audit \  
  --scheduled-audit-name AWSIoTDeviceDefenderDailyAudit
```

Salida:

```
{  
  "frequency": "DAILY",  
  "targetCheckNames": [  
    "AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK",  
    "CONFLICTING_CLIENT_IDS_CHECK",  
    "DEVICE_CERTIFICATE_SHARED_CHECK",  
    "IOT_POLICY_OVERLY_PERMISSIVE_CHECK",  
    "REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK",  
    "UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK"  
  ],  
  "scheduledAuditName": "AWSIoTDeviceDefenderDailyAudit",  
  "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/  
AWSIoTDeviceDefenderDailyAudit"  
}
```

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeScheduledAudit](#) la Referencia de AWS CLI comandos.

## describe-security-profile

En el siguiente ejemplo de código, se muestra cómo usar `describe-security-profile`.

### AWS CLI

Para obtener información sobre un perfil de seguridad

El siguiente `describe-security-profile` ejemplo obtiene información sobre el perfil de seguridad de AWS IoT Device Defender denominado `PossibleIssue`.

```
aws iot describe-security-profile \  
  --security-profile-name PossibleIssue
```

Salida:

```
{  
  "securityProfileName": "PossibleIssue",  
  "securityProfileArn": "arn:aws:iot:us-west-2:123456789012:securityprofile/  
PossibleIssue",  
  "securityProfileDescription": "check to see if authorization fails 10 times in 5  
minutes or if cellular bandwidth exceeds 128",  
  "behaviors": [  
    {  
      "name": "CellularBandwidth",  
      "metric": "aws:message-byte-size",  
      "criteria": {  
        "comparisonOperator": "greater-than",  
        "value": {  
          "count": 128  
        }  
      },  
      "consecutiveDatapointsToAlarm": 1,  
      "consecutiveDatapointsToClear": 1  
    }  
  ],  
  {
```

```
    "name": "Authorization",
    "metric": "aws:num-authorization-failures",
    "criteria": {
      "comparisonOperator": "greater-than",
      "value": {
        "count": 10
      },
      "durationSeconds": 300,
      "consecutiveDatapointsToAlarm": 1,
      "consecutiveDatapointsToClear": 1
    }
  ],
  "version": 1,
  "creationDate": 1560278102.528,
  "lastModifiedDate": 1560278102.528
}
```

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeSecurityProfile](#) la Referencia de AWS CLI comandos.

## describe-stream

En el siguiente ejemplo de código, se muestra cómo usar `describe-stream`.

### AWS CLI

Para obtener información sobre una transmisión

En el siguiente `describe-stream` ejemplo, se muestran los detalles de la transmisión especificada.

```
aws iot describe-stream \
  --stream-id stream12345
```

Salida:

```
{
```



```
"streamInfo": {
  "streamId": "stream12345",
  "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream12345",
  "streamVersion": 1,
  "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
  "files": [
    {
      "fileId": "123",
      "s3Location": {
        "bucket": "codesign-ota-bucket",
        "key": "48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
      }
    }
  ],
  "createdAt": 1557863215.995,
  "lastUpdatedAt": 1557863215.995,
  "roleArn": "arn:aws:iam:123456789012:role/service-role/my_ota_stream_role"
}
```

Para obtener más información, consulte [DescribeStream](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [DescribeStream](#) la Referencia de AWS CLI comandos.

## describe-thing-group

En el siguiente ejemplo de código, se muestra cómo usar `describe-thing-group`.

### AWS CLI

Para obtener información sobre un grupo de cosas

En el `describe-thing-group` ejemplo siguiente se obtiene información sobre el grupo de cosas denominado `HalogenBulbs`.

```
aws iot describe-thing-group \
  --thing-group-name HalogenBulbs
```

Salida:

```
{
```

```
"thingGroupName": "HalogenBulbs",
"thingGroupId": "f4ec6b84-b42b-499d-9ce1-4dbd4d4f6f6e",
"thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/HalogenBulbs",
"version": 1,
"thingGroupProperties": {},
"thingGroupMetadata": {
  "parentGroupName": "LightBulbs",
  "rootToParentThingGroups": [
    {
      "groupName": "LightBulbs",
      "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/
LightBulbs"
    }
  ],
  "creationDate": 1559927609.897
}
}
```

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeThingGroup](#) la Referencia de AWS CLI comandos.

## describe-thing-type

En el siguiente ejemplo de código, se muestra cómo usar `describe-thing-type`.

### AWS CLI

Para obtener información sobre un tipo de cosa

En el siguiente `describe-thing-type` ejemplo, se muestra información sobre el tipo de cosa especificado definido en su AWS cuenta.

```
aws iot describe-thing-type \
  --thing-type-name "LightBulb"
```

Salida:

```
{
```

```
"thingTypeName": "LightBulb",
"thingTypeId": "ce3573b0-0a3c-45a7-ac93-4e0ce14cd190",
"thingTypeArn": "arn:aws:iot:us-west-2:123456789012:thingtype/LightBulb",
"thingTypeProperties": {
  "thingTypeDescription": "light bulb type",
  "searchableAttributes": [
    "model",
    "wattage"
  ]
},
"thingTypeMetadata": {
  "deprecated": false,
  "creationDate": 1559772562.498
}
}
```

Para obtener más información, consulte [Tipos de cosas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeThingType](#) la Referencia de AWS CLI comandos.

## describe-thing

En el siguiente ejemplo de código, se muestra cómo usar `describe-thing`.

### AWS CLI

Para mostrar información detallada sobre una cosa

El siguiente `describe-thing` ejemplo muestra información sobre una cosa (dispositivo) que está definida en el registro de AWS IoT de su AWS cuenta.

¿Cómo describir-thing --thing-name» MyLightBulb

Salida:

```
{
  "defaultClientId": "MyLightBulb",
  "thingName": "MyLightBulb",
  "thingId": "40da2e73-c6af-406e-b415-15acae538797",
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyLightBulb",
```

```
"thingTypeName": "LightBulb",
"attributes": {
  "model": "123",
  "wattage": "75"
},
"version": 1
}
```

Para obtener más información, consulte [Administración de objetos con el registro](#) en la Guía para desarrolladores de AWS IoT.

- Para obtener API más información, consulte la Referencia de comandos. [DescribeThing](#) AWS CLI

## detach-policy

En el siguiente ejemplo de código, se muestra cómo usar `detach-policy`.

### AWS CLI

Ejemplo 1: Para separar una política de AWS IoT de un grupo de cosas

El siguiente `detach-policy` ejemplo separa la política especificada de un grupo de cosas y, por extensión, de todas las cosas de ese grupo y de cualquiera de los grupos secundarios del grupo.

```
aws iot detach-policy \  
  --target "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs" \  
  --policy-name "MyFirstGroup_Core-policy"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

Ejemplo 2: Para separar una política de AWS IoT del certificado de un dispositivo

En el siguiente `detach-policy` ejemplo, se separa la `TemperatureSensorPolicy` política de un certificado de dispositivo identificado por. ARN

```
aws iot detach-policy \  
  --policy-name TemperatureSensorPolicy \  
  --target "arn:aws:iot:us-west-2:123456789012:certificate/MyFirstGroup_Core-policy"
```

```
--target arn:aws:iot:us-west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DetachPolicy](#) la Referencia de AWS CLI comandos.

## detach-security-profile

En el siguiente ejemplo de código, se muestra cómo usar `detach-security-profile`.

### AWS CLI

Para desasociar un perfil de seguridad de un objetivo

El siguiente `detach-security-profile` ejemplo elimina la asociación entre el perfil de seguridad de AWS IoT Device Defender denominado `Testprofile` y el objetivo `all registered things`.

```
aws iot detach-security-profile \  
  --security-profile-name Testprofile \  
  --security-profile-target-arn "arn:aws:iot:us-west-2:123456789012:all/registered-things"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DetachSecurityProfile](#) la Referencia de AWS CLI comandos.

## detach-thing-principal

En el siguiente ejemplo de código, se muestra cómo usar `detach-thing-principal`.

### AWS CLI

Para separar un certificado/principal de una cosa

En el siguiente `detach-thing-principal` ejemplo, se elimina un certificado que representa un principal del elemento especificado.

```
aws iot detach-thing-principal \  
  --thing-name "MyLightBulb" \  
  --principal "arn:aws:iot:us-  
west-2:123456789012:cert/604c48437a57b7d5fc5d137c5be75011c6ee67c9a6943683a1acb4b1626bac36"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de objetos con el registro](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DetachThingPrincipal](#) la Referencia de AWS CLI comandos.

## disable-topic-rule

En el siguiente ejemplo de código, se muestra cómo usar `disable-topic-rule`.

### AWS CLI

Para deshabilitar una regla temática

En el siguiente `disable-topic-rule` ejemplo, se deshabilita la regla de tema especificada.

```
aws iot disable-topic-rule \  
  --rule-name "MyPlantPiMoistureAlertRule"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Cómo [ver sus reglas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DisableTopicRule](#) la Referencia de AWS CLI comandos.

## enable-topic-rule

En el siguiente ejemplo de código, se muestra cómo usar `enable-topic-rule`.

### AWS CLI

Para habilitar una regla temática

El siguiente `enable-topic-rule` ejemplo habilita (o vuelve a habilitar) la regla temática especificada.

```
aws iot enable-topic-rule \  
  --rule-name "MyPlantPiMoistureAlertRule"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Cómo [ver sus reglas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [EnableTopicRule](#) la Referencia de AWS CLI comandos.

## `get-behavior-model-training-summaries`

En el siguiente ejemplo de código, se muestra cómo usar `get-behavior-model-training-summaries`.

AWS CLI

Para ver el estado del modelo de formación del perfil de seguridad ML Detect de Device Defender

En el siguiente `get-behavior-model-training-summaries` ejemplo, se muestra el estado de entrenamiento del modelo para los comportamientos configurados en el perfil de seguridad elegido. Para cada comportamiento, se muestran el nombre, el estado del modelo y el porcentaje de puntos de datos recopilados.

```
aws iot get-behavior-model-training-summaries \  
  --security-profile-name MySecuirtyProfileName
```

Salida:

```
{  
  "summaries": [  
    {  
      "securityProfileName": "MySecuirtyProfileName",  
      "behaviorName": "Messages_sent_ML_behavior",  
      "modelStatus": "PENDING_BUILD",  
      "datapointsCollectionPercentage": 0.0
```

```

    },
    {
      "securityProfileName": "MySecuirtyProfileName",
      "behaviorName": "Messages_received_ML_behavior",
      "modelStatus": "PENDING_BUILD",
      "datapointsCollectionPercentage": 0.0
    },
    {
      "securityProfileName": "MySecuirtyProfileName",
      "behaviorName": "Authorization_failures_ML_behavior",
      "modelStatus": "PENDING_BUILD",
      "datapointsCollectionPercentage": 0.0
    },
    {
      "securityProfileName": "MySecuirtyProfileName",
      "behaviorName": "Message_size_ML_behavior",
      "modelStatus": "PENDING_BUILD",
      "datapointsCollectionPercentage": 0.0
    },
    {
      "securityProfileName": "MySecuirtyProfileName",
      "behaviorName": "Connection_attempts_ML_behavior",
      "modelStatus": "PENDING_BUILD",
      "datapointsCollectionPercentage": 0.0
    },
    {
      "securityProfileName": "MySPNoALerts",
      "behaviorName": "Disconnects_ML_behavior",
      "modelStatus": "PENDING_BUILD",
      "datapointsCollectionPercentage": 0.0
    }
  ]
}

```

Para obtener más información, consulte [GetBehaviorModelTrainingSummaries \(Detectar comandos\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetBehaviorModelTrainingSummaries](#) la Referencia de AWS CLI comandos.

## get-cardinality

En el siguiente ejemplo de código, se muestra cómo usar `get-cardinality`.



## AWS CLI

Para devolver el recuento aproximado de valores únicos que coinciden con la consulta

Puede utilizar el siguiente script de configuración para crear 10 elementos que representen 10 sensores de temperatura. Cada elemento nuevo tiene 3 atributos.

```
# Bash script. If in other shells, type `bash` before running
Temperatures=(70 71 72 73 74 75 47 97 98 99)
Racks=(Rack1 Rack1 Rack2 Rack2 Rack3 Rack4 Rack5 Rack6 Rack6 Rack6)
IsNormal=(true true true true true true false false false false)
for ((i=0; i<10 ; i++))
do
  thing=$(aws iot create-thing --thing-name "TempSensor$i" --attribute-payload
  attributes="{temperature=${Temperatures[i]},rackId=${Racks[i]},stateNormal=
  ${IsNormal[i]}}")
  aws iot describe-thing --thing-name "TempSensor$i"
done
```

Ejemplo de salida del script de configuración:

```
{
  "version": 1,
  "thingName": "TempSensor0",
  "defaultClientId": "TempSensor0",
  "attributes": {
    "rackId": "Rack1",
    "stateNormal": "true",
    "temperature": "70"
  },
  "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/TempSensor0",
  "thingId": "example1-90ab-cdef-fedc-ba987example"
}
```

El siguiente `get-cardinality` ejemplo consulta los 10 sensores creados por el script de configuración y devuelve el número de racks que tienen sensores de temperatura que indican valores de temperatura anormales. Si el valor de temperatura es inferior a 60 o superior a 80, el sensor de temperatura se encuentra en un estado anormal.

```
aws iot get-cardinality \
  --aggregation-field "attributes.rackId" \
```

```
--query-string "thingName:TempSensor* AND attributes.stateNormal:false"
```

Salida:

```
{
  "cardinality": 2
}
```

Para obtener más información, consulte Consulta de datos agregados < <https://docs.aws.amazon.com/iot/latest/developerguide/index-aggregate.html> > en la Guía para desarrolladores de IoT.AWS

- Para obtener API más información, [GetCardinality](#) consulte AWS CLI la Referencia de comandos.

## get-effective-policies

En el siguiente ejemplo de código, se muestra cómo usar `get-effective-policies`.

AWS CLI

Para enumerar las políticas que afectan a una cosa

En el siguiente `get-effective-policies` ejemplo, se enumeran las políticas que afectan al elemento especificado, incluidas las políticas asociadas a cualquier grupo al que pertenezca.

```
aws iot get-effective-policies \
  --thing-name TemperatureSensor-001 \
  --principal arn:aws:iot:us-west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142
```

Salida:

```
{
  "effectivePolicies": [
    {
      "policyName": "TemperatureSensorPolicy",
      "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TemperatureSensorPolicy",
      "policyDocument": "{
        \"Version\": \"2012-10-17\",
        \"Statement\": [
```

```

    {
      "Effect": "Allow",
      "Action": [
        "iot:Publish",
        "iot:Receive"
      ],
      "Resource": [
        "arn:aws:iot:us-west-2:123456789012:topic/topic_1",
        "arn:aws:iot:us-west-2:123456789012:topic/topic_2"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "iot:Subscribe"
      ],
      "Resource": [
        "arn:aws:iot:us-west-2:123456789012:topicfilter/
topic_1",
        "arn:aws:iot:us-west-2:123456789012:topicfilter/
topic_2"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "iot:Connect"
      ],
      "Resource": [
        "arn:aws:iot:us-west-2:123456789012:client/basicPubSub"
      ]
    }
  ]
}

```

Para obtener más información, consulte [Get Effective Policies for a Thing](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetEffectivePolicies](#) la Referencia de AWS CLI comandos.

## get-indexing-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-indexing-configuration`.

### AWS CLI

Para obtener la configuración de indexación de la cosa

El siguiente `get-indexing-configuration` ejemplo obtiene los datos de configuración actuales para la indexación de flotas de AWS IoT.

```
aws iot get-indexing-configuration
```

Salida:

```
{
  "thingIndexingConfiguration": {
    "thingIndexingMode": "OFF",
    "thingConnectivityIndexingMode": "OFF"
  },
  "thingGroupIndexingConfiguration": {
    "thingGroupIndexingMode": "OFF"
  }
}
```

Para obtener más información, consulte [Gestión de la indexación de cosas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetIndexingConfiguration](#) la Referencia de AWS CLI comandos.

## get-job-document

En el siguiente ejemplo de código, se muestra cómo usar `get-job-document`.

### AWS CLI

Para recuperar el documento de un trabajo

En el siguiente `get-job-document` ejemplo, se muestran detalles sobre el documento del trabajo cuyo identificador es `example-job-01`.

```
aws iot get-job-document \  
  --job-id "example-job-01"
```

Salida:

```
{  
  "document": "\n{\n  \"operation\": \"customJob\", \n  \"otherInfo\":  
  \"someValue\"\n}\n"  
}
```

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetJobDocument](#) la Referencia de AWS CLI comandos.

## get-logging-options

En el siguiente ejemplo de código, se muestra cómo usar `get-logging-options`.

AWS CLI

Para obtener las opciones de registro

En el siguiente `get-logging-options` ejemplo, se obtienen las opciones de registro actuales de su AWS cuenta.

```
aws iot get-logging-options
```

Salida:

```
{  
  "roleArn": "arn:aws:iam::123456789012:role/service-role/iotLoggingRole",  
  "logLevel": "ERROR"  
}
```

Para obtener más información, consulte el título en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetLoggingOptions](#) la Referencia de AWS CLI comandos.

## get-ota-update

En el siguiente ejemplo de código, se muestra cómo usar `get-ota-update`.

### AWS CLI

Para recuperar información sobre una OTA actualización

En el siguiente `get-ota-update` ejemplo, se muestran detalles sobre la OTA actualización especificada.

```
aws iot get-ota-update \  
  --ota-update-id ota12345
```

Salida:

```
{  
  "otaUpdateInfo": {  
    "otaUpdateId": "ota12345",  
    "otaUpdateArn": "arn:aws:iot:us-west-2:123456789012:otaupdate/itsaupdate",  
    "creationDate": 1557863215.995,  
    "lastModifiedDate": 1557863215.995,  
    "description": "A critical update needed right away.",  
    "targets": [  
      "device1",  
      "device2",  
      "device3",  
      "device4"  
    ],  
    "targetSelection": "SNAPSHOT",  
    "protocols": ["HTTP"],  
    "awsJobExecutionsRolloutConfig": {  
      "maximumPerMinute": 10  
    },  
    "otaUpdateFiles": [  
      {  
        "fileName": "firmware.bin",  
        "fileLocation": {  
          "stream": {  
            "streamId": "004",  
            "fileId": 123  
          }  
        }  
      },  
    ],  
  },  
}
```

```

        "codeSigning": {
            "awsSignerJobId": "48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
        }
    },
    "roleArn": "arn:aws:iam:123456789012:role/service-role/my_ota_role"
    "otaUpdateStatus": "CREATE_COMPLETE",
    "awsIotJobId": "job54321",
    "awsIotJobArn": "arn:aws:iot:us-west-2:123456789012:job/job54321",
    "errorInfo": {
    }
}
}

```

Para obtener más información, consulte [GetOTAUpdate](#) en la API Referencia de AWS IoT.

- Para API obtener más información, consulte [GetOtaUpdate](#) la Referencia de AWS CLI comandos.

## get-percentiles

En el siguiente ejemplo de código, se muestra cómo usar `get-percentiles`.

### AWS CLI

Para agrupar los valores agregados que coinciden con la consulta en agrupaciones de percentiles

Puede usar el siguiente script de configuración para crear 10 elementos que representen 10 sensores de temperatura. Cada elemento nuevo tiene un atributo.

```

# Bash script. If in other shells, type `bash` before running
Temperatures=(70 71 72 73 74 75 47 97 98 99)
for ((i=0; i<10 ; i++))
do
    thing=$(aws iot create-thing --thing-name "TempSensor$i" --attribute-payload
attributes="{temperature=${Temperatures[i]}}")
    aws iot describe-thing --thing-name "TempSensor$i"
done

```

Ejemplo de salida del script de configuración:

```
{
```

```

"version": 1,
"thingName": "TempSensor0",
"defaultClientId": "TempSensor0",
"attributes": {
  "temperature": "70"
},
"thingArn": "arn:aws:iot:us-east-1:123456789012:thing/TempSensor0",
"thingId": "example1-90ab-cdef-fedc-ba987example"
}

```

El siguiente `get-percentiles` ejemplo consulta los 10 sensores creados por el script de configuración y devuelve un valor para cada grupo de percentiles especificado. El grupo de percentiles «10» contiene el valor del campo agregado que aparece en aproximadamente el 10 por ciento de los valores que coinciden con la consulta. En el siguiente resultado, {"porcentaje»: 10,0, «valor»: 67,7} significa que aproximadamente el 10,0% de los valores de temperatura están por debajo del 67,7.

```

aws iot get-percentiles \
  --aggregation-field "attributes.temperature" \
  --query-string "thingName:TempSensor*" \
  --percents 10 25 50 75 90

```

Salida:

```

{
  "percentiles": [
    {
      "percent": 10.0,
      "value": 67.7
    },
    {
      "percent": 25.0,
      "value": 71.25
    },
    {
      "percent": 50.0,
      "value": 73.5
    },
    {
      "percent": 75.0,
      "value": 91.5
    },
  ],
}

```



```

    {
      "percent": 90.0,
      "value": 98.1
    }
  ]
}

```

Para obtener más información, consulte [Consultas de datos agregados en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetPercentiles](#) la Referencia de AWS CLI comandos.

## get-policy-version

En el siguiente ejemplo de código, se muestra cómo usar `get-policy-version`.

### AWS CLI

Para obtener información sobre una versión específica de una política

En el siguiente `get-policy-version` ejemplo, se obtiene información sobre la primera versión de la política especificada.

```

aws iot get-policy \
  --policy-name UpdateDeviceCertPolicy
  --policy-version-id "1"

```

Salida:

```

{
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/UpdateDeviceCertPolicy",
  "policyName": "UpdateDeviceCertPolicy",
  "policyDocument": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\": \"Allow\", \"Action\": \"iot:UpdateCertificate\", \"Resource\": \"*\" } ] }",
  "policyVersionId": "1",
  "isDefaultVersion": false,
  "creationDate": 1559925941.924,
  "lastModifiedDate": 1559926175.458,
  "generationId":
  "5066f1b6712ce9d2a1e56399771649a272d6a921762fead080e24fe52f24e042"
}

```

Para obtener más información, consulte [Políticas de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetPolicyVersion](#) la Referencia de AWS CLI comandos.

## get-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-policy`.

### AWS CLI

Para obtener información sobre la versión predeterminada de una política

En el siguiente `get-policy` ejemplo, se recupera información sobre la versión predeterminada de la política especificada.

```
aws iot get-policy \
  --policy-name UpdateDeviceCertPolicy
```

Salida:

```
{
  "policyName": "UpdateDeviceCertPolicy",
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/UpdateDeviceCertPolicy",
  "policyDocument": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\": \"Allow\", \"Action\": \"iot:UpdateCertificate\", \"Resource\": \"*\" } ] }",
  "defaultVersionId": "2",
  "creationDate": 1559925941.924,
  "lastModifiedDate": 1559925941.924,
  "generationId":
  "5066f1b6712ce9d2a1e56399771649a272d6a921762fead080e24fe52f24e042"
}
```

Para obtener más información, consulte [Políticas de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetPolicy](#) la Referencia de AWS CLI comandos.

## get-registration-code

En el siguiente ejemplo de código, se muestra cómo usar `get-registration-code`.

## AWS CLI

Para obtener el código de registro AWS específico de tu cuenta

En el siguiente `get-registration-code` ejemplo, se recupera el código de registro específico de tu AWS cuenta.

```
aws iot get-registration-code
```

Salida:

```
{
  "registrationCode":
  "15c51ae5e36ba59ba77042df1115862076bea4bd15841c838fcb68d5010a614c"
}
```

Para obtener más información, consulte [Use su propio certificado](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetRegistrationCode](#) la Referencia de AWS CLI comandos.

## get-statistics

En el siguiente ejemplo de código, se muestra cómo usar `get-statistics`.

## AWS CLI

Para buscar datos agregados en el índice de dispositivos

En el siguiente `get-statistics` ejemplo, se devuelve el número de elementos que tienen una propiedad denominada `connectivity.connected` establecida en `false` (es decir, el número de dispositivos que no están conectados) en su sombra de dispositivos.

```
aws iot get-statistics \
  --index-name AWS_Things \
  --query-string "connectivity.connected:false"
```

Salida:

```
{
  "statistics": {
    "count": 6
  }
}
```

Para obtener más información, consulte [Obtener estadísticas sobre su flota de dispositivos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetStatistics](#) la Referencia de AWS CLI comandos.

## get-topic-rule-destination

En el siguiente ejemplo de código, se muestra cómo usar `get-topic-rule-destination`.

### AWS CLI

Para obtener el destino de una regla de tema

En el siguiente `get-topic-rule-destination` ejemplo, se obtiene información sobre el destino de una regla temática.

```
aws iot get-topic-rule-destination \
  --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
```

Salida:

```
{
  "topicRuleDestination": {
    "arn": "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "status": "DISABLED",
    "httpUrlProperties": {
      "confirmationUrl": "https://example.com"
    }
  }
}
```

Para obtener más información, consulte [Trabajar con destinos de reglas temáticas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetTopicRuleDestination](#) la Referencia de AWS CLI comandos.

## get-topic-rule

En el siguiente ejemplo de código, se muestra cómo usar `get-topic-rule`.

### AWS CLI

Para obtener información sobre una regla

En el siguiente `get-topic-rule` ejemplo, se obtiene información sobre la regla especificada.

```
aws iot get-topic-rule \  
  --rule-name MyRPiLowMoistureAlertRule
```

Salida:

```
{  
  "ruleArn": "arn:aws:iot:us-west-2:123456789012:rule/MyRPiLowMoistureAlertRule",  
  "rule": {  
    "ruleName": "MyRPiLowMoistureAlertRule",  
    "sql": "SELECT * FROM '$aws/things/MyRPi/shadow/update/accepted' WHERE  
state.reported.moisture = 'low'\n          ",  
    "description": "Sends an alert whenever soil moisture level readings are too  
low.",  
    "createdAt": 1558624363.0,  
    "actions": [  
      {  
        "sns": {  
          "targetArn": "arn:aws:sns:us-  
west-2:123456789012:MyRPiLowMoistureTopic",  
          "roleArn": "arn:aws:iam::123456789012:role/service-role/  
MyRPiLowMoistureTopicRole",  
          "messageFormat": "RAW"  
        }  
      }  
    ],  
    "ruleDisabled": false,  
    "awsIotSqlVersion": "2016-03-23"  
  }  
}
```

Para obtener más información, consulte Cómo [ver sus reglas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetTopicRule](#) la Referencia de AWS CLI comandos.

## **get-v2-logging-options**

En el siguiente ejemplo de código, se muestra cómo usar `get-v2-logging-options`.

### AWS CLI

Para ver una lista de las opciones de registro actuales

En el siguiente `get-v2-logging-options` ejemplo, se enumeran las opciones de registro actuales para el AWS IoT.

```
aws iot get-v2-logging-options
```

Salida:

```
{
  "roleArn": "arn:aws:iam::094249569039:role/service-role/iotLoggingRole",
  "defaultLogLevel": "WARN",
  "disableAllLogs": false
}
```

Para obtener más información, consulte el título en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetV2 LoggingOptions](#) en la referencia de AWS CLI comandos.

## **list-active-violations**

En el siguiente ejemplo de código, se muestra cómo usar `list-active-violations`.

### AWS CLI

Para enumerar las infracciones activas

En el siguiente `list-active-violations` ejemplo, se enumeran todas las infracciones del perfil de seguridad especificado.

```
aws iot list-active-violations \  
--security-profile-name Testprofile
```

Salida:

```
{  
  "activeViolations": [  
    {  
      "violationId": "174db59167fa474c80a652ad1583fd44",  
      "thingName": "iotconsole-1560269126751-1",  
      "securityProfileName": "Testprofile",  
      "behavior": {  
        "name": "Authorization",  
        "metric": "aws:num-authorization-failures",  
        "criteria": {  
          "comparisonOperator": "greater-than",  
          "value": {  
            "count": 10  
          },  
          "durationSeconds": 300,  
          "consecutiveDatapointsToAlarm": 1,  
          "consecutiveDatapointsToClear": 1  
        }  
      },  
      "lastViolationValue": {  
        "count": 0  
      },  
      "lastViolationTime": 1560293700.0,  
      "violationStartTime": 1560279000.0  
    },  
    {  
      "violationId": "c8a9466a093d3b7b35cd44ca58bdbeab",  
      "thingName": "TvnQoEoU",  
      "securityProfileName": "Testprofile",  
      "behavior": {  
        "name": "CellularBandwidth",  
        "metric": "aws:message-byte-size",  
        "criteria": {  
          "comparisonOperator": "greater-than",  
          "value": {  
            "count": 128  
          },  
          "consecutiveDatapointsToAlarm": 1,  
          "consecutiveDatapointsToClear": 1  
        }  
      }  
    }  
  ]  
}
```

```
        "consecutiveDatapointsToClear": 1
      }
    },
    "lastViolationValue": {
      "count": 110
    },
    "lastViolationTime": 1560369000.0,
    "violationStartTime": 1560276600.0
  },
  {
    "violationId": "74aa393adea02e6648f3ac362beed55e",
    "thingName": "iotconsole-1560269232412-2",
    "securityProfileName": "Testprofile",
    "behavior": {
      "name": "Authorization",
      "metric": "aws:num-authorization-failures",
      "criteria": {
        "comparisonOperator": "greater-than",
        "value": {
          "count": 10
        }
      },
      "durationSeconds": 300,
      "consecutiveDatapointsToAlarm": 1,
      "consecutiveDatapointsToClear": 1
    }
  },
  "lastViolationValue": {
    "count": 0
  },
  "lastViolationTime": 1560276600.0,
  "violationStartTime": 1560276600.0
},
{
  "violationId": "1e6ab5f7cf39a1466fcd154e1377e406",
  "thingName": "TvnQoEoU",
  "securityProfileName": "Testprofile",
  "behavior": {
    "name": "Authorization",
    "metric": "aws:num-authorization-failures",
    "criteria": {
      "comparisonOperator": "greater-than",
      "value": {
        "count": 10
      }
    }
  },
```



```

        "durationSeconds": 300,
        "consecutiveDatapointsToAlarm": 1,
        "consecutiveDatapointsToClear": 1
    }
},
"lastViolationValue": {
    "count": 0
},
"lastViolationTime": 1560369000.0,
"violationStartTime": 1560276600.0
}
]
}

```

- Para API obtener más información, consulte [ListActiveViolations](#) la Referencia de AWS CLI comandos.

## list-attached-policies

En el siguiente ejemplo de código, se muestra cómo usar `list-attached-policies`.

AWS CLI

Ejemplo 1: Para enumerar las políticas asociadas a un grupo

En el siguiente `list-attached-policies` ejemplo, se enumeran las políticas asociadas al grupo especificado.

```

aws iot list-attached-policies \
  --target "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"

```

Salida:

```

{
  "policies": [
    {
      "policyName": "UpdateDeviceCertPolicy",
      "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/UpdateDeviceCertPolicy"
    }
  ]
}

```

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

Ejemplo 2: Para enumerar las políticas adjuntas a un certificado de dispositivo

En el siguiente `list-attached-policies` ejemplo, se enumeran las políticas de AWS IoT adjuntas al certificado del dispositivo. El certificado se identifica por su ARN.

```
aws iot list-attached-policies \
  --target arn:aws:iot:us-west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142
```

Salida:

```
{
  "policies": [
    {
      "policyName": "TemperatureSensorPolicy",
      "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TemperatureSensorPolicy"
    }
  ]
}
```

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListAttachedPolicies](#) la Referencia de AWS CLI comandos.

## list-audit-findings

En el siguiente ejemplo de código, se muestra cómo usar `list-audit-findings`.

AWS CLI

Ejemplo 1: Para enumerar todos los resultados de una auditoría

En el siguiente `list-audit-findings` ejemplo, se enumeran todos los resultados de una auditoría de AWS IoT Device Defender con un identificador de tarea específico.

```
aws iot list-audit-findings \
```

```
--task-id a3aea009955e501a31b764abe1bebd3d
```

Salida:

```
{
  "findings": []
}
```

Ejemplo 2: Para enumerar los resultados de un tipo de comprobación de auditoría

El siguiente `list-audit-findings` ejemplo muestra los resultados de las auditorías de AWS IoT Device Defender realizadas entre el 5 y el 19 de junio de 2019, en las que los dispositivos comparten un certificado de dispositivo. Al especificar un nombre de verificación, debe proporcionar una hora de inicio y finalización.

```
aws iot list-audit-findings \
  --check-name DEVICE_CERTIFICATE_SHARED_CHECK \
  --start-time 1559747125 \
  --end-time 1560962028
```

Salida:

```
{
  "findings": [
    {
      "taskId": "eef61068b0eb03c456d746c5a26ee04",
      "checkName": "DEVICE_CERTIFICATE_SHARED_CHECK",
      "taskStartTime": 1560161017.172,
      "findingTime": 1560161017.592,
      "severity": "CRITICAL",
      "nonCompliantResource": {
        "resourceType": "DEVICE_CERTIFICATE",
        "resourceIdentifier": {
          "deviceCertificateId":
            "b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b"
        }
      },
      "relatedResources": [
        {
          "resourceType": "CLIENT_ID",
          "resourceIdentifier": {
            "clientId": "ZipxgAll"
          }
        }
      ]
    }
  ]
}
```

```

        },
        "additionalInfo": {
            "CONNECTION_TIME": "1560086374068"
        }
    },
    {
        "resourceType": "CLIENT_ID",
        "resourceIdentifier": {
            "clientId": "ZipxgAII"
        },
        "additionalInfo": {
            "CONNECTION_TIME": "1560081552187",
            "DISCONNECTION_TIME": "1560086371552"
        }
    },
    {
        "resourceType": "CLIENT_ID",
        "resourceIdentifier": {
            "clientId": "ZipxgAII"
        },
        "additionalInfo": {
            "CONNECTION_TIME": "1559289863631",
            "DISCONNECTION_TIME": "1560081532716"
        }
    }
],
"reasonForNonCompliance": "Certificate shared by one or more devices.",
"reasonForNonComplianceCode": "CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES"
},
{
    "taskId": "bade6b5efd2e1b1569822f6021b39cf5",
    "checkName": "DEVICE_CERTIFICATE_SHARED_CHECK",
    "taskStartTime": 1559988217.27,
    "findingTime": 1559988217.655,
    "severity": "CRITICAL",
    "nonCompliantResource": {
        "resourceType": "DEVICE_CERTIFICATE",
        "resourceIdentifier": {
            "deviceCertificateId":
"b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b"
        }
    },
    "relatedResources": [
        {

```

```

        "resourceType": "CLIENT_ID",
        "resourceIdentifier": {
            "clientId": "xShGENLW"
        },
        "additionalInfo": {
            "CONNECTION_TIME": "1559972350825"
        }
    },
    {
        "resourceType": "CLIENT_ID",
        "resourceIdentifier": {
            "clientId": "xShGENLW"
        },
        "additionalInfo": {
            "CONNECTION_TIME": "1559255062002",
            "DISCONNECTION_TIME": "1559972350616"
        }
    }
],
"reasonForNonCompliance": "Certificate shared by one or more devices.",
"reasonForNonComplianceCode": "CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES"
},
{
    "taskId": "c23f6233ba2d35879c4bb2810fb5ffd6",
    "checkName": "DEVICE_CERTIFICATE_SHARED_CHECK",
    "taskStartTime": 1559901817.31,
    "findingTime": 1559901817.767,
    "severity": "CRITICAL",
    "nonCompliantResource": {
        "resourceType": "DEVICE_CERTIFICATE",
        "resourceIdentifier": {
            "deviceCertificateId":
"b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b"
        }
    },
    "relatedResources": [
        {
            "resourceType": "CLIENT_ID",
            "resourceIdentifier": {
                "clientId": "TvnQoEoU"
            },
            "additionalInfo": {
                "CONNECTION_TIME": "1559826729768"
            }
        }
    ]
}

```

```

    },
    {
      "resourceType": "CLIENT_ID",
      "resourceIdentifier": {
        "clientId": "TvnQoEoU"
      },
      "additionalInfo": {
        "CONNECTION_TIME": "1559345920964",
        "DISCONNECTION_TIME": "1559826728402"
      }
    }
  ],
  "reasonForNonCompliance": "Certificate shared by one or more devices.",
  "reasonForNonComplianceCode": "CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES"
}
]
}

```

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListAuditFindings](#) la Referencia de AWS CLI comandos.

## list-audit-mitigation-actions-executions

En el siguiente ejemplo de código, se muestra cómo usar `list-audit-mitigation-actions-executions`.

### AWS CLI

Para enumerar los detalles de la ejecución de una acción de mitigación de auditoría

Una tarea de acción de mitigación de auditoría aplica una acción de mitigación a uno o más hallazgos de una auditoría de AWS IoT Device Defender. En el siguiente `list-audit-mitigation-actions-executions` ejemplo, se enumeran los detalles de la tarea de acción de mitigación con el resultado especificado `taskId` y el especificado.

```

aws iot list-audit-mitigation-actions-executions \
  --task-id myActionsTaskId \
  --finding-id 0edbaec-2fe1-4cf5-abc9-d4c3e51f7464

```

Salida:

```
{
  "actionsExecutions": [
    {
      "taskId": "myActionsTaskId",
      "findingId": "0edbaaec-2fe1-4cf5-abc9-d4c3e51f7464",
      "actionName": "ResetPolicyVersionAction",
      "actionId": "1ea0b415-bef1-4a01-bd13-72fb63c59afb",
      "status": "COMPLETED",
      "startTime": "2019-12-10T15:19:13.279000-08:00",
      "endTime": "2019-12-10T15:19:13.337000-08:00"
    }
  ]
}
```

Para obtener más información, consulte [ListAuditMitigationActionsExecutions \(Comandos de acción de mitigación\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListAuditMitigationActionsExecutions](#) la Referencia de AWS CLI comandos.

## list-audit-mitigation-actions-tasks

En el siguiente ejemplo de código, se muestra cómo usar `list-audit-mitigation-actions-tasks`.

AWS CLI

Para enumerar las tareas de auditoría y acciones de mitigación

En el siguiente `list-audit-mitigation-actions-tasks` ejemplo, se enumeran las acciones de mitigación que se aplicaron a los hallazgos dentro del período de tiempo especificado.

```
aws iot list-audit-mitigation-actions-tasks \
  --start-time 1594157400 \
  --end-time 1594157430
```

Salida:

```
{
```

```
"tasks": [  
  {  
    "taskId": "0062f2d6-3999-488f-88c7-bef005414103",  
    "startTime": "2020-07-07T14:30:15.172000-07:00",  
    "taskStatus": "COMPLETED"  
  }  
]  
}
```

Para obtener más información, consulte [ListAuditMitigationActionsTasks \(Comandos de acción de mitigación\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListAuditMitigationActionsTasks](#) la Referencia de AWS CLI comandos.

## list-audit-suppressions

En el siguiente ejemplo de código, se muestra cómo usar `list-audit-suppressions`.

AWS CLI

Para enumerar todas las omisiones de los resultados de auditoría

En el siguiente `list-audit-suppressions` ejemplo, se enumeran todas las supresiones de las comprobaciones de auditoría activas.

```
aws iot list-audit-suppressions
```

Salida:

```
{  
  "suppressions": [  
    {  
      "checkName": "DEVICE_CERTIFICATE_EXPIRING_CHECK",  
      "resourceIdentifier": {  
        "deviceCertificateId": "c7691e<shortened>"  
      },  
      "expirationDate": 1597881600.0,  
      "suppressIndefinitely": false  
    }  
  ]  
}
```



```
}
```

Para obtener más información, consulte las [supresiones de detección de auditorías](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte la Referencia [ListAuditSuppressions](#) de AWS CLI comandos.

## list-audit-tasks

En el siguiente ejemplo de código, se muestra cómo usar `list-audit-tasks`.

### AWS CLI

Para enumerar todos los resultados de una auditoría

En el siguiente `list-audit-tasks` ejemplo, se enumeran las tareas de auditoría que se ejecutaron entre el 5 de junio de 2019 y el 12 de junio de 2019.

```
aws iot list-audit-tasks \  
  --start-time 1559747125 \  
  --end-time 1560357228
```

Salida:

```
{  
  "tasks": [  
    {  
      "taskId": "a3aea009955e501a31b764abe1bebd3d",  
      "taskStatus": "COMPLETED",  
      "taskType": "ON_DEMAND_AUDIT_TASK"  
    },  
    {  
      "taskId": "f76b4b5102b632cd9ae38a279c266da1",  
      "taskStatus": "COMPLETED",  
      "taskType": "SCHEDULED_AUDIT_TASK"  
    },  
    {  
      "taskId": "51d9967d9f9ff4d26529505f6d2c444a",  
      "taskStatus": "COMPLETED",  
      "taskType": "SCHEDULED_AUDIT_TASK"  
    }  
  ]  
}
```

```
    },
    {
      "taskId": "eeef61068b0eb03c456d746c5a26ee04",
      "taskStatus": "COMPLETED",
      "taskType": "SCHEDULED_AUDIT_TASK"
    },
    {
      "taskId": "041c49557b7c7b04c079a49514b55589",
      "taskStatus": "COMPLETED",
      "taskType": "SCHEDULED_AUDIT_TASK"
    },
    {
      "taskId": "82c7f2afac1562d18a4560be73998acc",
      "taskStatus": "COMPLETED",
      "taskType": "SCHEDULED_AUDIT_TASK"
    },
    {
      "taskId": "bade6b5efd2e1b1569822f6021b39cf5",
      "taskStatus": "COMPLETED",
      "taskType": "SCHEDULED_AUDIT_TASK"
    },
    {
      "taskId": "c23f6233ba2d35879c4bb2810fb5ffd6",
      "taskStatus": "COMPLETED",
      "taskType": "SCHEDULED_AUDIT_TASK"
    },
    {
      "taskId": "ac9086b7222a2f5e2e17bb6fd30b3aeb",
      "taskStatus": "COMPLETED",
      "taskType": "SCHEDULED_AUDIT_TASK"
    }
  ]
}
```

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListAuditTasks](#) la Referencia de AWS CLI comandos.

## list-authorizers

En el siguiente ejemplo de código, se muestra cómo usar `list-authorizers`.

## AWS CLI

Para incluir tu autorizador personalizado

En el siguiente `list-authorizers` ejemplo, se muestran los autorizadores personalizados de tu AWS cuenta.

```
aws iot list-authorizers
```

Salida:

```
{
  "authorizers": [
    {
      "authorizerName": "CustomAuthorizer",
      "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/CustomAuthorizer"
    },
    {
      "authorizerName": "CustomAuthorizer2",
      "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/CustomAuthorizer2"
    },
    {
      "authorizerName": "CustomAuthorizer3",
      "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/CustomAuthorizer3"
    }
  ]
}
```

Para obtener más información, consulte [ListAuthorizers](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [ListAuthorizers](#) la Referencia de AWS CLI comandos.

## **list-billing-groups**

En el siguiente ejemplo de código, se muestra cómo usar `list-billing-groups`.

## AWS CLI

Para ver una lista de los grupos de facturación de su AWS cuenta y región

En el siguiente `list-billing-groups` ejemplo, se enumeran todos los grupos de facturación que están definidos para su AWS cuenta y AWS región.

```
aws iot list-billing-groups
```

Salida:

```
{
  "billingGroups": [
    {
      "groupName": "GroupOne",
      "groupArn": "arn:aws:iot:us-west-2:123456789012:billinggroup/GroupOne"
    }
  ]
}
```

Para obtener más información, consulte [Grupos de facturación](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListBillingGroups](#) la Referencia de AWS CLI comandos.

## **list-ca-certificates**

En el siguiente ejemplo de código, se muestra cómo usar `list-ca-certificates`.

AWS CLI

Para ver una lista de los certificados de CA registrados en su AWS cuenta

En el siguiente `list-ca-certificates` ejemplo, se muestran los certificados de CA registrados en su AWS cuenta.

```
aws iot list-ca-certificates
```

Salida:

```
{
  "certificates": [
    {
      "certificateArn": "arn:aws:iot:us-west-2:123456789012:cacert/f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467",

```

```

        "certificateId":
        "f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467",
        "status": "INACTIVE",
        "creationDate": 1569365372.053
    }
]
}

```

Para obtener más información, consulte [Use su propio certificado](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListCaCertificates](#) la Referencia de AWS CLI comandos.

## list-certificates-by-ca

En el siguiente ejemplo de código, se muestra cómo usar `list-certificates-by-ca`.

### AWS CLI

Para ver una lista de todos los certificados de dispositivos firmados con un certificado de CA

En el siguiente `list-certificates-by-ca` ejemplo, se enumeran todos los certificados de dispositivo de su AWS cuenta que están firmados con el certificado de CA especificado.

```

aws iot list-certificates-by-ca \
  --ca-certificate-
  id f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467

```

Salida:

```

{
  "certificates": [
    {
      "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
      "certificateId":
      "488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
      "status": "ACTIVE",
      "creationDate": 1569363250.557
    }
  ]
}

```

```
}
```

Para obtener más información, consulte [ListCertificatesByCA](#) in the AWS IoT API Reference.

- Para API obtener más información, consulte [ListCertificatesByCa](#) la Referencia de AWS CLI comandos.

## list-certificates

En el siguiente ejemplo de código, se muestra cómo usar `list-certificates`.

### AWS CLI

Ejemplo 1: Para enumerar los certificados registrados en su AWS cuenta

En el siguiente `list-certificates` ejemplo, se enumeran todos los certificados registrados en su cuenta. Si supera el límite de paginación predeterminado, que es 25, puede utilizar el valor de `nextMarker` respuesta de este comando e introducirlo en el siguiente comando para obtener el siguiente lote de resultados. Repita el procedimiento hasta que se `nextMarker` devuelva sin ningún valor.

```
aws iot list-certificates
```

Salida:

```
{
  "certificates": [
    {
      "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/604c48437a57b7d5fc5d137c5be75011c6ee67c9a6943683a1acb4b1626bac36",
      "certificateId": "604c48437a57b7d5fc5d137c5be75011c6ee67c9a6943683a1acb4b1626bac36",
      "status": "ACTIVE",
      "creationDate": 1556810537.617
    },
    {
      "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/262a1ac8a7d8aa72f6e96e365480f7313aa9db74b8339ec65d34dc3074e1c31e",
      "certificateId": "262a1ac8a7d8aa72f6e96e365480f7313aa9db74b8339ec65d34dc3074e1c31e",
      "status": "ACTIVE",
```

```

        "creationDate": 1546447050.885
    },
    {
        "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/
b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b",
        "certificateId":
        "b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b",
        "status": "ACTIVE",
        "creationDate": 1546292258.322
    },
    {
        "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/7aebeea3845d14a44ec80b06b8b78a89f3f8a706974b8b34d18f5adf0741db42",
        "certificateId":
        "7aebeea3845d14a44ec80b06b8b78a89f3f8a706974b8b34d18f5adf0741db42",
        "status": "ACTIVE",
        "creationDate": 1541457693.453
    },
    {
        "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/54458aa39ebb3eb39c91ffbbdcc3a6ca1c7c094d1644b889f735a6fc2cd9a7e3",
        "certificateId":
        "54458aa39ebb3eb39c91ffbbdcc3a6ca1c7c094d1644b889f735a6fc2cd9a7e3",
        "status": "ACTIVE",
        "creationDate": 1541113568.611
    },
    {
        "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e",
        "certificateId":
        "4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e",
        "status": "ACTIVE",
        "creationDate": 1541022751.983
    }
]
}

```

- Para API obtener más información, consulte [ListCertificates](#) la Referencia de AWS CLI comandos.

## list-custom-metrics

En el siguiente ejemplo de código, se muestra cómo usar `list-custom-metrics`.

## AWS CLI

Para enumerar tus métricas personalizadas

En el siguiente `list-custom-metrics` ejemplo, se muestran todas las métricas personalizadas.

```
aws iot list-custom-metrics \  
  --region us-east-1
```

Salida:

```
{  
  "metricNames": [  
    "batteryPercentage"  
  ]  
}
```

Para obtener más información, consulte [Métricas personalizadas](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [ListCustomMetrics](#) la Referencia de AWS CLI comandos.

## **list-dimensions**

En el siguiente ejemplo de código, se muestra cómo usar `list-dimensions`.

## AWS CLI

Para ver las dimensiones de tu AWS cuenta

En el siguiente `list-dimensions` ejemplo, se enumeran todas las dimensiones de AWS IoT Device Defender definidas en su AWS cuenta.

```
aws iot list-dimensions
```

Salida:

```
{  
  "dimensionNames": [  

```



```
    "TopicFilterForAuthMessages",  
    "TopicFilterForActivityMessages"  
  ]  
}
```

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListDimensions](#) la Referencia de AWS CLI comandos.

## list-domain-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-domain-configurations`.

AWS CLI

Para enumerar las configuraciones de dominio

En el siguiente `list-domain-configurations` ejemplo, se enumeran las configuraciones de dominio de su AWS cuenta que tienen el tipo de servicio especificado.

```
aws iot list-domain-configurations \  
  --service-type "DATA"
```

Salida:

```
{  
  "domainConfigurations":  
  [  
    {  
      "domainConfigurationName": "additionalDataDomain",  
      "domainConfigurationArn": "arn:aws:iot:us-  
west-2:123456789012:domainconfiguration/additionalDataDomain/dikMh",  
      "serviceType": "DATA"  
    },  
    {  
      "domainConfigurationName": "iot:Jobs",  
      "domainConfigurationArn": "arn:aws:iot:us-  
west-2:123456789012:domainconfiguration/iot:Jobs",  
      "serviceType": "JOBS"  
    }  
  ]  
}
```

```
    },
    {
      "domainConfigurationName": "iot:Data-ATS",
      "domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/iot:Data-ATS",
      "serviceType": "DATA"
    },
    {
      "domainConfigurationName": "iot:CredentialProvider",
      "domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/iot:CredentialProvider",
      "serviceType": "CREDENTIAL_PROVIDER"
    }
  ]
}
```

Para obtener más información, consulte Terminales [configurables en](#) la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListDomainConfigurations](#) la Referencia de AWS CLI comandos.

## list-indices

En el siguiente ejemplo de código, se muestra cómo usar `list-indices`.

### AWS CLI

Para ver una lista de los índices de búsqueda configurados

El siguiente `list-indices` ejemplo muestra todos los índices de búsqueda configurados en su AWS cuenta. Si no has activado la indexación de cosas, es posible que no tengas ningún índice.

```
aws iot list-indices
```

Salida:

```
{
  "indexNames": [
    "AWS_Things"
  ]
}
```

Para obtener más información, consulte [Gestión de la indexación de cosas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListIndices](#) la Referencia de AWS CLI comandos.

## **list-job-executions-for-job**

En el siguiente ejemplo de código, se muestra cómo usar `list-job-executions-for-job`.

### AWS CLI

Para enumerar los trabajos de su AWS cuenta

En el siguiente `list-job-executions-for-job` ejemplo, se enumeran todas las ejecuciones de tareas de una tarea de su AWS cuenta, especificadas en `jobId`.

```
aws iot list-job-executions-for-job \  
  --job-id my-ota-job
```

Salida:

```
{  
  "executionSummaries": [  
    {  
      "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/my_thing",  
      "jobExecutionSummary": {  
        "status": "QUEUED",  
        "queuedAt": "2022-03-07T15:58:42.195000-08:00",  
        "lastUpdatedAt": "2022-03-07T15:58:42.195000-08:00",  
        "executionNumber": 1,  
        "retryAttempt": 0  
      }  
    }  
  ]  
}
```

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListJobExecutionsForJob](#) la Referencia de AWS CLI comandos.

## list-job-executions-for-thing

En el siguiente ejemplo de código, se muestra cómo usar `list-job-executions-for-thing`.

### AWS CLI

Para enumerar los trabajos que se ejecutaron para una cosa

En el `list-job-executions-for-thing` ejemplo siguiente se enumeran todos los trabajos que se ejecutaron para la cosa denominada `MyRaspberryPi`.

```
aws iot list-job-executions-for-thing \
  --thing-name "MyRaspberryPi"
```

Salida:

```
{
  "executionSummaries": [
    {
      "jobId": "example-job-01",
      "jobExecutionSummary": {
        "status": "QUEUED",
        "queuedAt": 1560787023.636,
        "lastUpdatedAt": 1560787023.636,
        "executionNumber": 1
      }
    }
  ]
}
```

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListJobExecutionsForThing](#) la Referencia de AWS CLI comandos.

## list-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-jobs`.

### AWS CLI

Para enumerar los trabajos de su AWS cuenta

En el siguiente `list-jobs` ejemplo, se enumeran todos los trabajos de su AWS cuenta, ordenados por su estado.

```
aws iot list-jobs
```

Salida:

```
{
  "jobs": [
    {
      "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-01",
      "jobId": "example-job-01",
      "targetSelection": "SNAPSHOT",
      "status": "IN_PROGRESS",
      "createdAt": 1560787022.733,
      "lastUpdatedAt": 1560787026.294
    }
  ]
}
```

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListJobs](#) la Referencia de AWS CLI comandos.

## **list-mitigation-actions**

En el siguiente ejemplo de código, se muestra cómo usar `list-mitigation-actions`.

AWS CLI

Para enumerar todas las acciones de mitigación definidas

En el siguiente `list-mitigation-actions` ejemplo, se enumeran todas las acciones de mitigación definidas para su AWS cuenta y región. Para cada acción, se muestran el nombre y la fecha de creación. ARN

```
aws iot list-mitigation-actions
```

Salida:

```
{
  "actionIdentifiers": [
    {
      "actionName": "DeactivateCACertAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/DeactivateCACertAction",
      "creationDate": "2019-12-10T11:12:47.574000-08:00"
    },
    {
      "actionName": "ResetPolicyVersionAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/ResetPolicyVersionAction",
      "creationDate": "2019-12-10T11:11:48.920000-08:00"
    },
    {
      "actionName": "PublishFindingToSNSAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/PublishFindingToSNSAction",
      "creationDate": "2019-12-10T11:10:49.546000-08:00"
    },
    {
      "actionName": "AddThingsToQuarantineGroupAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/AddThingsToQuarantineGroupAction",
      "creationDate": "2019-12-10T11:09:35.999000-08:00"
    },
    {
      "actionName": "UpdateDeviceCertAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/UpdateDeviceCertAction",
      "creationDate": "2019-12-10T11:08:44.263000-08:00"
    },
    {
      "actionName": "SampleMitigationAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/SampleMitigationAction",
      "creationDate": "2019-12-10T11:03:41.840000-08:00"
    }
  ]
}
```

Para obtener más información, consulte [ListMitigationActions \(Comandos de acción de mitigación\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListMitigationActions](#) la Referencia de AWS CLI comandos.

## list-mitigations-actions

En el siguiente ejemplo de código, se muestra cómo usar `list-mitigations-actions`.

AWS CLI

Para enumerar todas las acciones de mitigación definidas

En el siguiente `list-mitigations-actions` ejemplo, se enumeran todas las acciones de mitigación definidas para su AWS cuenta y región. Para cada acción, se muestran el nombre y la fecha de creación. ARN

```
aws iot list-mitigation-actions
```

Salida:

```
{
  "actionIdentifiers": [
    {
      "actionName": "DeactivateCACertAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/DeactivateCACertAction",
      "creationDate": "2019-12-10T11:12:47.574000-08:00"
    },
    {
      "actionName": "ResetPolicyVersionAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/ResetPolicyVersionAction",
      "creationDate": "2019-12-10T11:11:48.920000-08:00"
    },
    {
      "actionName": "PublishFindingToSNSAction",
      "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/PublishFindingToSNSAction",
      "creationDate": "2019-12-10T11:10:49.546000-08:00"
    },
    {
      "actionName": "AddThingsToQuarantineGroupAction",
```

```

        "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
AddThingsToQuarantineGroupAction",
        "creationDate": "2019-12-10T11:09:35.999000-08:00"
    },
    {
        "actionName": "UpdateDeviceCertAction",
        "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
UpdateDeviceCertAction",
        "creationDate": "2019-12-10T11:08:44.263000-08:00"
    },
    {
        "actionName": "SampleMitigationAction",
        "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
SampleMitigationAction",
        "creationDate": "2019-12-10T11:03:41.840000-08:00"
    }
]
}

```

Para obtener más información, consulte [ListMitigationActions \(Comandos de acción de mitigación\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListMitigationsActions](#) la Referencia de AWS CLI comandos.

## list-ota-updates

En el siguiente ejemplo de código, se muestra cómo usar `list-ota-updates`.

AWS CLI

Para enumerar OTA las actualizaciones de la cuenta

En el siguiente `list-ota-updates` ejemplo, se enumeran las OTA actualizaciones disponibles.

```
aws iot list-ota-updates
```

Salida:

```
{
  "otaUpdates": [
```



```
{
  "otaUpdateId": "itsaupdate",
  "otaUpdateArn": "arn:aws:iot:us-west-2:123456789012:otaupdate/
itsaupdate",
  "creationDate": 1557863215.995
}
]
```

Para obtener más información, consulte [ListOTAUpdates](#) en la APIReferencia de AWS IoT.

- Para API obtener más información, consulte [ListOtaUpdates](#)la Referencia de AWS CLI comandos.

## list-outgoing-certificates

En el siguiente ejemplo de código, se muestra cómo usar `list-outgoing-certificates`.

### AWS CLI

Para ver una lista de los certificados que se están transfiriendo a una AWS cuenta diferente

En el siguiente `list-outgoing-certificates` ejemplo, se enumeran todos los certificados de dispositivo que se están transfiriendo a una AWS cuenta diferente mediante el `transfer-certificate` comando.

```
aws iot list-outgoing-certificates
```

Salida:

```
{
  "outgoingCertificates": [
    {
      "certificateArn": "arn:aws:iot:us-
west-2:030714055129:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
      "certificateId":
"488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
      "transferredTo": "030714055129",
      "transferDate": 1569427780.441,
      "creationDate": 1569363250.557
    }
  ]
}
```

```
]
}
```

Para obtener más información, consulte [ListOutgoingCertificates](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [ListOutgoingCertificates](#) la Referencia de AWS CLI comandos.

## list-policies

En el siguiente ejemplo de código, se muestra cómo usar `list-policies`.

### AWS CLI

Para ver una lista de las políticas definidas en su AWS cuenta

En el siguiente `list-policies` ejemplo, se enumeran todas las políticas definidas en su AWS cuenta.

```
aws iot list-policies
```

Salida:

```
{
  "policies": [
    {
      "policyName": "UpdateDeviceCertPolicy",
      "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/UpdateDeviceCertPolicy"
    },
    {
      "policyName": "PlantIoTPolicy",
      "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/PlantIoTPolicy"
    },
    {
      "policyName": "MyPiGroup_Core-policy",
      "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/MyPiGroup_Core-policy"
    }
  ]
}
```

Para obtener más información, consulte [Políticas de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListPolicies](#) la Referencia de AWS CLI comandos.

## list-policy-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-policy-versions`.

### AWS CLI

Ejemplo 1: Para ver todas las versiones de una política

En el siguiente `list-policy-versions` ejemplo, se enumeran todas las versiones de la política especificada y sus fechas de creación.

```
aws iot list-policy-versions \  
  --policy-name LightBulbPolicy
```

Salida:

```
{  
  "policyVersions": [  
    {  
      "versionId": "2",  
      "isDefaultVersion": true,  
      "createDate": 1559925941.924  
    },  
    {  
      "versionId": "1",  
      "isDefaultVersion": false,  
      "createDate": 1559925941.924  
    }  
  ]  
}
```

Para obtener más información, consulte [Políticas de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListPolicyVersions](#) la Referencia de AWS CLI comandos.

## list-principal-things

En el siguiente ejemplo de código, se muestra cómo usar `list-principal-things`.

### AWS CLI

Para enumerar las cosas adjuntas a un principio

En el siguiente `list-principal-things` ejemplo se enumeran los elementos adjuntos al principal especificado por un ARN.

```
aws iot list-principal-things \  
  --principal arn:aws:iot:us-  
west-2:123456789012:cert/2e1eb273792174ec2b9bf4e9b37e6c6c692345499506002a35159767055278e8
```

Salida:

```
{  
  "things": [  
    "DeskLamp",  
    "TableLamp"  
  ]  
}
```

Para obtener más información, consulte [ListPrincipalThings](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [ListPrincipalThings](#) la Referencia de AWS CLI comandos.

## list-provisioning-template-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-provisioning-template-versions`.

### AWS CLI

Para enumerar las versiones de las plantillas de aprovisionamiento

En el siguiente `list-provisioning-template-versions` ejemplo, se enumeran las versiones disponibles de la plantilla de aprovisionamiento especificada.

```
aws iot list-provisioning-template-versions \  
  --template-name
```

```
--template-name "widget-template"
```

Salida:

```
{
  "versions": [
    {
      "versionId": 1,
      "creationDate": 1574800471.339,
      "isDefaultVersion": true
    },
    {
      "versionId": 2,
      "creationDate": 1574801192.317,
      "isDefaultVersion": false
    }
  ]
}
```

Para obtener más información, consulte [AWS IoT Secure Tunneling](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte la Referencia [ListProvisioningTemplateVersions](#) de AWS CLI comandos.

## list-provisioning-templates

En el siguiente ejemplo de código, se muestra cómo usar `list-provisioning-templates`.

AWS CLI

Para enumerar las plantillas de aprovisionamiento

En el siguiente `list-provisioning-templates` ejemplo, se enumeran todas las plantillas de aprovisionamiento de su AWS cuenta.

```
aws iot list-provisioning-templates
```

Salida:

```
{
```

```
"templates": [  
  {  
    "templateArn": "arn:aws:iot:us-east-1:123456789012:provisioningtemplate/  
widget-template",  
    "templateName": "widget-template",  
    "description": "A provisioning template for widgets",  
    "creationDate": 1574800471.367,  
    "lastModifiedDate": 1574801192.324,  
    "enabled": false  
  }  
]
```

Para obtener más información, consulte [AWS IoT Secure Tunneling](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte la Referencia [ListProvisioningTemplates](#) de AWS CLI comandos.

## **list-role-aliases**

En el siguiente ejemplo de código, se muestra cómo usar `list-role-aliases`.

### AWS CLI

Para enumerar los alias de los roles de AWS IoT en tu cuenta AWS

En el siguiente `list-role-aliases` ejemplo, se enumeran los alias de rol de AWS IoT de su AWS cuenta.

```
aws iot list-role-aliases
```

Salida:

```
{  
  "roleAliases": [  
    "ResidentAlias",  
    "ElectricianAlias"  
  ]  
}
```

Para obtener más información, consulte [ListRoleAliases](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [ListRoleAliases](#) la Referencia de AWS CLI comandos.

## list-scheduled-audits

En el siguiente ejemplo de código, se muestra cómo usar `list-scheduled-audits`.

### AWS CLI

Para ver una lista de las auditorías programadas para su AWS cuenta

En el siguiente `list-scheduled-audits` ejemplo, se enumeran todas las auditorías programadas para su AWS cuenta.

```
aws iot list-scheduled-audits
```

Salida:

```
{
  "scheduledAudits": [
    {
      "scheduledAuditName": "AWSIoTDeviceDefenderDailyAudit",
      "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/AWSIoTDeviceDefenderDailyAudit",
      "frequency": "DAILY"
    },
    {
      "scheduledAuditName": "AWSDeviceDefenderWeeklyAudit",
      "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/AWSDeviceDefenderWeeklyAudit",
      "frequency": "WEEKLY",
      "dayOfWeek": "SUN"
    }
  ]
}
```

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListScheduledAudits](#) la Referencia de AWS CLI comandos.

## list-security-profiles-for-target

En el siguiente ejemplo de código, se muestra cómo usar `list-security-profiles-for-target`.

### AWS CLI

Para ver una lista de los perfiles de seguridad adjuntos a un objetivo

En el siguiente `list-security-profiles-for-target` ejemplo, se enumeran los perfiles de seguridad de AWS IoT Device Defender que están conectados a dispositivos no registrados.

```
aws iot list-security-profiles-for-target \
  --security-profile-target-arn "arn:aws:iot:us-west-2:123456789012:all/
  unregistered-things"
```

Salida:

```
{
  "securityProfileTargetMappings": [
    {
      "securityProfileIdentifier": {
        "name": "Testprofile",
        "arn": "arn:aws:iot:us-west-2:123456789012:securityprofile/
Testprofile"
      },
      "target": {
        "arn": "arn:aws:iot:us-west-2:123456789012:all/unregistered-things"
      }
    }
  ]
}
```

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListSecurityProfilesForTarget](#) la Referencia de AWS CLI comandos.

## list-security-profiles

En el siguiente ejemplo de código, se muestra cómo usar `list-security-profiles`.



## AWS CLI

Para ver una lista de los perfiles de seguridad de su AWS cuenta

En el siguiente `list-security-profiles` ejemplo, se enumeran todos los perfiles de seguridad de AWS IoT Device Defender que están definidos en su AWS cuenta.

```
aws iot list-security-profiles
```

Salida:

```
{
  "securityProfileIdentifiers": [
    {
      "name": "Testprofile",
      "arn": "arn:aws:iot:us-west-2:123456789012:securityprofile/Testprofile"
    }
  ]
}
```

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListSecurityProfiles](#) la Referencia de AWS CLI comandos.

## `list-streams`

En el siguiente ejemplo de código, se muestra cómo usar `list-streams`.

## AWS CLI

Para ver una lista de las transmisiones de la cuenta

En el siguiente `list-streams` ejemplo, se enumeran todas las transmisiones de tu AWS cuenta.

```
aws iot list-streams
```

Salida:

```
{
```

```
"streams": [  
  {  
    "streamId": "stream12345",  
    "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream12345",  
    "streamVersion": 1,  
    "description": "This stream is used for Amazon FreeRTOS OTA Update  
12345."  
  },  
  {  
    "streamId": "stream54321",  
    "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream54321",  
    "streamVersion": 1,  
    "description": "This stream is used for Amazon FreeRTOS OTA Update  
54321."  
  }  
]
```

Para obtener más información, consulte [ListStreams](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [ListStreams](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para mostrar las etiquetas y sus valores asociados a un recurso

En el siguiente `list-tags-for-resource` ejemplo, se muestran las etiquetas y los valores asociados al grupo de cosas `LightBulbs`.

```
aws iot list-tags-for-resource \  
  --resource-arn "arn:aws:iot:us-west-2:094249569039:thinggroup/LightBulbs"
```

Salida:

```
{  
  "tags": [  
    {  
      "Key": "Assembly",  
      "Value": "Fact1NW"    }  
  ]  
}
```

```
    },  
    {  
      "Key": "MyTag",  
      "Value": "777"  
    }  
  ]  
}
```

Para obtener más información, consulte [Etiquetar sus recursos de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## list-targets-for-policy

En el siguiente ejemplo de código, se muestra cómo usar `list-targets-for-policy`.

### AWS CLI

Para enumerar los principios asociados a una política de AWS IoT

En el siguiente `list-targets-for-policy` ejemplo, se enumeran los certificados de dispositivo a los que se adjunta la política especificada.

```
aws iot list-targets-for-policy \  
  --policy-name UpdateDeviceCertPolicy
```

Salida:

```
{  
  "targets": [  
    "arn:aws:iot:us-west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",  
    "arn:aws:iot:us-west-2:123456789012:cert/  
d1eb269fb55a628552143c8f96eb3c258fcd5331ea113e766ba0c82bf225f0be"  
  ]  
}
```

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListTargetsForPolicy](#) la Referencia de AWS CLI comandos.

## list-targets-for-security-profile

En el siguiente ejemplo de código, se muestra cómo usar `list-targets-for-security-profile`.

### AWS CLI

Para enumerar los objetivos a los que se aplica un perfil de seguridad

En el siguiente `list-targets-for-security-profile` ejemplo, se enumeran los objetivos a los que se aplica el perfil de seguridad de AWS IoT Device Defender denominado `PossibleIssue`.

```
aws iot list-targets-for-security-profile \  
  --security-profile-name Testprofile
```

Salida:

```
{  
  "securityProfileTargets": [  
    {  
      "arn": "arn:aws:iot:us-west-2:123456789012:all/unregistered-things"  
    },  
    {  
      "arn": "arn:aws:iot:us-west-2:123456789012:all/registered-things"  
    }  
  ]  
}
```

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListTargetsForSecurityProfile](#) la Referencia de AWS CLI comandos.

## list-thing-groups-for-thing

En el siguiente ejemplo de código, se muestra cómo usar `list-thing-groups-for-thing`.

## AWS CLI

Para enumerar los grupos a los que pertenece una cosa

En el siguiente `list-thing-groups-for-thing` ejemplo, se enumeran los grupos a los que pertenece la cosa especificada.

```
aws iot list-thing-groups-for-thing \  
  --thing-name MyLightBulb
```

Salida:

```
{  
  "thingGroups": [  
    {  
      "groupName": "DeadBulbs",  
      "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/DeadBulbs"  
    },  
    {  
      "groupName": "LightBulbs",  
      "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"  
    }  
  ]  
}
```

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListThingGroupsForThing](#) la Referencia de AWS CLI comandos.

## **list-thing-groups**

En el siguiente ejemplo de código, se muestra cómo usar `list-thing-groups`.

## AWS CLI

Para enumerar los grupos de cosas definidos en su AWS cuenta

En el siguiente `describe-thing-group` ejemplo, se enumeran todos los grupos de cosas definidos en su AWS cuenta.

```
aws iot list-thing-groups
```

Salida:

```
{
  "thingGroups": [
    {
      "groupName": "HalogenBulbs",
      "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/HalogenBulbs"
    },
    {
      "groupName": "LightBulbs",
      "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"
    }
  ]
}
```

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListThingGroups](#) la Referencia de AWS CLI comandos.

## list-thing-principals

En el siguiente ejemplo de código, se muestra cómo usar `list-thing-principals`.

AWS CLI

Para enumerar los principios asociados a una cosa

En el siguiente `list-thing-principals` ejemplo, se enumeran los principales (certificados X.509, IAM usuarios, grupos, roles, identidades de Amazon Cognito o identidades federadas) asociados al elemento especificado.

```
aws iot list-thing-principals \  
  --thing-name MyRaspberryPi
```

Salida:

```
{
```

```
"principals": [  
  "arn:aws:iot:us-  
west-2:123456789012:cert/33475ac865079a5ffd5ecd44240640349293facc760642d7d8d5dbb6b4c86893"  
]  
}
```

Para obtener más información, consulte [ListThingPrincipals](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [ListThingPrincipals](#) la Referencia de AWS CLI comandos.

## list-thing-types

En el siguiente ejemplo de código, se muestra cómo usar `list-thing-types`.

### AWS CLI

Para enumerar los tipos de cosas definidos

En el siguiente `list-thing-types` ejemplo, se muestra una lista de los tipos de cosas definidos en su AWS cuenta.

```
aws iot list-thing-types
```

Salida:

```
{  
  "thingTypes": [  
    {  
      "thingTypeName": "LightBulb",  
      "thingTypeArn": "arn:aws:iot:us-west-2:123456789012:thingtype/  
LightBulb",  
      "thingTypeProperties": {  
        "thingTypeDescription": "light bulb type",  
        "searchableAttributes": [  
          "model",  
          "wattage"  
        ]  
      },  
      "thingTypeMetadata": {  
        "deprecated": false,  
        "creationDate": 1559772562.498  
      }  
    }  
  ]  
}
```

```
    }  
  ]  
}
```

Para obtener más información, consulte [Tipos de cosas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListThingTypes](#) la Referencia de AWS CLI comandos.

## **list-things-in-billing-group**

En el siguiente ejemplo de código, se muestra cómo usar `list-things-in-billing-group`.

AWS CLI

Para enumerar los elementos de un grupo de facturación

En el siguiente `list-things-in-billing-group` ejemplo, se enumeran los elementos que se encuentran en el grupo de facturación especificado.

```
aws iot list-things-in-billing-group \  
  --billing-group-name GroupOne
```

Salida:

```
{  
  "things": [  
    "MyOtherLightBulb",  
    "MyLightBulb"  
  ]  
}
```

Para obtener más información, consulte [Grupos de facturación](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListThingsInBillingGroup](#) la Referencia de AWS CLI comandos.

## **list-things-in-thing-group**

En el siguiente ejemplo de código, se muestra cómo usar `list-things-in-thing-group`.



## AWS CLI

Para enumerar las cosas que pertenecen a un grupo

En el siguiente `list-things-in-thing-group` ejemplo, se enumeran las cosas que pertenecen al grupo de cosas especificado.

```
aws iot list-things-in-thing-group \  
  --thing-group-name LightBulbs
```

Salida:

```
{  
  "things": [  
    "MyLightBulb"  
  ]  
}
```

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListThingsInThingGroup](#) la Referencia de AWS CLI comandos.

## **list-things**

En el siguiente ejemplo de código, se muestra cómo usar `list-things`.

## AWS CLI

Ejemplo 1: Creación de una lista de todos los objetos del registro

En el siguiente `list-things` ejemplo, se enumeran las cosas (dispositivos) que están definidas en el registro de AWS IoT de su AWS cuenta.

```
aws iot list-things
```

Salida:

```
{  
  "things": [  

```

```
{
  "thingName": "ThirdBulb",
  "thingTypeName": "LightBulb",
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/ThirdBulb",
  "attributes": {
    "model": "123",
    "wattage": "75"
  },
  "version": 2
},
{
  "thingName": "MyOtherLightBulb",
  "thingTypeName": "LightBulb",
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyOtherLightBulb",
  "attributes": {
    "model": "123",
    "wattage": "75"
  },
  "version": 3
},
{
  "thingName": "MyLightBulb",
  "thingTypeName": "LightBulb",
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyLightBulb",
  "attributes": {
    "model": "123",
    "wattage": "75"
  },
  "version": 1
},
{
  "thingName": "SampleIoTThing",
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/SampleIoTThing",
  "attributes": {},
  "version": 1
}
]
```

Ejemplo 2: Creación de una lista de los objetos definidos que tienen un atributo específico

En el siguiente ejemplo de `list-things` se muestra una lista de objetos que tienen un atributo denominado `wattage`.

```
aws iot list-things \  
  --attribute-name wattage
```

Salida:

```
{  
  "things": [  
    {  
      "thingName": "MyLightBulb",  
      "thingTypeName": "LightBulb",  
      "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyLightBulb",  
      "attributes": {  
        "model": "123",  
        "wattage": "75"  
      },  
      "version": 1  
    },  
    {  
      "thingName": "MyOtherLightBulb",  
      "thingTypeName": "LightBulb",  
      "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyOtherLightBulb",  
      "attributes": {  
        "model": "123",  
        "wattage": "75"  
      },  
      "version": 3  
    }  
  ]  
}
```

Para obtener más información, consulte [Administración de objetos con el registro](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListThings](#) la Referencia de AWS CLI comandos.

## list-topic-rule-destinations

En el siguiente ejemplo de código, se muestra cómo usar `list-topic-rule-destinations`.

### AWS CLI

Para enumerar los destinos de las reglas de su tema

En el siguiente `list-topic-rule-destinations` ejemplo, se enumeran todos los destinos de las reglas temáticas que ha definido en la AWS región actual.

```
aws iot list-topic-rule-destinations
```

Salida:

```
{
  "destinationSummaries": [
    {
      "arn": "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "status": "ENABLED",
      "httpUrlSummary": {
        "confirmationUrl": "https://example.com"
      }
    }
  ]
}
```

Para obtener más información, consulte [Trabajar con destinos de reglas temáticas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListTopicRuleDestinations](#) la Referencia de AWS CLI comandos.

## **list-topic-rules**

En el siguiente ejemplo de código, se muestra cómo usar `list-topic-rules`.

AWS CLI

Para enumerar sus reglas

En el `list-topic-rules` ejemplo siguiente se enumeran todas las reglas que ha definido.

```
aws iot list-topic-rules
```

Salida:

```
{
  "rules": [
```

```
{
  "ruleArn": "arn:aws:iot:us-west-2:123456789012:rule/
MyRPiLowMoistureAlertRule",
  "ruleName": "MyRPiLowMoistureAlertRule",
  "topicPattern": "$aws/things/MyRPi/shadow/update/accepted",
  "createdAt": 1558624363.0,
  "ruleDisabled": false
},
{
  "ruleArn": "arn:aws:iot:us-west-2:123456789012:rule/
MyPlantPiMoistureAlertRule",
  "ruleName": "MyPlantPiMoistureAlertRule",
  "topicPattern": "$aws/things/MyPlantPi/shadow/update/accepted",
  "createdAt": 1541458459.0,
  "ruleDisabled": false
}
]
```

Para obtener más información, consulte Cómo [ver sus reglas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListTopicRules](#) la Referencia de AWS CLI comandos.

## list-v2-logging-levels

En el siguiente ejemplo de código, se muestra cómo usar `list-v2-logging-levels`.

### AWS CLI

Para enumerar los niveles de registro

En el siguiente `list-v2-logging-levels` ejemplo, se enumeran los niveles de registro configurados. Si no se establecieron los niveles de registro, `NotConfiguredException` se produce una al ejecutar este comando.

```
aws iot list-v2-logging-levels
```

Salida:

```
{
```

```

    "logTargetConfigurations": [
      {
        "logTarget": {
          "targetType": "DEFAULT"
        },
        "logLevel": "ERROR"
      }
    ]
  }
}

```

- Para API obtener más información, consulte [ListV2 LoggingLevels](#) en la Referencia de AWS CLI comandos.

## list-violation-events

En el siguiente ejemplo de código, se muestra cómo usar `list-violation-events`.

### AWS CLI

Para enumerar las infracciones del perfil de seguridad durante un período

En el siguiente `list-violation-events` ejemplo, se enumeran las infracciones que se produjeron entre el 5 y el 12 de junio de 2019 en todos los perfiles de seguridad de AWS IoT Device Defender de la AWS cuenta corriente y la AWS región.

```

aws iot list-violation-events \
  --start-time 1559747125 \
  --end-time 1560351925

```

Salida:

```

{
  "violationEvents": [
    {
      "violationId": "174db59167fa474c80a652ad1583fd44",
      "thingName": "iotconsole-1560269126751-1",
      "securityProfileName": "Testprofile",
      "behavior": {
        "name": "Authorization",
        "metric": "aws:num-authorization-failures",
        "criteria": {
          "comparisonOperator": "greater-than",

```

```
        "value": {
            "count": 10
        },
        "durationSeconds": 300,
        "consecutiveDatapointsToAlarm": 1,
        "consecutiveDatapointsToClear": 1
    }
},
"metricValue": {
    "count": 0
},
"violationEventType": "in-alarm",
"violationEventTime": 1560279000.0
},
{
    "violationId": "c8a9466a093d3b7b35cd44ca58bdbbeab",
    "thingName": "TvnQoEoU",
    "securityProfileName": "Testprofile",
    "behavior": {
        "name": "CellularBandwidth",
        "metric": "aws:message-byte-size",
        "criteria": {
            "comparisonOperator": "greater-than",
            "value": {
                "count": 128
            },
            "consecutiveDatapointsToAlarm": 1,
            "consecutiveDatapointsToClear": 1
        }
    },
    "metricValue": {
        "count": 110
    },
    "violationEventType": "in-alarm",
    "violationEventTime": 1560276600.0
},
{
    "violationId": "74aa393adea02e6648f3ac362beed55e",
    "thingName": "iotconsole-1560269232412-2",
    "securityProfileName": "Testprofile",
    "behavior": {
        "name": "Authorization",
        "metric": "aws:num-authorization-failures",
        "criteria": {
```

```

        "comparisonOperator": "greater-than",
        "value": {
            "count": 10
        },
        "durationSeconds": 300,
        "consecutiveDatapointsToAlarm": 1,
        "consecutiveDatapointsToClear": 1
    }
},
"metricValue": {
    "count": 0
},
"violationEventType": "in-alarm",
"violationEventTime": 1560276600.0
},
{
    "violationId": "1e6ab5f7cf39a1466fcd154e1377e406",
    "thingName": "TvnQoEoU",
    "securityProfileName": "Testprofile",
    "behavior": {
        "name": "Authorization",
        "metric": "aws:num-authorization-failures",
        "criteria": {
            "comparisonOperator": "greater-than",
            "value": {
                "count": 10
            },
            "durationSeconds": 300,
            "consecutiveDatapointsToAlarm": 1,
            "consecutiveDatapointsToClear": 1
        }
    },
    "metricValue": {
        "count": 0
    },
    "violationEventType": "in-alarm",
    "violationEventTime": 1560276600.0
}
]
}

```

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.



- Para API obtener más información, consulte [ListViolationEvents](#) la Referencia de AWS CLI comandos.

## register-ca-certificate

En el siguiente ejemplo de código, se muestra cómo usar `register-ca-certificate`.

### AWS CLI

Para registrar un certificado de una entidad emisora de certificados (CA)

En el siguiente `register-ca-certificate` ejemplo, se registra un certificado de CA. El comando proporciona el certificado de CA y un certificado de verificación de clave que demuestra que usted es el propietario de la clave privada asociada al certificado de CA.

```
aws iot register-ca-certificate \  
  --ca-certificate file://rootCA.pem \  
  --verification-cert file://verificationCert.pem
```

Salida:

```
{  
  "certificateArn": "arn:aws:iot:us-west-2:123456789012:cacert/  
f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467",  
  "certificateId":  
  "f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467"  
}
```

Para obtener más información, consulte [RegisterCACertificate](#) en la API Referencia de AWS IoT.

- Para API obtener más información, consulte [RegisterCaCertificate](#) la Referencia de AWS CLI comandos.

## register-certificate

En el siguiente ejemplo de código, se muestra cómo usar `register-certificate`.

### AWS CLI

Para registrar un certificado de dispositivo autofirmado

En el siguiente `register-certificate` ejemplo, se registra el certificado de `deviceCert.pem` dispositivo firmado por el certificado de `rootCA.pem` CA. El certificado de CA debe registrarse antes de usarlo para registrar un certificado de dispositivo autofirmado. El certificado autofirmado debe estar firmado por el mismo certificado de CA que se pasa a este comando.

```
aws iot register-certificate \
  --certificate-pem file://deviceCert.pem \
  --ca-certificate-pem file://rootCA.pem
```

Salida:

```
{
  "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
  "certificateId":
  "488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142"
}
```

Para obtener más información, consulte [RegisterCertificate](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [RegisterCertificate](#) la Referencia de AWS CLI comandos.

## register-thing

En el siguiente ejemplo de código, se muestra cómo usar `register-thing`.

### AWS CLI

Para registrar una cosa

En el siguiente `register-thing` ejemplo, se registra una cosa mediante una plantilla de aprovisionamiento.

```
aws iot register-thing \
  --template-body '{"Parameters":{"ThingName":
{"Type":"String"},"AWS::IoT::Certificate::Id":{"Type":"String"}},'Resources':
{"certificate":{"Properties":{"CertificateId":
{"Ref":"AWS::IoT::Certificate::Id"},"Status":"Active"},"Type":"AWS::IoT::Certificate"},"poli
{"Properties":{"PolicyName":"MyIotPolicy"},"Type":"AWS::IoT::Policy"},"thing":
{"OverrideSettings":
```

```
{ "AttributePayload": "MERGE", "ThingGroups": "DO_NOTHING", "ThingTypeName": "REPLACE", "Properties": { "AttributePayload": {}, "ThingGroups": [], "ThingName": { "Ref": "ThingName", "ThingTypeName": "VirtualThings", "Type": "AWS::IoT::Thing" } } }' \
--parameters '{ "ThingName": "Register-thing-trial-1", "AWS::IoT::Certificate::Id": "799a9ea048a1e6aea42b55EXAMPLEf8697b4bafcd77a318a3068e30404b9233c" }
```

Salida:

```
{
  "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCCAkGgAwIBAgIUYLk81I35cIppobpw
Hi0J2jNjboIwDQYJKoZIhvcNAQEL
\nBQAwTTFLEkGA1UECwwxCQW1hem9uIFd1YiBTZXJ2aWN1cyBPPUftYXpvbi
5jb20g\nSW5jLiBMPVNlYXR0bGUgU1Q9V2FzaGluZ3RvbiBDPVVTMB4XDTIwMDcyMzE2NDUw
\n0VoXDTQ5MTIzMT
IzNTk10VowHjEcMBoGA1UEAwwTQVdTIElvcyBDbDZXJ0aWZpY2F0\nZTCCASIdQYJKoZIhvcNAQEBBQADggEPADCC
AQoCggEBA071uADhdBajqTmgiMV5\nnmCFfBZQRMo1MdtVoZr2X+M4MzL
+RARrtUzH9a2SMackeX8Keb1I0TKz0RI
RDXnyE
\n6lV0wjgAsd0ku22rFxex4eG2ikha7pYYkvuToqA7L3TxItRvfKrxRI4ZfJoFPip4\nKqiuBJVNOGKTcQ
Hd1RN0rddwwu6kFJLeKDMEXAMPLEdUF0N+qfR9yKnZQkm
+g6Q2\nGXu7u0W3hn6nLRN8qVoka0uW12p53xM7oHVz
Gf+cxKBx1b0hGkp6yCfTskUBm3Sp\n9zLw35kiHXVm4EvpwgNlnk6XcIGIkw8a/iy4pzmVUgAANY1/uU/
zgCjymw
ZT5S30\nBV0CAwEAAaNgMF4wHwYDVR0jBBgwFoAUGx0tCcU3q2n1WXAuUCv6hugXjKswHQYD
\nVR00BBYEF0VtvZ
9Aj2RYFnkX7Iu01XTRUdxgMAwGA1UdEwEB/wQMAAwDgYDVR0P\nAQH/
BAQDAgeAMA0GCSqGSIB3DQEBcWUAA4IB
AQXCQcp0tubS5ft0sDMTcP/jNX
\nDHyaRmxjpSc2aCdm7WX591TKWyAdxGAvqaDVWqTo0oXI7tZ8w7aINlGi5
pXnifx\n3SBebMUoBbTktrC97yUaeL025mCFv8emDnTR/fe7PTsBKjW0g/rrfpwBxZLXDFwN
\nnqkQjy3EDfifj2
6j0xYIqqWMPogyn4sr0CKynS5wMJuQZ1HQ0nabVwnwK4Y0Mf1p
\np9+4susFUR9aT3BT1AcIwqSpzh1Khh4Iz7ND
kRn4amsUT210jg/z0010w+BTHcVQ\nJly8XDu0CWSu04q6SnaBzHmlySIajxuRTP/AdfRouP10Xe
+q1bPOBcvVvF
8o\n-----END CERTIFICATE-----\n",
  "resourceArns": {
    "certificate": "arn:aws:iot:us-west-2:571032923833:cert/799a9ea048a1e6aea42b55EXAMPLEf8697b4bafcd77a318a3068e30404b9233c",
    "thing": "arn:aws:iot:us-west-2:571032923833:thing/Register-thing-trial-1"
  }
}
```

Para obtener más información, consulte [Aprovisionamiento por usuario de confianza](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [RegisterThing](#) la Referencia de AWS CLI comandos.

## **reject-certificate-transfer**

En el siguiente ejemplo de código, se muestra cómo usar `reject-certificate-transfer`.

### AWS CLI

Para rechazar la transferencia de un certificado

El siguiente `reject-certificate-transfer` ejemplo rechaza la transferencia del certificado del dispositivo especificado desde otra AWS cuenta.

```
aws iot reject-certificate-transfer \  
  --certificate-  
  id f0f33678c7c9a046e5cc87b2b1a58dfa0beec26db78add5e605d630e05c7fc8
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Transferir un certificado a otra cuenta](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [RejectCertificateTransfer](#) la Referencia de AWS CLI comandos.

## **remove-thing-from-billing-group**

En el siguiente ejemplo de código, se muestra cómo usar `remove-thing-from-billing-group`.

### AWS CLI

Para eliminar un elemento de un grupo de facturación

En el siguiente `remove-thing-from-billing-group` ejemplo, se elimina el elemento especificado de un grupo de facturación.

```
aws iot remove-thing-from-billing-group \  
  --billing-group-name GroupOne \  
  --thing-name ThingOne
```

```
--thing-name MyOtherLightBulb
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Grupos de facturación](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [RemoveThingFromBillingGroup](#) la Referencia de AWS CLI comandos.

## **remove-thing-from-thing-group**

En el siguiente ejemplo de código, se muestra cómo usar `remove-thing-from-thing-group`.

### AWS CLI

Para eliminar una cosa de un grupo de cosas

En el siguiente `remove-thing-from-thing-group` ejemplo, se elimina el elemento especificado de un grupo de cosas.

```
aws iot remove-thing-from-thing-group \  
  --thing-name bulb7 \  
  --thing-group-name DeadBulbs
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Thing Groups < <https://docs.aws.amazon.com/iot/latest/developerguide/thing-groups.html> > en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [RemoveThingFromThingGroup](#) la Referencia de AWS CLI comandos.

## **replace-topic-rule**

En el siguiente ejemplo de código, se muestra cómo usar `replace-topic-rule`.

### AWS CLI

Para actualizar la definición de reglas de un tema

El siguiente `replace-topic-rule` ejemplo actualiza la regla especificada para enviar una SNS alerta cuando las lecturas del nivel de humedad del suelo sean demasiado bajas.

```
aws iot replace-topic-rule \
  --rule-name MyRPIlowMoistureAlertRule \
  --topic-rule-payload "{\"sql\": \"SELECT * FROM '$aws/things/MyRPI/shadow/
update/accepted' WHERE state.reported.moisture = 'low'\", \"description\": \"Sends
an alert when soil moisture level readings are too low.\", \"actions\": [{\"sns
\": {\"targetArn\": \"arn:aws:sns:us-west-2:123456789012:MyRPIlowMoistureTopic\",
\"roleArn\": \"arn:aws:iam::123456789012:role/service-role/MyRPIlowMoistureTopicRole
\", \"messageFormat\": \"RAW\"}}], \"ruleDisabled\": false, \"awsIotSqlVersion\":
\"2016-03-23\"}"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de una regla de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ReplaceTopicRule](#) la Referencia de AWS CLI comandos.

## search-index

En el siguiente ejemplo de código, se muestra cómo usar search-index.

### AWS CLI

Para consultar el índice de cosas

En el siguiente search-index ejemplo, se AWS\_Things consultan en el índice los elementos que tienen un tipo deLightBulb.

```
aws iot search-index \
  --index-name "AWS_Things" \
  --query-string "thingTypeName:LightBulb"
```

Salida:

```
{
  "things": [
    {
      "thingName": "MyLightBulb",
      "thingId": "40da2e73-c6af-406e-b415-15acae538797",
      "thingTypeName": "LightBulb",
```

```
    "thingGroupNames": [
      "LightBulbs",
      "DeadBulbs"
    ],
    "attributes": {
      "model": "123",
      "wattage": "75"
    },
    "connectivity": {
      "connected": false
    }
  },
  {
    "thingName": "ThirdBulb",
    "thingId": "615c8455-33d5-40e8-95fd-3ee8b24490af",
    "thingTypeName": "LightBulb",
    "attributes": {
      "model": "123",
      "wattage": "75"
    },
    "connectivity": {
      "connected": false
    }
  },
  {
    "thingName": "MyOtherLightBulb",
    "thingId": "6dae0d3f-40c1-476a-80c4-1ed24ba6aa11",
    "thingTypeName": "LightBulb",
    "attributes": {
      "model": "123",
      "wattage": "75"
    },
    "connectivity": {
      "connected": false
    }
  }
]
}
```

Para obtener más información, consulte [Gestión de la indexación de cosas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [SearchIndex](#) la Referencia de AWS CLI comandos.

## set-default-authorizer

En el siguiente ejemplo de código, se muestra cómo usar `set-default-authorizer`.

### AWS CLI

Para configurar un autorizador predeterminado

El siguiente `set-default-authorizer` ejemplo establece el autorizador personalizado `CustomAuthorizer` denominado autorizador predeterminado.

```
aws iot set-default-authorizer \  
  --authorizer-name CustomAuthorizer
```

Salida:

```
{  
  "authorizerName": "CustomAuthorizer",  
  "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/  
CustomAuthorizer"  
}
```

Para obtener más información, consulte [CreateDefaultAuthorizer](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [SetDefaultAuthorizer](#) la Referencia de AWS CLI comandos.

## set-default-policy-version

En el siguiente ejemplo de código, se muestra cómo usar `set-default-policy-version`.

### AWS CLI

Para configurar la versión predeterminada de una política

En el siguiente `set-default-policy-version` ejemplo, se establece la versión predeterminada en la política denominada `UpdateDeviceCertPolicy`. 2

```
aws iot set-default-policy-version \  
  --policy-name UpdateDeviceCertPolicy \  
  --policy-version 2
```



**--policy-version-id 2**

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SetDefaultPolicyVersion](#) la Referencia de AWS CLI comandos.

**set-v2-logging-level**

En el siguiente ejemplo de código, se muestra cómo usar `set-v2-logging-level`.

## AWS CLI

Para establecer el nivel de registro de un grupo de cosas

En el siguiente `set-v2-logging-level` ejemplo, se establece el nivel de registro para registrar las advertencias del grupo de cosas especificado.

```
aws iot set-v2-logging-level \  
  --log-target "{\"targetType\":\"THING_GROUP\",\"targetName\":\"LightBulbs\"}" \  
  --log-level WARN
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SetV2 LoggingLevel](#) en la Referencia de AWS CLI comandos.

**set-v2-logging-options**

En el siguiente ejemplo de código, se muestra cómo usar `set-v2-logging-options`.

## AWS CLI

Para configurar las opciones de registro

En el siguiente `set-v2-logging-options` ejemplo, se establece el nivel de detalle de registro predeterminado `ERROR` y se especifica el que se debe utilizar ARN para el registro.

```
aws iot set-v2-logging-options \  
  --default-log-level ERROR \  
  --role-arn "arn:aws:iam::094249569039:role/service-role/iotLoggingRole"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SetV2 LoggingOptions](#) en AWS CLI la referencia de comandos.

## start-audit-mitigation-actions-task

En el siguiente ejemplo de código, se muestra cómo usar `start-audit-mitigation-actions-task`.

### AWS CLI

Para aplicar una acción de mitigación a los resultados de una auditoría

En el siguiente `start-audit-mitigation-actions-task` ejemplo, se aplica la `ResetPolicyVersionAction` acción (que borra la política) a la única constatación especificada.

```
aws iot start-audit-mitigation-actions-task \  
  --task-id "myActionsTaskId" \  
  --target "findingIds=[\"0edbaaec-2fe1-4cf5-abc9-d4c3e51f7464\"]" \  
  --audit-check-to-actions-mapping  
  "IOT_POLICY_OVERLY_PERMISSIVE_CHECK=[\"ResetPolicyVersionAction\"]" \  
  --client-request-token "adhadhahda"
```

Salida:

```
{  
  "taskId": "myActionsTaskId"  
}
```

Para obtener más información, consulte [StartAuditMitigationActionsTask \(Comandos de acción de mitigación\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [StartAuditMitigationActionsTask](#) la Referencia de AWS CLI comandos.

## start-on-demand-audit-task

En el siguiente ejemplo de código, se muestra cómo usar `start-on-demand-audit-task`.

## AWS CLI

Para iniciar una auditoría de inmediato

El siguiente `start-on-demand-audit-task` ejemplo inicia una auditoría de AWS IoT Device Defender y realiza tres comprobaciones de certificados.

```
aws iot start-on-demand-audit-task \  
  --target-check-  
  names CA_CERTIFICATE_EXPIRING_CHECK DEVICE_CERTIFICATE_EXPIRING_CHECK REVOKED_CA_CERTIFICATE
```

Salida:

```
{  
  "taskId": "a3aea009955e501a31b764abe1bebd3d"  
}
```

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [StartOnDemandAuditTask](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

## AWS CLI

Para especificar una etiqueta, clave y valor para un recurso

En el siguiente `tag-resource` ejemplo, se aplica la etiqueta con una clave `Assembly` y el valor `Fact1NW` al grupo de cosas `LightBulbs`.

```
aws iot tag-resource \  
  --tags Key=Assembly,Value="Fact1NW" \  
  --resource-arn "arn:aws:iot:us-west-2:094249569039:thinggroup/LightBulbs"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar sus recursos de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## test-authorization

En el siguiente ejemplo de código, se muestra cómo usar `test-authorization`.

### AWS CLI

Para probar sus políticas AWS de IoT

El siguiente `test-authorization` ejemplo prueba las políticas de AWS IoT asociadas al principal especificado.

```
aws iot test-authorization \  
  --auth-infos actionType=CONNECT,resources=arn:aws:iot:us-east-1:123456789012:client/client1 \  
  --principal arn:aws:iot:us-west-2:123456789012:cert/aab1068f7f43ac3e3cae4b3a8aa3f308d2a750e6350507962e32c1eb465d9775
```

Salida:

```
{  
  "authResults": [  
    {  
      "authInfo": {  
        "actionType": "CONNECT",  
        "resources": [  
          "arn:aws:iot:us-east-1:123456789012:client/client1"  
        ]  
      },  
      "allowed": {  
        "policies": [  
          {  
            "policyName": "TestPolicyAllowed",  
            "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/  
TestPolicyAllowed"  
          }  
        ]  
      }  
    }  
  ]  
}
```

```

    },
    "denied": {
      "implicitDeny": {
        "policies": [
          {
            "policyName": "TestPolicyDenied",
            "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TestPolicyDenied"
          }
        ]
      },
      "explicitDeny": {
        "policies": [
          {
            "policyName": "TestPolicyExplicitDenied",
            "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TestPolicyExplicitDenied"
          }
        ]
      }
    },
    "authDecision": "IMPLICIT_DENY",
    "missingContextValues": []
  }
]
}

```

Para obtener más información, consulte [TestAuthorization](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [TestAuthorization](#) la Referencia de AWS CLI comandos.

## test-invoke-authorizer

En el siguiente ejemplo de código, se muestra cómo usar `test-invoke-authorizer`.

### AWS CLI

Para probar su autorizador personalizado

En el siguiente `test-invoke-authorizer` ejemplo, se prueba el autorizador personalizado.

```
aws iot test-invoke-authorizer \
```

```

--authorizer-name IoTAuthorizer \
--token allow \
--token-signature "mE0GvaHqy9nER/
FdgtJX51XYEJ3b3vE7t1gEszc0TKGgLKWXtnPkb2AbKn0AZ81GyoN5dVtWDWVmr25m7+
+zjbYIMk2TBvyGXh0mvKFBPkdgyA43KL6SiZy0cTqLPMcQDsP7VX2rXr7CTowCxSNKphGXdQe0/
I5dQ+J06KUaHwCmupt0/MejKtaNwiiA064j6wpr0AUwG5S1IYFuRd0X
+wfo8pb0DubAIX1Ua705kuhRUcTx4SxUShEYKmN4IDEvLB6FsIr0B2wvB7y4iPmcajxzG102ExvyCUNctCV9dY1RRGJj"

```

Salida:

```

{
  "isAuthenticated": true,
  "principalId": "principalId",
  "policyDocuments": [
    [{"Version":"2012-10-17","Statement":
[{"Action":"iot:Publish","Effect":"Allow","Resource":"arn:aws:iot:us-
west-2:123456789012:topic/customauthtesting"}]}]
  ],
  "refreshAfterInSeconds": 600,
  "disconnectAfterInSeconds": 3600
}

```

Para obtener más información, consulte [TestInvokeAuthorizer](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [TestInvokeAuthorizer](#) la Referencia de AWS CLI comandos.

## transfer-certificate

En el siguiente ejemplo de código, se muestra cómo usar `transfer-certificate`.

### AWS CLI

Para transferir un certificado de dispositivo a una AWS cuenta diferente

En el siguiente `transfer-certificate` ejemplo, se transfiere un certificado de dispositivo a otra AWS cuenta. El certificado y la AWS cuenta se identifican mediante un identificador.

```

aws iot transfer-certificate \
--certificate-
id 488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142 \
--target-aws-account 030714055129

```

Salida:

```
{
  "transferredCertificateArn": "arn:aws:iot:us-
west-2:030714055129:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142"
}
```

Para obtener más información, consulte [Transferir un certificado a otra cuenta](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [TransferCertificate](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

AWS CLI

Para eliminar una clave de etiqueta de un recurso

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta `MyTag` y su valor del grupo de `cosasLightBulbs`.

```
command
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar sus recursos de AWS IoT](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-account-audit-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-account-audit-configuration`.

AWS CLI

Ejemplo 1: Para habilitar las notificaciones de Amazon para SNS las notificaciones de auditoría

El siguiente `update-account-audit-configuration` ejemplo habilita SNS las notificaciones de Amazon para las notificaciones de auditoría de AWS IoT Device Defender, especificando un destino y la función utilizada para escribir en ese objetivo.

```
aws iot update-account-audit-configuration \  
  --audit-notification-target-configurations "SNS={targetArn=\"arn:aws:sns:us-  
west-2:123456789012:ddaudits\",roleArn=\"arn:aws:iam::123456789012:role/service-  
role/AWSIoTDeviceDefenderAudit\",enabled=true}"
```

Este comando no genera ninguna salida.

Ejemplo 2: Para habilitar una comprobación de auditoría

El siguiente `update-account-audit-configuration` ejemplo habilita la comprobación de auditoría de AWS IoT Device Defender denominada `AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK`. No puede deshabilitar una comprobación de auditoría si forma parte de `targetCheckNames` una o más auditorías programadas para la AWS cuenta.

```
aws iot update-account-audit-configuration \  
  --audit-check-configurations  
  "{\"AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK\":{\"enabled\":true}}"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateAccountAuditConfiguration](#) la Referencia de AWS CLI comandos.

## update-audit-suppression

En el siguiente ejemplo de código, se muestra cómo usar `update-audit-suppression`.

### AWS CLI

Para actualizar la supresión de un hallazgo de auditoría

En el siguiente `update-audit-suppression` ejemplo, se actualiza la fecha de caducidad de la supresión de un hallazgo de auditoría al 21 de septiembre de 2020.



```
aws iot update-audit-suppression \  
  --check-name DEVICE_CERTIFICATE_EXPIRING_CHECK \  
  --resource-identifier deviceCertificateId=c7691e<shortened> \  
  --no-suppress-indefinitely \  
  --expiration-date 2020-09-21
```

Este comando no genera ninguna salida.

Para obtener más información, consulte las [supresiones de detección de auditorías](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte la Referencia [UpdateAuditSuppression](#) de AWS CLI comandos.

## update-authorizer

En el siguiente ejemplo de código, se muestra cómo usar update-authorizer.

### AWS CLI

Para actualizar un autorizador personalizado

En el siguiente update-authorizer ejemplo, el estado de dosCustomAuthorizer2.  
INACTIVE

```
aws iot update-authorizer \  
  --authorizer-name CustomAuthorizer2 \  
  --status INACTIVE
```

Salida:

```
{  
  "authorizerName": "CustomAuthorizer2",  
  "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/  
CustomAuthorizer2"  
}
```

Para obtener más información, consulte [UpdateAuthorizer](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [UpdateAuthorizer](#) la Referencia de AWS CLI comandos.

## update-billing-group

En el siguiente ejemplo de código, se muestra cómo usar `update-billing-group`.

### AWS CLI

Para actualizar la información sobre un grupo de facturación

En el siguiente `update-billing-group` ejemplo, se actualiza la descripción del grupo de facturación especificado.

```
aws iot update-billing-group \  
  --billing-group-name GroupOne \  
  --billing-group-properties "billingGroupDescription=\"Primary bulb billing group  
  \""
```

Salida:

```
{  
  "version": 2  
}
```

Para obtener más información, consulte [Grupos de facturación](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateBillingGroup](#) la Referencia de AWS CLI comandos.

## update-ca-certificate

En el siguiente ejemplo de código, se muestra cómo usar `update-ca-certificate`.

### AWS CLI

Para actualizar un certificado de una entidad emisora de certificados (CA)

El siguiente `update-ca-certificate` ejemplo establece el ACTIVE estado del certificado de CA especificado.

```
aws iot update-ca-certificate \  
  --certificate-  
  id f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467 \  
  --
```

```
--new-status ACTIVE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [UpdateCACertificate](#) en la APIReferencia de AWS IoT.

- Para API obtener más información, consulte [UpdateCaCertificate](#) la Referencia de AWS CLI comandos.

## update-certificate

En el siguiente ejemplo de código, se muestra cómo usar `update-certificate`.

### AWS CLI

Para actualizar un certificado de dispositivo

En el siguiente `update-certificate` ejemplo, se establece el `INACTIVE` estado del certificado del dispositivo especificado.

```
aws iot update-certificate \  
  --certificate-  
id d1eb269fb55a628552143c8f96eb3c258fcd5331ea113e766ba0c82bf225f0be \  
  --new-status INACTIVE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [UpdateCertificate](#) la APIReferencia de AWS IoT.

- Para API obtener más información, consulte [UpdateCertificate](#) la Referencia de AWS CLI comandos.

## update-custom-metric

En el siguiente ejemplo de código, se muestra cómo usar `update-custom-metric`.

### AWS CLI

Para actualizar una métrica personalizada

En el siguiente `update-custom-metric` ejemplo, se actualiza una métrica personalizada para que tenga una nueva `display-name`.

```
aws iot update-custom-metric \  
  --metric-name batteryPercentage \  
  --display-name 'remaining battery percentage on device' \  
  --region us-east-1
```

Salida:

```
{  
  "metricName": "batteryPercentage",  
  "metricArn": "arn:aws:iot:us-east-1:1234564789012:custommetric/  
batteryPercentage",  
  "metricType": "number",  
  "displayName": "remaining battery percentage on device",  
  "creationDate": "2020-11-17T23:01:35.110000-08:00",  
  "lastModifiedDate": "2020-11-17T23:02:12.879000-08:00"  
}
```

Para obtener más información, consulte [Métricas personalizadas](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [UpdateCustomMetric](#) la Referencia de AWS CLI comandos.

## update-dimension

En el siguiente ejemplo de código, se muestra cómo usar `update-dimension`.

AWS CLI

Para actualizar una dimensión

El siguiente `update-dimension` ejemplo actualiza una dimensión.

```
aws iot update-dimension \  
  --name TopicFilterForAuthMessages \  
  --string-values device/${iot:ClientId}/auth
```

Salida:

```
{  
  "name": "TopicFilterForAuthMessages",
```

```
"lastModifiedDate": 1585866222.317,  
"stringValue": [  
  "device/${iot:ClientId}/auth"  
],  
"creationDate": 1585854500.474,  
"type": "TOPIC_FILTER",  
"arn": "arn:aws:iot:us-west-2:1234564789012:dimension/  
TopicFilterForAuthMessages"  
}
```

Para obtener más información, consulte [Definir el alcance de las métricas en los perfiles de seguridad mediante dimensiones](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [UpdateDimension](#) la Referencia de AWS CLI comandos.

## update-domain-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-domain-configuration`.

### AWS CLI

Para actualizar la configuración de un dominio

El siguiente `update-domain-configuration` ejemplo deshabilita la configuración de dominio especificada.

```
aws iot update-domain-configuration \  
  --domain-configuration-name "additionalDataDomain" \  
  --domain-configuration-status "DISABLED"
```

Salida:

```
{  
  "domainConfigurationName": "additionalDataDomain",  
  "domainConfigurationArn": "arn:aws:iot:us-  
west-2:123456789012:domainconfiguration/additionalDataDomain/dikMh"  
}
```

Para obtener más información, consulte Terminales [configurables en](#) la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateDomainConfiguration](#) la Referencia de AWS CLI comandos.

## update-dynamic-thing-group

En el siguiente ejemplo de código, se muestra cómo usar `update-dynamic-thing-group`.

### AWS CLI

Para actualizar un grupo de cosas dinámico

En el siguiente `update-dynamic-thing-group` ejemplo, se actualiza el grupo de cosas dinámico especificado. Proporciona una descripción y actualiza la cadena de consulta para cambiar los criterios de pertenencia al grupo.

```
aws iot update-dynamic-thing-group \  
  --thing-group-name "RoomTooWarm" \  
  --thing-group-properties "thingGroupDescription=\"This thing group contains  
rooms warmer than 65F.\"\" \  
  --query-string "attributes.temperature>65"
```

Salida:

```
{  
  "version": 2  
}
```

Para obtener más información, consulte [Grupos de cosas dinámicos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateDynamicThingGroup](#) la Referencia de AWS CLI comandos.

## update-event-configurations

En el siguiente ejemplo de código, se muestra cómo usar `update-event-configurations`.

### AWS CLI

Para mostrar qué tipos de eventos se publican

El siguiente `update-event-configurations` ejemplo actualiza la configuración para habilitar los mensajes cuando se agrega, actualiza o elimina el certificado de CA.

```
aws iot update-event-configurations \  
  --event-configurations "{\"CA_CERTIFICATE\":{\"Enabled\":true}}"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Mensajes de eventos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateEventConfigurations](#) la Referencia de AWS CLI comandos.

## update-indexing-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-indexing-configuration`.

### AWS CLI

Para habilitar la indexación de cosas

El siguiente `update-indexing-configuration` ejemplo permite que la indexación de cosas permita buscar datos de registro, datos ocultos y estado de conectividad de cosas mediante el índice `AWS_Things`.

```
aws iot update-indexing-configuration \  
  --thing-indexing-configuration thingIndexingMode=REGISTRY_AND_SHADOW,thingConnectivityIndexingMode=STATUS
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión de la indexación de cosas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateIndexingConfiguration](#) la Referencia de AWS CLI comandos.

## update-job

En el siguiente ejemplo de código, se muestra cómo usar `update-job`.

## AWS CLI

Para obtener el estado detallado de un trabajo

En el siguiente `update-job` ejemplo se obtiene el estado detallado del trabajo cuyo identificador es `example-job-01`.

```
aws iot describe-job \  
  --job-id "example-job-01"
```

Salida:

```
{  
  "job": {  
    "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-01",  
    "jobId": "example-job-01",  
    "targetSelection": "SNAPSHOT",  
    "status": "IN_PROGRESS",  
    "targets": [  
      "arn:aws:iot:us-west-2:123456789012:thing/MyRaspberryPi"  
    ],  
    "description": "example job test",  
    "presignedUrlConfig": {},  
    "jobExecutionsRolloutConfig": {},  
    "createdAt": 1560787022.733,  
    "lastUpdatedAt": 1560787026.294,  
    "jobProcessDetails": {  
      "numberOfCanceledThings": 0,  
      "numberOfSucceededThings": 0,  
      "numberOfFailedThings": 0,  
      "numberOfRejectedThings": 0,  
      "numberOfQueuedThings": 1,  
      "numberOfInProgressThings": 0,  
      "numberOfRemovedThings": 0,  
      "numberOfTimedOutThings": 0  
    },  
    "timeoutConfig": {}  
  }  
}
```

Para obtener más información, consulte [Creación y administración de trabajos \(CLI\)](#) en la Guía para desarrolladores de AWS IoT.



- Para API obtener más información, consulte [UpdateJob](#) la Referencia de AWS CLI comandos.

## update-mitigation-action

En el siguiente ejemplo de código, se muestra cómo usar `update-mitigation-action`.

### AWS CLI

Para actualizar una acción de mitigación

En el siguiente `update-mitigation-action` ejemplo, se actualiza el nombre de la acción de mitigación especificada `AddThingsToQuarantineGroupAction`, se cambia el nombre del grupo de cosas y se establece `overrideDynamicGroups` en `false`. Puede comprobar los cambios mediante el `describe-mitigation-action` comando.

```
aws iot update-mitigation-action \  
  --cli-input-json "{ \"actionName\": \"AddThingsToQuarantineGroupAction\",  
  \"actionParams\": { \"addThingsToThingGroupParams\": {\"thingGroupNames\":  
  [\"QuarantineGroup2\"],\"overrideDynamicGroups\": false}}}"
```

Salida:

```
{  
  "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/  
  AddThingsToQuarantineGroupAction",  
  "actionId": "2fd2726d-98e1-4abf-b10f-09465ccd6bfa"  
}
```

Para obtener más información, consulte [UpdateMitigationAction \(Comandos de acción de mitigación\)](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateMitigationAction](#) la Referencia de AWS CLI comandos.

## update-provisioning-template

En el siguiente ejemplo de código, se muestra cómo usar `update-provisioning-template`.

### AWS CLI

Para actualizar una plantilla de aprovisionamiento

En el siguiente `update-provisioning-template` ejemplo, se modifican la descripción y la función `arn` de la plantilla de aprovisionamiento especificada y se habilita la plantilla.

```
aws iot update-provisioning-template \  
  --template-name widget-template \  
  --enabled \  
  --description "An updated provisioning template for widgets" \  
  --provisioning-role-arn arn:aws:iam::504350838278:role/Provision_role
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [AWS IoT Secure Tunneling](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte la Referencia [UpdateProvisioningTemplate](#) de AWS CLI comandos.

## update-role-alias

En el siguiente ejemplo de código, se muestra cómo usar `update-role-alias`.

### AWS CLI

Para actualizar un alias de rol

En el siguiente `update-role-alias` ejemplo, se actualiza el alias del `LightBulbRole` rol.

```
aws iot update-role-alias \  
  --role-alias LightBulbRole \  
  --role-arn arn:aws:iam::123456789012:role/lightbulbrole-001
```

Salida:

```
{  
  "roleAlias": "LightBulbRole",  
  "roleAliasArn": "arn:aws:iot:us-west-2:123456789012:rolealias/LightBulbRole"  
}
```

Para obtener más información, consulte [UpdateRoleAlias](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [UpdateRoleAlias](#) la Referencia de AWS CLI comandos.

## update-scheduled-audit

En el siguiente ejemplo de código, se muestra cómo usar `update-scheduled-audit`.

### AWS CLI

Para actualizar una definición de auditoría programada

El siguiente `update-scheduled-audit` ejemplo cambia los nombres de las comprobaciones de destino para una auditoría programada de AWS IoT Device Defender.

```
aws iot update-scheduled-audit \  
  --scheduled-audit-name WednesdayCertCheck \  
  --target-check-  
names CA_CERTIFICATE_EXPIRING_CHECK DEVICE_CERTIFICATE_EXPIRING_CHECK REVOKED_CA_CERTIFICATE
```

Salida:

```
{  
  "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/  
WednesdayCertCheck"  
}
```

Para obtener más información, consulte [Audit Commands](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateScheduledAudit](#) la Referencia de AWS CLI comandos.

## update-security-profile

En el siguiente ejemplo de código, se muestra cómo usar `update-security-profile`.

### AWS CLI

Para cambiar un perfil de seguridad

El siguiente `update-security-profile` ejemplo actualiza la descripción y los comportamientos de un perfil de seguridad de AWS IoT Device Defender.

```
aws iot update-security-profile \  

```

```

--security-profile-name PossibleIssue \
--security-profile-description "Check to see if authorization fails 12 times in
5 minutes or if cellular bandwidth exceeds 128" \
--behaviors "[{"name\":\"CellularBandwidth\",\"metric\":\"aws:message-byte-size
\",\"criteria\":{\"comparisonOperator\":\"greater-than\",\"value\":{\"count\":128},
\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}}, {"name
\":\"Authorization\",\"metric\":\"aws:num-authorization-failures\",\"criteria\":{\"
comparisonOperator\":\"less-than\",\"value\":{\"count\":12},\"durationSeconds
\":300,\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}}]"

```

Salida:

```

{
  "securityProfileName": "PossibleIssue",
  "securityProfileArn": "arn:aws:iot:us-west-2:123456789012:securityprofile/
PossibleIssue",
  "securityProfileDescription": "check to see if authorization fails 12 times in 5
minutes or if cellular bandwidth exceeds 128",
  "behaviors": [
    {
      "name": "CellularBandwidth",
      "metric": "aws:message-byte-size",
      "criteria": {
        "comparisonOperator": "greater-than",
        "value": {
          "count": 128
        },
        "consecutiveDatapointsToAlarm": 1,
        "consecutiveDatapointsToClear": 1
      }
    },
    {
      "name": "Authorization",
      "metric": "aws:num-authorization-failures",
      "criteria": {
        "comparisonOperator": "less-than",
        "value": {
          "count": 12
        },
        "durationSeconds": 300,
        "consecutiveDatapointsToAlarm": 1,
        "consecutiveDatapointsToClear": 1
      }
    }
  ]
}

```

```
    }
  ],
  "version": 2,
  "creationDate": 1560278102.528,
  "lastModifiedDate": 1560352711.207
}
```

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateSecurityProfile](#) la Referencia de AWS CLI comandos.

## update-stream

En el siguiente ejemplo de código, se muestra cómo usar `update-stream`.

### AWS CLI

Para actualizar una transmisión

En el siguiente `update-stream` ejemplo, se actualiza una transmisión existente. La versión de transmisión se incrementa en uno.

```
aws iot update-stream \
  --cli-input-json file://update-stream.json
```

Contenidos de `update-stream.json`:

```
{
  "streamId": "stream12345",
  "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
  "files": [
    {
      "fileId": 123,
      "s3Location": {
        "bucket": "codesign-ota-bucket",
        "key": "48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
      }
    }
  ]
}
```

```
"roleArn": "arn:aws:iam:us-west-2:123456789012:role/service-role/
my_ota_stream_role"
}
```

Salida:

```
{
  "streamId": "stream12345",
  "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream12345",
  "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
  "streamVersion": 2
}
```

Para obtener más información, consulte [UpdateStream](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [UpdateStream](#) la Referencia de AWS CLI comandos.

## update-thing-group

En el siguiente ejemplo de código, se muestra cómo usar `update-thing-group`.

### AWS CLI

Para actualizar la definición de un grupo de cosas

En el siguiente `update-thing-group` ejemplo, se actualiza la definición del grupo de cosas especificado y se cambian la descripción y dos atributos.

```
aws iot update-thing-group \
  --thing-group-name HalogenBulbs \
  --thing-group-properties "thingGroupDescription=\"Halogen bulb group\",
attributePayload={attributes={Manufacturer=AnyCompany,wattage=60}}"
```

Salida:

```
{
  "version": 2
}
```

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateThingGroup](#) la Referencia de AWS CLI comandos.

## update-thing-groups-for-thing

En el siguiente ejemplo de código, se muestra cómo usar `update-thing-groups-for-thing`.

### AWS CLI

Para cambiar los grupos a los que pertenece una cosa

En el siguiente `update-thing-groups-for-thing` ejemplo, se `MyLightBulb` quita la cosa nombrada del grupo nombrado `DeadBulbs` y se agrega al grupo nombrado `replaceableItems` al mismo tiempo.

```
aws iot update-thing-groups-for-thing \  
  --thing-name MyLightBulb \  
  --thing-groups-to-add "replaceableItems" \  
  --thing-groups-to-remove "DeadBulbs"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Thing Groups](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateThingGroupsForThing](#) la Referencia de AWS CLI comandos.

## update-thing

En el siguiente ejemplo de código, se muestra cómo usar `update-thing`.

### AWS CLI

Para asociar una cosa a un tipo de cosa

El siguiente `update-thing` ejemplo asocia una cosa del registro de AWS IoT a un tipo de cosa. Al realizar la asociación, se proporcionan valores para los atributos definidos por el tipo de cosa.

```
aws iot update-thing \  
  --thing-name MyLightBulb \  
  --thing-type MyLightBulb \  
  --thing-attributes { "color": "red", "brightness": 100 }
```

```
--thing-name "MyOtherLightBulb" \  
--thing-type-name "LightBulb" \  
--attribute-payload '{"attributes": {"wattage": "75", "model": "123"}}'
```

Este comando no produce resultados. Utilice el `describe-thing` comando para ver el resultado.

Para obtener más información, consulte [Tipos de cosas](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateThing](#) la Referencia de AWS CLI comandos.

## update-topic-rule-destination

En el siguiente ejemplo de código, se muestra cómo usar `update-topic-rule-destination`.

### AWS CLI

Ejemplo 1: Para habilitar el destino de una regla temática

El siguiente `update-topic-rule-destination` ejemplo habilita el tráfico hacia el destino de una regla temática.

```
aws iot update-topic-rule-destination \  
  --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/  
a1b2c3d4-5678-90ab-cdef-1111EXAMPLE" \  
  --status ENABLED
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Habilitar el destino de una regla temática](#) en la Guía para desarrolladores de AWS IoT.

Ejemplo 2: Para deshabilitar el destino de una regla temática

El siguiente `update-topic-rule-destination` ejemplo deshabilita el tráfico hacia el destino de una regla temática.

```
aws iot update-topic-rule-destination \  
  --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/  
a1b2c3d4-5678-90ab-cdef-1111EXAMPLE" \  
  --status DISABLED
```



```
--status DISABLED
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Deshabilitar el destino de una regla temática](#) en la Guía para desarrolladores de AWS IoT.

Ejemplo 3: Para enviar un nuevo mensaje de confirmación

El siguiente `update-topic-rule-destination` ejemplo envía un nuevo mensaje de confirmación para el destino de una regla temática.

```
aws iot update-topic-rule-destination \  
  --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE" \  
  --status IN_PROGRESS
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Enviar un nuevo mensaje de confirmación](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateTopicRuleDestination](#) la Referencia de AWS CLI comandos.

## **validate-security-profile-behaviors**

En el siguiente ejemplo de código, se muestra cómo usar `validate-security-profile-behaviors`.

### AWS CLI

Ejemplo 1: Para validar los parámetros de comportamiento de un perfil de seguridad

El siguiente `validate-security-profile-behaviors` ejemplo valida un conjunto de comportamientos correcto y bien formado para un perfil de seguridad de AWS IoT Device Defender.

```
aws iot validate-security-profile-behaviors \  
  --behaviors "[{\\"name\\":\\"CellularBandwidth\\",\\"metric\\":\\"aws:message-byte-size\\",\\"criteria\\":{\\"comparisonOperator\\":\\"greater-than\\",\\"value\\":{\\"count\\":128},
```

```
\\"consecutiveDatapointsToAlarm\\":1,\\"consecutiveDatapointsToClear\\":1}},{\\"name
\\":\\"Authorization\\",\\"metric\\":\\"aws:num-authorization-failures\\",\\"criteria\\":
{\\"comparisonOperator\\":\\"greater-than\\",\\"value\\":{\\"count\\":12},\\"durationSeconds
\\":300,\\"consecutiveDatapointsToAlarm\\":1,\\"consecutiveDatapointsToClear\\":1}}]"
```

Salida:

```
{
  "valid": true,
  "validationErrors": []
}
```

Ejemplo 2: Para validar los parámetros de comportamiento incorrectos de un perfil de seguridad

El siguiente `validate-security-profile-behaviors` ejemplo valida un conjunto de comportamientos que contienen un error para un perfil de seguridad de AWS IoT Device Defender.

```
aws iot validate-security-profile-behaviors \
  --behaviors "[{\\"name\\":\\"CellularBandwidth\\",\\"metric\\":\\"aws:message-byte-size
\\",\\"criteria\\":{\\"comparisonOperator\\":\\"greater-than\\",\\"value\\":{\\"count\\":128},
\\"consecutiveDatapointsToAlarm\\":1,\\"consecutiveDatapointsToClear\\":1}},{\\"name
\\":\\"Authorization\\",\\"metric\\":\\"aws:num-authorization-failures\\",\\"criteria\\":
{\\"comparisonOperator\\":\\"greater-than\\",\\"value\\":{\\"count\\":12},\\"durationSeconds
\\":300,\\"consecutiveDatapointsToAlarm\\":100000,\\"consecutiveDatapointsToClear
\\":1}}]"
```

Salida:

```
{
  "valid": false,
  "validationErrors": [
    {
      "errorMessage": "Behavior Authorization is malformed.
consecutiveDatapointsToAlarm 100000 should be in range[1,10]"
    }
  ]
}
```

Para obtener más información, consulte [Detectar comandos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ValidateSecurityProfileBehaviors](#) la Referencia de AWS CLI comandos.

## AWS IoT 1-Click Ejemplos de dispositivos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante el uso de AWS Command Line Interface with AWS IoT 1-Click Devices.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **claim-devices-by-claim-code**

En el siguiente ejemplo de código, se muestra cómo usar `claim-devices-by-claim-code`.

AWS CLI

Para reclamar uno o más dispositivos AWS IoT 1-Click mediante un código promocional

En el siguiente `claim-devices-by-claim-code` ejemplo, se reivindica el dispositivo AWS IoT 1-Click especificado mediante un código promocional (en lugar de un ID de dispositivo).

```
aws iot1click-devices claim-devices-by-claim-code \  
  --claim-code C-123EXAMPLE
```

Salida:

```
{  
  "Total": 9  
  "ClaimCode": "C-123EXAMPLE"  
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [ClaimDevicesByClaimCode](#) de AWS CLI comandos.

## describe-device

En el siguiente ejemplo de código, se muestra cómo usar describe-device.

### AWS CLI

Para describir un dispositivo

El siguiente describe-device ejemplo describe el dispositivo especificado.

```
aws iot1click-devices describe-device \  
  --device-id G030PM0123456789
```

Salida:

```
{  
  "DeviceDescription": {  
    "Arn": "arn:aws:iot1click:us-west-2:012345678901:devices/G030PM0123456789",  
    "Attributes": {  
      "projectRegion": "us-west-2",  
      "projectName": "AnytownDumpsters",  
      "placementName": "customer217",  
      "deviceTemplateName": "empty-dumpster-request"  
    },  
    "DeviceId": "G030PM0123456789",  
    "Enabled": false,  
    "RemainingLife": 99.9,  
    "Type": "button",  
    "Tags": {}  
  }  
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [DescribeDevice](#) de AWS CLI comandos.

## finalize-device-claim

En el siguiente ejemplo de código, se muestra cómo usar `finalize-device-claim`.

### AWS CLI

Para finalizar una solicitud de reclamación para un dispositivo AWS IoT 1-Click utilizando un ID de dispositivo

El siguiente `finalize-device-claim` ejemplo finaliza una solicitud de reclamación para el dispositivo AWS IoT 1-Click especificado mediante un ID de dispositivo (en lugar de un código promocional).

```
aws iot1click-devices finalize-device-claim \  
  --device-id G030PM0123456789
```

Salida:

```
{  
  "State": "CLAIMED"  
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [FinalizeDeviceClaim](#) de AWS CLI comandos.

## get-device-methods

En el siguiente ejemplo de código, se muestra cómo usar `get-device-methods`.

### AWS CLI

Para ver una lista de los métodos disponibles para un dispositivo

En el siguiente `get-device-methods` ejemplo, se enumeran los métodos disponibles para un dispositivo.

```
aws iot1click-devices get-device-methods \  
  --device-id G030PM0123456789
```

**Salida:**

```
{
  "DeviceMethods": [
    {
      "MethodName": "getDeviceHealthParameters"
    },
    {
      "MethodName": "setDeviceHealthMonitorCallback"
    },
    {
      "MethodName": "getDeviceHealthMonitorCallback"
    },
    {
      "MethodName": "setOnClickCallback"
    },
    {
      "MethodName": "getOnClickCallback"
    }
  ]
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [GetDeviceMethods](#) de AWS CLI comandos.

**initiate-device-claim**

En el siguiente ejemplo de código, se muestra cómo usar `initiate-device-claim`.

**AWS CLI**

Para iniciar una solicitud de reclamación para un dispositivo AWS IoT 1-Click utilizando un ID de dispositivo

En el siguiente `initiate-device-claim` ejemplo, se inicia una solicitud de reclamación para el dispositivo AWS IoT 1-Click especificado mediante un ID de dispositivo (en lugar de un código promocional).

```
aws iot1click-devices initiate-device-claim \
```

```
--device-id G030PM0123456789
```

Salida:

```
{  
  "State": "CLAIM_INITIATED"  
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [InitiateDeviceClaim](#) de AWS CLI comandos.

## invoke-device-method

En el siguiente ejemplo de código, se muestra cómo usar `invoke-device-method`.

AWS CLI

Para invocar un método de dispositivo en un dispositivo

El siguiente `invoke-device-method` ejemplo invoca el método especificado en un dispositivo.

```
aws iot1click-devices invoke-device-method \  
  --cli-input-json file://invoke-device-method.json
```

Contenidos de `invoke-device-method.json`:

```
{  
  "DeviceId": "G030PM0123456789",  
  "DeviceMethod": {  
    "DeviceType": "device",  
    "MethodName": "getDeviceHealthParameters"  
  }  
}
```

Salida:

```
{
```

```
"DeviceMethodResponse": "{\\"remainingLife\\": 99.8}"
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [InvokeDeviceMethod](#) de AWS CLI comandos.

## list-device-events

En el siguiente ejemplo de código, se muestra cómo usar `list-device-events`.

### AWS CLI

Para enumerar los eventos de un dispositivo durante un intervalo de tiempo específico

En el siguiente `list-device-events` ejemplo, se enumeran los eventos del dispositivo especificado para el intervalo de tiempo especificado.

```
aws iot1click-devices list-device-events \
  --device-id G030PM0123456789 \
  --from-time-stamp 2019-07-17T15:45:12.880Z --to-time-
stamp 2019-07-19T15:45:12.880Z
```

Salida:

```
{
  "Events": [
    {
      "Device": {
        "Attributes": {},
        "DeviceId": "G030PM0123456789",
        "Type": "button"
      },
      "StdEvent": "{\\"clickType\\": \\"SINGLE\\",
\\"reportedTime\\": \\"2019-07-18T23:47:55.015Z\\", \\"certificateId\\":
\\"fe8798a6c97c62ef8756b80eeefdcf2280f3352f82faa8080c74cc4f4a4d1811\\",
\\"remainingLife\\": 99.85000000000001, \\"testMode\\": false}"
    },
    {
      "Device": {
```



```

        "Attributes": {},
        "DeviceId": "G030PM0123456789",
        "Type": "button"
    },
    "StdEvent": "{\"clickType\": \"DOUBLE\",
\"reportedTime\": \"2019-07-19T00:14:41.353Z\", \"certificateId\":
\"fe8798a6c97c62ef8756b80eeefdcf2280f3352f82faa8080c74cc4f4a4d1811\",
\"remainingLife\": 99.8, \"testMode\": false}"
    }
]
}

```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [ListDeviceEvents](#) de AWS CLI comandos.

## list-devices

En el siguiente ejemplo de código, se muestra cómo usar `list-devices`.

### AWS CLI

Para enumerar los dispositivos de un tipo específico

En el siguiente `list-devices` ejemplo, se enumeran los dispositivos de un tipo específico.

```

aws iot1click-devices list-devices \
  --device-type button

```

Este comando no genera ninguna salida.

Salida:

```

{
  "Devices": [
    {
      "remainingLife": 99.9,
      "attributes": {
        "arn": "arn:aws:iot1click:us-west-2:123456789012:devices/
G030PM0123456789",

```

```
        "type": "button",
        "deviceId": "G030PM0123456789",
        "enabled": false
      }
    ]
  }
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [ListDevices](#) de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un dispositivo

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas del dispositivo especificado.

```
aws iot1click-devices list-tags-for-resource \
  --resource-arn "arn:aws:iot1click:us-west-2:012345678901:devices/
G030PM0123456789"
```

Salida:

```
{
  "Tags": {
    "Driver Phone": "123-555-0199",
    "Driver": "Jorge Souza"
  }
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [ListTagsForResource](#) de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir etiquetas a un AWS recurso de dispositivo

En el siguiente `tag-resource` ejemplo, se agregan dos etiquetas al recurso especificado.

```
aws iot1click-devices tag-resource \  
  --cli-input-json file://devices-tag-resource.json
```

Contenidos de `devices-tag-resource.json`:

```
{  
  "ResourceArn": "arn:aws:iot1click:us-west-2:123456789012:devices/  
G030PM0123456789",  
  "Tags": {  
    "Driver": "Jorge Souza",  
    "Driver Phone": "123-555-0199"  
  }  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [TagResource](#) de AWS CLI comandos.

## unclaim-device

En el siguiente ejemplo de código, se muestra cómo usar `unclaim-device`.

### AWS CLI

Para anular la reclamación (anular el registro) de un dispositivo de tu cuenta AWS

El siguiente `unclaim-device` ejemplo anula la reclamación (anula el registro) del dispositivo especificado de tu cuenta. AWS

```
aws iot1click-devices unclaim-device \  
  --device-id G030PM0123456789
```

Salida:

```
{  
  "State": "UNCLAIMED"  
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [UnclaimDevice](#) de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

AWS CLI

Para eliminar etiquetas de un AWS recurso de dispositivo

En el siguiente `untag-resource` ejemplo, se eliminan las etiquetas con los nombres `Driver Phone` y `Driver` del recurso de dispositivo especificado.

```
aws iot1click-devices untag-resource \  
  --resource-arn "arn:aws:iot1click:us-west-2:123456789012:projects/  
AnytownDumpsters" \  
  --tag-keys "Driver Phone" "Driver"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [UntagResource](#) de AWS CLI comandos.

## update-device-state

En el siguiente ejemplo de código, se muestra cómo usar `update-device-state`.

### AWS CLI

Para actualizar el estado ``activado`` de un dispositivo

A continuación, se `update-device-state` establece el estado del dispositivo especificado en `enabled`

```
aws iot1click-devices update-device-state \  
  --device-id G030PM0123456789 \  
  --enabled
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [UpdateDeviceState](#) de AWS CLI comandos.

## AWS IoT 1-Click Ejemplos de proyectos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface with AWS IoT 1-Click Projects.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **associate-device-with-placement**

En el siguiente ejemplo de código, se muestra cómo usar `associate-device-with-placement`.

#### AWS CLI

Para asociar un dispositivo AWS IoT 1-Click a una ubicación existente

El siguiente `associate-device-with-placement` ejemplo asocia el dispositivo AWS IoT 1-Click especificado a una ubicación existente.

```
aws iot1click-projects associate-device-with-placement \  
  --project-name AnytownDumpsters \  
  --placement-name customer217 \  
  --device-template-name empty-dumpster-request \  
  --device-id G030PM0123456789
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [AssociateDeviceWithPlacement](#) de AWS CLI comandos.

### **create-placement**

En el siguiente ejemplo de código, se muestra cómo usar `create-placement`.

#### AWS CLI

Para crear una ubicación de AWS IoT en 1-Click para un proyecto

En el siguiente `create-placement` ejemplo, se crea una ubicación de AWS IoT 1-Click para el proyecto especificado.

```
aws iot1click-projects create-placement \  
  --project-name AnytownDumpsters \  
  --placement-name customer217 \  
  --device-id G030PM0123456789
```

```
--attributes "{\"location\": \"123 Any Street Anytown, USA 10001\", \"phone\": \"123-456-7890\"}"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [CreatePlacement](#) de AWS CLI comandos.

## create-project

En el siguiente ejemplo de código, se muestra cómo usar create-project.

### AWS CLI

Para crear un proyecto de 1-Click de AWS IoT para cero o más ubicaciones

En el siguiente create-project ejemplo, se crea un proyecto de 1-Click de AWS IoT para una ubicación.

```
as iot1click-projects create-project: file: //create-project.json cli-input-json
```

Contenidos de create-project.json:

```
{
  "projectName": "AnytownDumpsters",
  "description": "All dumpsters in the Anytown region.",
  "placementTemplate": {
    "defaultAttributes": {
      "City" : "Anytown"
    },
    "deviceTemplates": {
      "empty-dumpster-request" : {
        "deviceType": "button"
      }
    }
  }
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [CreateProject](#) de AWS CLI comandos.

## delete-placement

En el siguiente ejemplo de código, se muestra cómo usar delete-placement.

### AWS CLI

Para eliminar una ubicación de un proyecto

En el siguiente delete-placement ejemplo, se elimina la ubicación especificada de un proyecto.

```
aws iot1click-projects delete-placement \  
  --project-name AnytownDumpsters \  
  --placement-name customer217
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [DeletePlacement](#) de AWS CLI comandos.

## delete-project

En el siguiente ejemplo de código, se muestra cómo usar delete-project.

### AWS CLI

Para eliminar un proyecto de tu AWS cuenta

En el siguiente delete-project ejemplo, se elimina el proyecto especificado de tu AWS cuenta.

```
aws iot1click-projects delete-project \  
  --project-name AnytownDumpsters
```



```
--project-name AnytownDumpsters
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [DeleteProject](#) de AWS CLI comandos.

## describe-placement

En el siguiente ejemplo de código, se muestra cómo usar `describe-placement`.

AWS CLI

Para describir la ubicación de un proyecto

En el siguiente `describe-placement` ejemplo, se describe una ubicación para el proyecto especificado.

```
aws iot1click-projects describe-placement \  
  --project-name AnytownDumpsters \  
  --placement-name customer217
```

Salida:

```
{  
  "placement": {  
    "projectName": "AnytownDumpsters",  
    "placementName": "customer217",  
    "attributes": {  
      "phone": "123-555-0110",  
      "location": "123 Any Street Anytown, USA 10001"  
    },  
    "createdDate": 1563488454,  
    "updatedAt": 1563488454  
  }  
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [DescribePlacement](#) de AWS CLI comandos.

## describe-project

En el siguiente ejemplo de código, se muestra cómo usar `describe-project`.

### AWS CLI

Para describir un proyecto de AWS IoT 1-Click

El siguiente `describe-project` ejemplo describe el proyecto AWS IoT 1-Click especificado.

```
aws iot1click-projects describe-project \  
  --project-name AnytownDumpsters
```

Salida:

```
{  
  "project": {  
    "arn": "arn:aws:iot1click:us-west-2:012345678901:projects/AnytownDumpsters",  
    "projectName": "AnytownDumpsters",  
    "description": "All dumpsters in the Anytown region.",  
    "createdDate": 1563483100,  
    "updatedAt": 1563483100,  
    "placementTemplate": {  
      "defaultAttributes": {  
        "City": "Anytown"  
      },  
      "deviceTemplates": {  
        "empty-dumpster-request": {  
          "deviceType": "button",  
          "callbackOverrides": {}  
        }  
      }  
    },  
    "tags": {}  
  }  
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [DescribeProject](#) de AWS CLI comandos.

## **disassociate-device-from-placement**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-device-from-placement`.

### AWS CLI

Para desasociar un dispositivo de una ubicación

El siguiente `disassociate-device-from-placement` ejemplo disocia el dispositivo especificado de una ubicación.

```
aws iot1click-projects disassociate-device-from-placement \  
  --project-name AnytownDumpsters \  
  --placement-name customer217 \  
  --device-template-name empty-dumpster-request
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [DisassociateDeviceFromPlacement](#) de AWS CLI comandos.

## **get-devices-in-placement**

En el siguiente ejemplo de código, se muestra cómo usar `get-devices-in-placement`.

### AWS CLI

Para enumerar todos los dispositivos de una ubicación incluida en un proyecto

En el siguiente `get-devices-in-placement` ejemplo, se enumeran todos los dispositivos en la ubicación especificada incluida en el proyecto especificado.

```
aws iot1click-projects get-devices-in-placement \  
  --placement-name customer217
```

```
--project-name AnytownDumpsters \  
--placement-name customer217
```

Salida:

```
{  
  "devices": {  
    "empty-dumpster-request": "G030PM0123456789"  
  }  
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [GetDevicesInPlacement](#) de AWS CLI comandos.

## list-placements

En el siguiente ejemplo de código, se muestra cómo usar `list-placements`.

AWS CLI

Para enumerar todas las ubicaciones de AWS IoT 1-Click de un proyecto

En el siguiente `list-placements` ejemplo, se enumeran todas las ubicaciones de AWS IoT 1-Click para el proyecto especificado.

```
aws iot1click-projects list-placements \  
--project-name AnytownDumpsters
```

Salida:

```
{  
  "placements": [  
    {  
      "projectName": "AnytownDumpsters",  
      "placementName": "customer217",  
      "createdDate": 1563488454,  
      "updatedAt": 1563488454  
    }  
  ]  
}
```

```
]
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [ListPlacements](#) de AWS CLI comandos.

## list-projects

En el siguiente ejemplo de código, se muestra cómo usar `list-projects`.

### AWS CLI

Para enumerar todos los proyectos AWS de IoT 1-Click

En el siguiente `list-projects` ejemplo, se enumeran todos los proyectos de AWS IoT 1-Click de tu cuenta.

```
aws iot1click-projects list-projects
```

Salida:

```
{
  "projects": [
    {
      "arn": "arn:aws:iot1click:us-west-2:012345678901:projects/
AnytownDumpsters",
      "projectName": "AnytownDumpsters",
      "createdDate": 1563483100,
      "updatedAt": 1563483100,
      "tags": {}
    }
  ]
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [ListProjects](#) de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un recurso de proyecto

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas del recurso de proyecto especificado.

```
aws iot1click-projects list-tags-for-resource \  
  --resource-arn "arn:aws:iot1click:us-west-2:123456789012:projects/  
  AnytownDumpsters"
```

Salida:

```
{  
  "tags": {  
    "Manager": "Li Juan",  
    "Account": "45215"  
  }  
}
```

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [ListTagsForResource](#) de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir etiquetas a un recurso de proyecto

En el siguiente `tag-resource` ejemplo, se agregan dos etiquetas al recurso de proyecto especificado.

```
aws iot1click-projects tag-resource \  
  --resource-arn "arn:aws:iot1click:us-west-2:123456789012:projects/  
  AnytownDumpsters"
```

```
--cli-input-json file://devices-tag-resource.json
```

Contenidos de `devices-tag-resource.json`:

```
{
  "resourceArn": "arn:aws:iot1click:us-west-2:123456789012:projects/
AnytownDumpsters",
  "tags": {
    "Account": "45215",
    "Manager": "Li Juan"
  }
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [TagResource](#) de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar etiquetas de un recurso del proyecto

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta con el nombre `Manager` de clave del proyecto especificado.

```
aws iot1click-projects untag-resource \  
  --resource-arn "arn:aws:iot1click:us-west-2:123456789012:projects/  
AnytownDumpsters" \  
  --tag-keys "Manager"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [UntagResource](#) de AWS CLI comandos.

## update-placement

En el siguiente ejemplo de código, se muestra cómo usar `update-placement`.

### AWS CLI

Para actualizar los pares clave-valor de los «atributos» de una ubicación

En el siguiente `update-placement` ejemplo, se actualizan los pares clave-valor de «atributos» de una ubicación.

```
aws iot1click-projects update-placement \  
  --cli-input-json file://update-placement.json
```

Contenidos de `update-placement.json`:

```
{  
  "projectName": "AnytownDumpsters",  
  "placementName": "customer217",  
  "attributes": {  
    "phone": "123-456-7890",  
    "location": "123 Any Street Anytown, USA 10001"  
  }  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [UpdatePlacement](#) de AWS CLI comandos.

## update-project

En el siguiente ejemplo de código, se muestra cómo usar `update-project`.



## AWS CLI

Para actualizar la configuración de un proyecto

En el siguiente `update-project` ejemplo, se actualiza la descripción de un proyecto.

```
aws iot1click-projects update-project \  
  --project-name AnytownDumpsters \  
  --description "All dumpsters (yard waste, recycling, garbage) in the Anytown  
region."
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de AWS IoT 1-Click con la Guía](#) para AWS CLI desarrolladores de IoT AWS 1-Click.

- Para API obtener más información, consulte la Referencia [UpdateProject](#) de AWS CLI comandos.

## AWS IoT Analytics ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS IoT Analytics.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **batch-put-message**

En el siguiente ejemplo de código, se muestra cómo usar `batch-put-message`.

## AWS CLI

Para enviar un mensaje a un canal

El siguiente `batch-put-message` ejemplo envía un mensaje al canal especificado.

```
aws iotanalytics batch-put-message \  
  --cli-binary-format raw-in-base64-out \  
  --cli-input-json file://batch-put-message.json
```

Contenidos de `batch-put-message.json`:

```
{  
  "channelName": "mychannel",  
  "messages": [  
    {  
      "messageId": "0001",  
      "payload": "eyAidGVtcGVyYXR1cmUiOiAyMCB9"  
    }  
  ]  
}
```

Salida:

```
{  
  "batchPutMessageErrorEntries": []  
}
```

Para obtener más información, consulte [BatchPutMessage](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [BatchPutMessage](#) la Referencia de AWS CLI comandos.

## **cancel-pipeline-reprocessing**

En el siguiente ejemplo de código, se muestra cómo usar `cancel-pipeline-reprocessing`.

## AWS CLI

Para cancelar el reprocesamiento de datos a través de una canalización

El siguiente `cancel-pipeline-reprocessing` ejemplo cancela el reprocesamiento de los datos a través de la canalización especificada.

```
aws iotanalytics cancel-pipeline-reprocessing \  
  --pipeline-name mypipeline \  
  --reprocessing-id "6ad2764f-fb13-4de3-b101-4e74af03b043"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [CancelPipelineReprocessing](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [CancelPipelineReprocessing](#) la Referencia de AWS CLI comandos.

## `create-channel`

En el siguiente ejemplo de código, se muestra cómo usar `create-channel`.

### AWS CLI

Para crear un canal

El siguiente `create-channel` ejemplo crea un canal con la configuración especificada. Un canal recopila datos de un MQTT tema y archiva los mensajes sin procesar y sin procesar antes de publicarlos en una canalización.

```
aws iotanalytics create-channel \  
  --cli-input-json file://create-channel.json
```

Contenidos de `create-channel.json`:

```
{  
  "channelName": "mychannel",  
  "retentionPeriod": {  
    "unlimited": true  
  },  
  "tags": [  
    {  
      "key": "Environment",
```

```
        "value": "Production"
      }
    ]
  }
```

Salida:

```
{
  "channelArn": "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel",
  "channelName": "mychannel",
  "retentionPeriod": {
    "unlimited": true
  }
}
```

Para obtener más información, consulte [CreateChannel](#) en la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [CreateChannel](#) en la Referencia de AWS CLI comandos.

## create-dataset-content

En el siguiente ejemplo de código, se muestra cómo usar `create-dataset-content`.

### AWS CLI

Para crear el contenido de un conjunto de datos

El siguiente `create-dataset-content` ejemplo crea el contenido del conjunto de datos especificado aplicando una `queryAction` (una SQL consulta) o una `containerAction` (ejecutando una aplicación contenerizada).

```
aws iotanalytics create-dataset-content \
  --dataset-name mydataset
```

Salida:

```
{
  "versionId": "d494b416-9850-4670-b885-ca22f1e89d62"
}
```

Para obtener más información, consulte [CreateDatasetContent](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [CreateDatasetContent](#) la Referencia de AWS CLI comandos.

## create-dataset

En el siguiente ejemplo de código, se muestra cómo usar create-dataset.

### AWS CLI

Para crear un conjunto de datos

En el siguiente create-dataset ejemplo, se crea un conjunto de datos. Un conjunto de datos almacena los datos recuperados de un banco de datos mediante la aplicación de una queryAction (una SQL consulta) o una containerAction (la ejecución de una aplicación contenerizada). Esta operación crea el esqueleto de un conjunto de datos. Puede rellenar el conjunto de datos manualmente mediante una trigger llamada CreateDatasetContent o automáticamente según lo que especifique.

```
aws iotanalytics create-dataset \  
  --cli-input-json file://create-dataset.json
```

Contenidos de create-dataset.json:

```
{  
  "datasetName": "mydataset",  
  "actions": [  
    {  
      "actionName": "myDatasetAction",  
      "queryAction": {  
        "sqlQuery": "SELECT * FROM mydatastore"  
      }  
    }  
  ],  
  "retentionPeriod": {  
    "unlimited": true  
  },  
  "tags": [  
    {
```

```
        "key": "Environment",
        "value": "Production"
      }
    ]
  }
```

Salida:

```
{
  "datasetName": "mydataset",
  "retentionPeriod": {
    "unlimited": true
  },
  "datasetArn": "arn:aws:iotanalytics:us-west-2:123456789012:dataset/mydataset"
}
```

Para obtener más información, consulte [CreateDataset](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [CreateDataset](#) la Referencia de AWS CLI comandos.

## create-datastore

En el siguiente ejemplo de código, se muestra cómo usar create-datastore.

### AWS CLI

Creación de un almacén de datos

El siguiente create-datastore ejemplo crea un almacén de datos, que es un repositorio de mensajes.

```
aws iotanalytics create-datastore \
  --cli-input-json file://create-datastore.json
```

Contenidos de create-datastore.json:

```
{
  "datastoreName": "mydatastore",
  "retentionPeriod": {
```

```
        "numberOfDays": 90
    },
    "tags": [
        {
            "key": "Environment",
            "value": "Production"
        }
    ]
}
```

Salida:

```
{
  "datastoreName": "mydatastore",
  "datastoreArn": "arn:aws:iotanalytics:us-west-2:123456789012:datastore/
mydatastore",
  "retentionPeriod": {
    "numberOfDays": 90,
    "unlimited": false
  }
}
```

Para obtener más información, consulte [CreateDatastore](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [CreateDatastore](#) la Referencia de AWS CLI comandos.

## create-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `create-pipeline`.

### AWS CLI

Cree una canalización de IoT Analytics

En el siguiente `create-pipeline` ejemplo, se crea una canalización. Una canalización consume mensajes de un canal y permite procesar los mensajes antes de guardarlos en un almacén de datos. Debe especificar tanto un canal como una actividad de almacén de datos y, de forma opcional, hasta 23 actividades adicionales en la `pipelineActivities` matriz.

```
aws iotanalytics create-pipeline \
```

```
--cli-input-json file://create-pipeline.json
```

Contenidos de `create-pipeline.json`:

```
{
  "pipelineName": "mypipeline",
  "pipelineActivities": [
    {
      "channel": {
        "name": "myChannelActivity",
        "channelName": "mychannel",
        "next": "myMathActivity"
      }
    },
    {
      "datastore": {
        "name": "myDatastoreActivity",
        "datastoreName": "mydatastore"
      }
    },
    {
      "math": {
        "name": "myMathActivity",
        "math": "((temp - 32) * 5.0) / 9.0",
        "attribute": "tempC",
        "next": "myDatastoreActivity"
      }
    }
  ],
  "tags": [
    {
      "key": "Environment",
      "value": "Beta"
    }
  ]
}
```

Salida:

```
{
  "pipelineArn": "arn:aws:iotanalytics:us-west-2:123456789012:pipeline/
mypipeline",
  "pipelineName": "mypipeline"
}
```



```
}
```

Para obtener más información, consulte [CreatePipeline](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [CreatePipeline](#) la Referencia de AWS CLI comandos.

## **delete-channel**

En el siguiente ejemplo de código, se muestra cómo usar `delete-channel`.

### AWS CLI

Eliminar un canal de IoT Analytics

En el siguiente `delete-channel` ejemplo, se elimina el canal especificado.

```
aws iotanalytics delete-channel \  
  --channel-name mychannel
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteChannel](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [DeleteChannel](#) la Referencia de AWS CLI comandos.

## **delete-dataset-content**

En el siguiente ejemplo de código, se muestra cómo usar `delete-dataset-content`.

### AWS CLI

Para eliminar el contenido del conjunto de datos

En el siguiente `delete-dataset-content` ejemplo, se elimina el contenido del conjunto de datos especificado.

```
aws iotanalytics delete-dataset-content \  
  --dataset-name mydataset
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteDatasetContent](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [DeleteDatasetContent](#) la Referencia de AWS CLI comandos.

## delete-dataset

En el siguiente ejemplo de código, se muestra cómo usar delete-dataset.

### AWS CLI

Para eliminar un conjunto de datos

En el siguiente delete-dataset ejemplo, se elimina el conjunto de datos especificado. No es necesario eliminar el contenido del conjunto de datos antes de realizar esta operación.

```
aws iotanalytics delete-dataset \  
  --dataset-name mydataset
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteDataset](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [DeleteDataset](#) la Referencia de AWS CLI comandos.

## delete-datastore

En el siguiente ejemplo de código, se muestra cómo usar delete-datastore.

### AWS CLI

Eliminación de un almacén de datos

En el siguiente delete-datastore ejemplo, se elimina el banco de datos especificado.

```
aws iotanalytics delete-datastore \  
  --datastore-name mydatastore
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteDatastore](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [DeleteDatastore](#) la Referencia de AWS CLI comandos.

## delete-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `delete-pipeline`.

### AWS CLI

Para eliminar una canalización

En el siguiente `delete-pipeline` ejemplo, se elimina la canalización especificada.

```
aws iotanalytics delete-pipeline \  
  --pipeline-name mypipeline
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeletePipeline](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [DeletePipeline](#) la Referencia de AWS CLI comandos.

## describe-channel

En el siguiente ejemplo de código, se muestra cómo usar `describe-channel`.

### AWS CLI

Para recuperar información sobre un canal

En el siguiente `describe-channel` ejemplo, se muestran los detalles, incluidas las estadísticas, del canal especificado.

```
aws iotanalytics describe-channel \  
  --channel-name mychannel \  
  --include-statistics
```

Salida:

```
{
  "statistics": {
    "size": {
      "estimatedSizeInBytes": 402.0,
      "estimatedOn": 1561504380.0
    }
  },
  "channel": {
    "status": "ACTIVE",
    "name": "mychannel",
    "lastUpdateTime": 1557860351.001,
    "creationTime": 1557860351.001,
    "retentionPeriod": {
      "unlimited": true
    },
    "arn": "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel"
  }
}
```

Para obtener más información, consulte [DescribeChannel](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [DescribeChannel](#) la Referencia de AWS CLI comandos.

## describe-dataset

En el siguiente ejemplo de código, se muestra cómo usar `describe-dataset`.

### AWS CLI

Para recuperar información sobre un conjunto de datos

En el siguiente `describe-dataset` ejemplo, se muestran los detalles del conjunto de datos especificado.

```
aws iotanalytics describe-dataset \
  --dataset-name mydataset
```

Salida:

```
{
  "dataset": {
```

```

    "status": "ACTIVE",
    "contentDeliveryRules": [],
    "name": "mydataset",
    "lastUpdateTime": 1557859240.658,
    "triggers": [],
    "creationTime": 1557859240.658,
    "actions": [
      {
        "actionName": "query_32",
        "queryAction": {
          "sqlQuery": "SELECT * FROM mydatastore",
          "filters": []
        }
      }
    ],
    "retentionPeriod": {
      "numberOfDays": 90,
      "unlimited": false
    },
    "arn": "arn:aws:iotanalytics:us-west-2:123456789012:dataset/mydataset"
  }
}

```

Para obtener más información, consulte [DescribeDataset](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [DescribeDataset](#) la Referencia de AWS CLI comandos.

## describe-datastore

En el siguiente ejemplo de código, se muestra cómo usar describe-datastore.

### AWS CLI

Para recuperar información sobre un banco de datos

En el siguiente describe-datastore ejemplo, se muestran los detalles, incluidas las estadísticas, del banco de datos especificado.

```

aws iotanalytics describe-datastore \
  --datastore-name mydatastore \
  --include-statistics

```

Salida:

```
{
  "datastore": {
    "status": "ACTIVE",
    "name": "mydatastore",
    "lastUpdateTime": 1557858971.02,
    "creationTime": 1557858971.02,
    "retentionPeriod": {
      "unlimited": true
    },
    "arn": "arn:aws:iotanalytics:us-west-2:123456789012:datastore/mydatastore"
  },
  "statistics": {
    "size": {
      "estimatedSizeInBytes": 397.0,
      "estimatedOn": 1561592040.0
    }
  }
}
```

Para obtener más información, consulte [DescribeDatastore](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [DescribeDatastore](#) la Referencia de AWS CLI comandos.

## describe-logging-options

En el siguiente ejemplo de código, se muestra cómo usar describe-logging-options.

AWS CLI

Para recuperar las opciones de registro actuales

El siguiente describe-logging-options ejemplo muestra las opciones de registro actuales de AWS IoT Analytics.

```
aws iotanalytics describe-logging-options
```

Este comando no genera ninguna salida. Salida:

```
{
  "loggingOptions": {
    "roleArn": "arn:aws:iam::123456789012:role/service-role/myIoTAnalyticsRole",
    "enabled": true,
    "level": "ERROR"
  }
}
```

Para obtener más información, consulte [DescribeLoggingOptions](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [DescribeLoggingOptions](#) la Referencia de AWS CLI comandos.

## describe-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `describe-pipeline`.

### AWS CLI

Para recuperar información sobre una canalización

En el siguiente `describe-pipeline` ejemplo, se muestran los detalles de la canalización especificada.

```
aws iotanalytics describe-pipeline \
  --pipeline-name mypipeline
```

Salida:

```
{
  "pipeline": {
    "activities": [
      {
        "channel": {
          "channelName": "mychannel",
          "name": "mychannel_28",
          "next": "mydatastore_29"
        }
      },
      {
        "datastore": {
```

```

        "datastoreName": "mydatastore",
        "name": "mydatastore_29"
    }
  ],
  "name": "mypipeline",
  "lastUpdateTime": 1561676362.515,
  "creationTime": 1557859124.432,
  "reprocessingSummaries": [
    {
      "status": "SUCCEEDED",
      "creationTime": 1561676362.189,
      "id": "6ad2764f-fb13-4de3-b101-4e74af03b043"
    }
  ],
  "arn": "arn:aws:iotanalytics:us-west-2:123456789012:pipeline/mypipeline"
}

```

Para obtener más información, consulte [DescribePipeline](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [DescribePipeline](#) la Referencia de AWS CLI comandos.

## get-dataset-content

En el siguiente ejemplo de código, se muestra cómo usar `get-dataset-content`.

### AWS CLI

Para recuperar el contenido de un conjunto de datos

El siguiente `get-dataset-content` ejemplo recupera el contenido de un conjunto de datos URIs prefirmado.

```
aws iotanalytics get-dataset-content --dataset-name mydataset
```

Salida:

```

{
  "status": {
    "state": "SUCCEEDED"
  },

```



```

    "timestamp": 1557863215.995,
    "entries": [
      {
        "dataURI": "https://aws-radiant-
dataset-12345678-1234-1234-1234-123456789012.s3.us-west-2.amazonaws.com/
results/12345678-e8b3-46ba-b2dd-efe8d86cf385.csv?X-Amz-Security-Token=...-Amz-
Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20190628T173437Z&X-Amz-SignedHeaders=host&X-
Amz-Expires=7200&X-Amz-Credential=...F20190628%2Fus-west-2%2Fs3%2Faws4_request&X-
Amz-Signature=..."
      }
    ]
  }

```

Para obtener más información, consulte [GetDatasetContent](#) la guía.

- Para API obtener más información, consulte [GetDatasetContent](#) la Referencia de AWS CLI comandos.

## list-channels

En el siguiente ejemplo de código, se muestra cómo usar `list-channels`.

### AWS CLI

Para recuperar una lista de canales

El siguiente `list-channels` ejemplo muestra información resumida de los canales disponibles.

```
aws iotanalytics list-channels
```

Salida:

```

{
  "channelSummaries": [
    {
      "status": "ACTIVE",
      "channelName": "mychannel",
      "creationTime": 1557860351.001,
      "lastUpdateTime": 1557860351.001
    }
  ]
}

```

Para obtener más información, consulte [ListChannels](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [ListChannels](#) la Referencia de AWS CLI comandos.

## list-dataset-contents

En el siguiente ejemplo de código, se muestra cómo usar `list-dataset-contents`.

### AWS CLI

Para mostrar información sobre el contenido del conjunto de datos

En el siguiente `list-dataset-contents` ejemplo, se muestra información sobre el contenido del conjunto de datos que se ha creado.

```
aws iotanalytics list-dataset-contents \  
  --dataset-name mydataset
```

Salida:

```
{  
  "datasetContentSummaries": [  
    {  
      "status": {  
        "state": "SUCCEEDED"  
      },  
      "scheduleTime": 1557863215.995,  
      "version": "b10ea2a9-66c1-4d99-8d1f-518113b738d0",  
      "creationTime": 1557863215.995  
    }  
  ]  
}
```

Para obtener más información, consulte [ListDatasetContents](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [ListDatasetContents](#) la Referencia de AWS CLI comandos.

## list-datasets

En el siguiente ejemplo de código, se muestra cómo usar `list-datasets`.

## AWS CLI

Para recuperar información sobre conjuntos de datos

En el siguiente `list-datasets` ejemplo, se muestra un resumen de la información sobre los conjuntos de datos disponibles.

```
aws iotanalytics list-datasets
```

Salida:

```
{
  "datasetSummaries": [
    {
      "status": "ACTIVE",
      "datasetName": "mydataset",
      "lastUpdateTime": 1557859240.658,
      "triggers": [],
      "creationTime": 1557859240.658,
      "actions": [
        {
          "actionName": "query_32",
          "actionType": "QUERY"
        }
      ]
    }
  ]
}
```

Para obtener más información, consulte [ListDatasets](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [ListDatasets](#) la Referencia de AWS CLI comandos.

## **list-datastores**

En el siguiente ejemplo de código, se muestra cómo usar `list-datastores`.

## AWS CLI

Para recuperar una lista de almacenes de datos

El siguiente `list-datastores` ejemplo muestra información resumida sobre los bancos de datos disponibles.

```
aws iotanalytics list-datastores
```

Salida:

```
{
  "datastoreSummaries": [
    {
      "status": "ACTIVE",
      "datastoreName": "mydatastore",
      "creationTime": 1557858971.02,
      "lastUpdateTime": 1557858971.02
    }
  ]
}
```

Para obtener más información, consulte [ListDatastores](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [ListDatastores](#) la Referencia de AWS CLI comandos.

## list-pipelines

En el siguiente ejemplo de código, se muestra cómo usar `list-pipelines`.

AWS CLI

Para recuperar una lista de canalizaciones

En el siguiente `list-pipelines` ejemplo, se muestra una lista de las canalizaciones disponibles.

```
aws iotanalytics list-pipelines
```

Salida:

```
{
  "pipelineSummaries": [
    {
      "pipelineName": "mypipeline",
      "creationTime": 1557859124.432,
      "lastUpdateTime": 1557859124.432,
      "reprocessingSummaries": []
    }
  ]
}
```

```
    }  
  ]  
}
```

Para obtener más información, consulte [ListPipelines](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [ListPipelines](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas que ha adjuntado al recurso especificado.

```
aws iotanalytics list-tags-for-resource \  
  --resource-arn "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel"
```

Salida:

```
{  
  "tags": [  
    {  
      "value": "bar",  
      "key": "foo"  
    }  
  ]  
}
```

Para obtener más información, consulte [ListTagsForResource](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## put-logging-options

En el siguiente ejemplo de código, se muestra cómo usar `put-logging-options`.

## AWS CLI

Para configurar o actualizar las opciones de registro

El siguiente `put-logging-options` ejemplo establece o actualiza las opciones de registro de AWS IoT Analytics. Si actualiza el valor de cualquier `loggingOptions` campo, el cambio puede tardar hasta un minuto en surtir efecto. Además, si cambias la política asociada a la función que especificaste en el campo `roleArn` «» (por ejemplo, para corregir una política no válida), el cambio puede tardar hasta cinco minutos en surtir efecto.

```
aws iotanalytics put-logging-options \  
  --cli-input-json file://put-logging-options.json
```

Contenidos de `put-logging-options.json`:

```
{  
  "loggingOptions": {  
    "roleArn": "arn:aws:iam::123456789012:role/service-role/myIoTAnalyticsRole",  
    "level": "ERROR",  
    "enabled": true  
  }  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [PutLoggingOptions](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [PutLoggingOptions](#) la Referencia de AWS CLI comandos.

## run-pipeline-activity

En el siguiente ejemplo de código, se muestra cómo usar `run-pipeline-activity`.

## AWS CLI

Para simular una actividad de canalización

El siguiente `run-pipeline-activity` ejemplo simula los resultados de ejecutar una actividad de canalización en una carga útil de mensajes.

```
aws iotanalytics run-pipeline-activity \  
  --pipeline-activity file://maths.json \  
  --payloads file://payloads.json
```

Contenidos de `maths.json`:

```
{  
  "math": {  
    "name": "MyMathActivity",  
    "math": "((temp - 32) * 5.0) / 9.0",  
    "attribute": "tempC"  
  }  
}
```

Contenidos de `payloads.json`:

```
[  
  {"humidity": 52, "temp": 68 },  
  {"humidity": 52, "temp": 32 }  
]
```

Salida:

```
{  
  "logResult": "",  
  "payloads": [  
    "eyJodW1pZG10eSI6NTIsInRlbXAiOjY4LCJ0ZW1wQyI6MjB9",  
    "eyJodW1pZG10eSI6NTIsInRlbXAiOjMyLCJ0ZW1wQyI6MH0="  ]  
}
```

Para obtener más información, consulte [RunPipelineActivity](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [RunPipelineActivity](#) la Referencia de AWS CLI comandos.

## sample-channel-data

En el siguiente ejemplo de código, se muestra cómo usar `sample-channel-data`.

## AWS CLI

Para recuperar mensajes de muestra de un canal

El siguiente `sample-channel-data` ejemplo recupera una muestra de mensajes del canal especificado ingeridos durante el período de tiempo especificado. Puede recuperar hasta 10 mensajes.

```
aws iotanalytics sample-channel-data \  
  --channel-name mychannel
```

Salida:

```
{  
  "payloads": [  
    "eyJhdGVtcGVyYXR1cmUiOiAyMCA9",  
    "eyJhZm9vIjogImJhcnVzIj0="
```

Para obtener más información, consulte [SampleChannelData](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [SampleChannelData](#) la Referencia de AWS CLI comandos.

## start-pipeline-reprocessing

En el siguiente ejemplo de código, se muestra cómo usar `start-pipeline-reprocessing`.

### AWS CLI

Para iniciar el reprocesamiento de la canalización

En el siguiente `start-pipeline-reprocessing` ejemplo, se inicia el reprocesamiento de los datos de los mensajes sin procesar a través de la canalización especificada.

```
aws iotanalytics start-pipeline-reprocessing \  
  --pipeline-name mypipeline
```



Salida:

```
{
  "reprocessingId": "6ad2764f-fb13-4de3-b101-4e74af03b043"
}
```

Para obtener más información, consulte [StartPipelineReprocessing](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [StartPipelineReprocessing](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

AWS CLI

Para añadir o modificar las etiquetas de un recurso

En el siguiente `tag-resource` ejemplo, se agregan o modifican las etiquetas adjuntas al recurso especificado.

```
aws iotanalytics tag-resource \
  --resource-arn "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel" \
  --tags "[{"key": "Environment", "value": "Production"}]"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [TagResource](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

AWS CLI

Cómo eliminar etiquetas de un recurso

En el siguiente `untag-resource` ejemplo, se eliminan las etiquetas con los nombres de clave especificados del recurso especificado.

```
aws iotanalytics untag-resource \  
  --resource-arn "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel" \  
  --tag-keys "[\"Environment\"]"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte `UntagResource` < [https://docs.aws.amazon.com/iotanalytics/APIReference/latest/API\\_UntagResource.html](https://docs.aws.amazon.com/iotanalytics/APIReference/latest/API_UntagResource.html) > en la Referencia de AWS IoT Analytics. API

- Para API obtener más información, consulte la Referencia de [UntagResource](#) comandos AWS CLI .

## update-channel

En el siguiente ejemplo de código, se muestra cómo usar `update-channel`.

### AWS CLI

Para modificar un canal

En el siguiente `update-channel` ejemplo, se modifican los ajustes del canal especificado.

```
aws iotanalytics update-channel \  
  --cli-input-json file://update-channel.json
```

Contenidos de `update-channel.json`:

```
{  
  "channelName": "mychannel",  
  "retentionPeriod": {  
    "numberOfDays": 92  
  }  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [UpdateChannel](#) la APIReferencia de AWS IoT Analytics.

- Para API obtener más información, consulte [UpdateChannel](#) en la Referencia de AWS CLI comandos.

## update-dataset

En el siguiente ejemplo de código, se muestra cómo usar `update-dataset`.

### AWS CLI

Para actualizar un conjunto de datos

El siguiente `update-dataset` ejemplo modifica la configuración del conjunto de datos especificado.

```
aws iotanalytics update-dataset \  
  --cli-input-json file://update-dataset.json
```

Contenidos de `update-dataset.json`:

```
{  
  "datasetName": "mydataset",  
  "actions": [  
    {  
      "actionName": "myDatasetUpdateAction",  
      "queryAction": {  
        "sqlQuery": "SELECT * FROM mydatastore"  
      }  
    }  
  ],  
  "retentionPeriod": {  
    "numberOfDays": 92  
  }  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte `UpdateDataset < https://docs.aws.amazon.com/iotanalytics/APIReference/latest/API_UpdateDataset.html >` en la Referencia de AWS IoT Analytics. API

- Para API obtener más información, consulte la Referencia de [UpdateDataset](#) comandos AWS CLI .

## update-datastore

En el siguiente ejemplo de código, se muestra cómo usar `update-datastore`.

### AWS CLI

Para actualizar un banco de datos

En el siguiente `update-datastore` ejemplo, se modifica la configuración del banco de datos especificado.

```
aws iotanalytics update-datastore \  
  --cli-input-json file://update-datastore.json
```

Contenido de `update-datastore.json`:

```
{  
  "datastoreName": "mydatastore",  
  "retentionPeriod": {  
    "numberOfDays": 93  
  }  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [UpdateDatastore](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [UpdateDatastore](#) la Referencia de AWS CLI comandos.

## update-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `update-pipeline`.

### AWS CLI

Para actualizar una canalización

El siguiente `update-pipeline` ejemplo modifica la configuración de la canalización especificada. Debe especificar un canal y una actividad de almacén de datos y, opcionalmente, hasta 23 actividades adicionales en la `pipelineActivities` matriz.

```
aws iotanalytics update-pipeline \  
  --cli-input-json file://update-pipeline.json
```

```
--cli-input-json file://update-pipeline.json
```

Contenido de update-pipeline.json:

```
{
  "pipelineName": "mypipeline",
  "pipelineActivities": [
    {
      "channel": {
        "name": "myChannelActivity",
        "channelName": "mychannel",
        "next": "myMathActivity"
      }
    },
    {
      "datastore": {
        "name": "myDatastoreActivity",
        "datastoreName": "mydatastore"
      }
    },
    {
      "math": {
        "name": "myMathActivity",
        "math": "(((temp - 32) * 5.0) / 9.0) + 273.15",
        "attribute": "tempK",
        "next": "myDatastoreActivity"
      }
    }
  ]
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [UpdatePipeline](#) la API Referencia de AWS IoT Analytics.

- Para API obtener más información, consulte [UpdatePipeline](#) la Referencia de AWS CLI comandos.

## Ejemplos de Device Advisor que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante el AWS Command Line Interface uso de Device Advisor.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **create-suite-definition**

En el siguiente ejemplo de código, se muestra cómo usar `create-suite-definition`.

## AWS CLI

Ejemplo 1: Para crear un conjunto de pruebas de IoT Device Advisor

El siguiente `create-suite-definition` ejemplo crea un conjunto de pruebas de Device Advisor en el AWS IoT con la configuración de definición de conjunto especificada.

```
aws iotdeviceadvisor create-suite-definition \  
  --suite-definition-configuration '{ \  
    "suiteDefinitionName": "TestSuiteName", \  
    "devices": [{"thingArn": "arn:aws:iot:us-east-1:123456789012:thing/  
MyIoTThing"}], \  
    "intendedForQualification": false, \  
    "rootGroup": "{ \"configuration\": {}, \"tests\": [{ \"name\": \"MQTT Connect\",  
\"configuration\": { \"EXECUTION_TIMEOUT\": 120 }, \"tests\": [{ \"name\": \"MQTT_Connect\",  
\"configuration\": {}, \"test\": { \"id\": \"MQTT_Connect\", \"testCase\": null, \"version  
\": \"0.0.0\" } } ] } ] }", \  
    "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole" }
```

## Salida:

```
{  
  "suiteDefinitionId": "0jtsigio7yenu",  
  "suiteDefinitionArn": "arn:aws:iotdeviceadvisor:us-  
east-1:123456789012:suitedefinition/0jtsigio7yenu",
```

```
"suiteDefinitionName": "TestSuiteName",
"createdAt": "2022-12-02T11:38:13.263000-05:00"
}
```

Para obtener más información, consulte [Crear una definición de conjunto de pruebas](#) en la Guía para desarrolladores de AWS IoT Core.

Ejemplo 2: Para crear un conjunto de pruebas de calificación más reciente de IoT Device Advisor

El siguiente `create-suite-definition` ejemplo crea un conjunto de pruebas de calificación de Device Advisor con la última versión del AWS IoT con la configuración de definición de suite especificada.

```
aws iotdeviceadvisor create-suite-definition \
  --suite-definition-configuration '{ \
    "suiteDefinitionName": "TestSuiteName", \
    "devices": [{"thingArn": "arn:aws:iot:us-east-1:123456789012:thing/MyIoTThing"}], \
    "intendedForQualification": true, \
    "rootGroup": "", \
    "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole"}
```

Salida:

```
{
  "suiteDefinitionId": "txgsuolk2myj",
  "suiteDefinitionArn": "arn:aws:iotdeviceadvisor:us-east-1:123456789012:suitedefinition/txgsuolk2myj",
  "suiteDefinitionName": "TestSuiteName",
  "createdAt": "2022-12-02T11:38:13.263000-05:00"
}
```

Para obtener más información, consulte [Crear una definición de conjunto de pruebas](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [CreateSuiteDefinition](#) la Referencia de AWS CLI comandos.

## delete-suite-definition

En el siguiente ejemplo de código, se muestra cómo usar `delete-suite-definition`.

## AWS CLI

Para eliminar el conjunto de pruebas de IoT Device Advisor

En el siguiente `delete-suite-definition` ejemplo, se elimina el conjunto de pruebas de Device Advisor con el ID de definición de conjunto especificado.

```
aws iotdeviceadvisor delete-suite-definition \  
  --suite-definition-id 0jtsgio7yenu
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteSuiteDefinition](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [DeleteSuiteDefinition](#) la Referencia de AWS CLI comandos.

## get-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `get-endpoint`.

### AWS CLI

Ejemplo 1: Para obtener la información sobre un punto final a nivel de cuenta de IoT Device Advisor

En el siguiente `get-endpoint` ejemplo, se obtiene la información sobre un terminal de prueba a nivel de cuenta de Device Advisor.

```
aws iotdeviceadvisor get-endpoint
```

Salida:

```
{  
  "endpoint": "t6y4c143x9sfo.deviceadvisor.iot.us-east-1.amazonaws.com"  
}
```

Ejemplo 2: Para obtener la información sobre un punto final a nivel de dispositivo de IoT Device Advisor

En el siguiente `get-endpoint` ejemplo, se obtiene la información sobre un punto final de prueba a nivel de dispositivo de Device Advisor con el `thing-arn` o el `certificate-arn` especificados.



```
aws iotdeviceadvisor get-endpoint \  
  --thing-arn arn:aws:iot:us-east-1:123456789012:thing/MyIotThing
```

Salida:

```
{  
  "endpoint": "tdb7719be5t6y4c143x9sfo.deviceadvisor.iot.us-east-1.amazonaws.com"  
}
```

Para obtener más información, consulte [Obtenga un punto final de prueba](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [GetEndpoint](#) la Referencia de AWS CLI comandos.

## get-suite-definition

En el siguiente ejemplo de código, se muestra cómo usar `get-suite-definition`.

### AWS CLI

Para obtener información sobre un conjunto de pruebas de IoT Device Advisor

En el siguiente `get-suite-definition` ejemplo, se obtiene la información sobre un conjunto de pruebas de aevice Advisor con el ID de definición de conjunto especificado.

```
aws iotdeviceadvisor get-suite-definition \  
  --suite-definition-id qqcsmtyyjabl
```

Salida:

```
{  
  "suiteDefinitionId": "qqcsmtyyjabl",  
  "suiteDefinitionArn": "arn:aws:iotdeviceadvisor:us-  
east-1:123456789012:suitedefinition/qqcsmtyyjabl",  
  "suiteDefinitionVersion": "v1",  
  "latestVersion": "v1",  
  "suiteDefinitionConfiguration": {  
    "suiteDefinitionName": "MQTT connection",  
    "devices": [],  
    "intendedForQualification": false,  
    "isLongDurationTest": false,  
  }  
}
```

```

    "rootGroup": "{ \"configuration\": {}, \"tests\": [ { \"id\": \"uta5d9j1kvwc\",
  \"name\": \"Test group 1\", \"configuration\": {}, \"tests\": [ { \"id\": \"awr8pq5vc9yp\",
  \"name\": \"MQTT Connect\", \"configuration\": {}, \"test\": { \"id\": \"MQTT_Connect\",
  \"testCase\": null, \"version\": \"0.0.0\" } } ] } ] }",
    "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole",
    "protocol": "MqttV3_1_1"
  },
  "createdAt": "2022-11-11T22:28:52.389000-05:00",
  "lastModifiedAt": "2022-11-11T22:28:52.389000-05:00",
  "tags": {}
}

```

Para obtener más información, consulte [Obtener una definición de conjunto de pruebas](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [GetSuiteDefinition](#) la Referencia de AWS CLI comandos.

## get-suite-run-report

En el siguiente ejemplo de código, se muestra cómo usar `get-suite-run-report`.

### AWS CLI

Para obtener información sobre un informe de ejecución de un conjunto de pruebas apto para IoT Device Advisor

En el siguiente `get-suite-run-report` ejemplo, se obtiene el enlace de descarga del informe correspondiente a un conjunto de pruebas que cumpla los requisitos de Device Advisor y que se ejecute correctamente con el identificador de definición de conjunto y el identificador de ejecución de conjunto especificados.

```

aws iotdeviceadvisor get-suite-run-report \
  --suite-definition-id ztvb5aek4w4x \
  --suite-run-id p6awv83nre6v

```

Salida:

```

{
  "qualificationReportDownloadUrl": "https://senate-apn-reports-us-east-1-
  prod.s3.amazonaws.com/report.downloadlink"
}

```

```
}
```

Para obtener más información, consulte [Obtenga un informe de calificación para ejecutar con éxito un conjunto de pruebas de calificación](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [GetSuiteRunReport](#) la Referencia de AWS CLI comandos.

## get-suite-run

En el siguiente ejemplo de código, se muestra cómo usar `get-suite-run`.

### AWS CLI

Para obtener información sobre el estado de ejecución de un conjunto de pruebas de IoT Device Advisor

En el siguiente `get-suite-run` ejemplo, se obtiene la información sobre el estado de ejecución de un conjunto de pruebas de Device Advisor con el ID de definición de conjunto y el ID de ejecución de conjunto especificados.

```
aws iotdeviceadvisor get-suite-run \  
  --suite-definition-id qqcsmtyyjabl \  
  --suite-run-id nzlfyhaa18oa
```

Salida:

```
{  
  "suiteDefinitionId": "qqcsmtyyjabl",  
  "suiteDefinitionVersion": "v1",  
  "suiteRunId": "nzlfyhaa18oa",  
  "suiteRunArn": "arn:aws:iotdeviceadvisor:us-east-1:123456789012:suiterun/  
qqcsmtyyjabl/nzlfyhaa18oa",  
  "suiteRunConfiguration": {  
    "primaryDevice": {  
      "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/MyIotThing",  
      "certificateArn": "arn:aws:iot:us-east-1:123456789012:cert/certFile"  
    },  
    "parallelRun": false  
  },  
  "testResult": {
```

```

    "groups": [
      {
        "groupId": "uta5d9j1kvwc",
        "groupName": "Test group 1",
        "tests": [
          {
            "testCaseRunId": "2ve2twrqyr0s",
            "testCaseDefinitionId": "awr8pq5vc9yp",
            "testCaseDefinitionName": "MQTT Connect",
            "status": "PASS",
            "startTime": "2022-11-12T00:01:53.693000-05:00",
            "endTime": "2022-11-12T00:02:15.443000-05:00",
            "logUrl": "https://console.aws.amazon.com/cloudwatch/home?region=us-east-1#logEventViewer:group=/aws/iot/deviceadvisor/qqsmtyyjabl;stream=nzlfyhaa18oa_2ve2twrqyr0s",
            "warnings": "null",
            "failure": "null"
          }
        ]
      }
    ],
    "startTime": "2022-11-12T00:01:52.673000-05:00",
    "endTime": "2022-11-12T00:02:16.496000-05:00",
    "status": "PASS",
    "tags": {}
  }
}

```

Para [obtener más información, consulte Ejecutar un conjunto de pruebas](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [GetSuiteRun](#) la Referencia de AWS CLI comandos.

## list-suite-definitions

En el siguiente ejemplo de código, se muestra cómo usar `list-suite-definitions`.

### AWS CLI

Ejemplo 1: Para enumerar los conjuntos de pruebas de IoT Device Advisor que ha creado

En el siguiente `list-suite-definitions` ejemplo, se enumeran hasta 25 conjuntos de pruebas de Device Advisor que creó en el AWS IoT. Si tiene más de 25 conjuntos de pruebas, se

mostrará el símbolo `nextToken` "» en el resultado. Puede usar este `nextToken` para mostrar el resto de los conjuntos de pruebas que ha creado.

```
aws iotdeviceadvisor list-suite-definitions
```

Salida:

```
{
  "suiteDefinitionInformationList": [
    {
      "suiteDefinitionId": "3hsn88h4p2g5",
      "suiteDefinitionName": "TestSuite1",
      "defaultDevices": [
        {
          "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/
MyIotThing"
        }
      ],
      "intendedForQualification": false,
      "isLongDurationTest": false,
      "protocol": "MqttV3_1_1",
      "createdAt": "2022-11-17T14:15:56.830000-05:00"
    },
    {
      .....
    }
  ],
  "nextToken": "nextTokenValue"
}
```

Ejemplo 2: Para enumerar los conjuntos de pruebas de IoT Device Advisor que creó con la configuración especificada

En el siguiente `list-suite-definitions` ejemplo, se enumeran los conjuntos de pruebas de Device Advisor que creó en el AWS IoT con el número máximo de resultados especificado. Si tiene más conjuntos de pruebas que el número máximo, en el resultado se mostrará el símbolo `nextToken` "». Si tiene `nextToken`«, puede usar `nextToken` para mostrar los conjuntos de pruebas que creó y que no se mostraban antes.

```
aws iotdeviceadvisor list-suite-definitions \
  --max-result 1 \
```

```
--next-token "nextTokenValue"
```

Salida:

```
{
  "suiteDefinitionInformationList": [
    {
      "suiteDefinitionId": "ztlv5aew4w4x",
      "suiteDefinitionName": "TestSuite2",
      "defaultDevices": [],
      "intendedForQualification": true,
      "isLongDurationTest": false,
      "protocol": "MqttV3_1_1",
      "createdAt": "2022-11-17T14:15:56.830000-05:00"
    }
  ],
  "nextToken": "nextTokenValue"
}
```

Para obtener más información, consulte [ListSuiteDefinitions](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [ListSuiteDefinitions](#) la Referencia de AWS CLI comandos.

## list-suite-runs

En el siguiente ejemplo de código, se muestra cómo usar `list-suite-runs`.

### AWS CLI

Ejemplo 1: Para enumerar toda la información sobre el estado de ejecución del conjunto de pruebas de IoT Device Advisor especificado

En el siguiente `list-suite-runs` ejemplo, se muestra toda la información sobre el estado de ejecución de un conjunto de pruebas de Device Advisor con el ID de definición de conjunto especificado. Si ha ejecutado más de 25 conjuntos de pruebas, en el resultado se mostrará el símbolo `nextToken` "». Puede usar este "nextToken" para mostrar el resto de las ejecuciones del conjunto de pruebas.

```
aws iotdeviceadvisor list-suite-runs \
  --suite-definition-id ztlv5aew4w4x
```

Salida:

```
{
  "suiteRunsList": [
    {
      "suiteDefinitionId": "ztlvb5aew4w4x",
      "suiteDefinitionVersion": "v1",
      "suiteDefinitionName": "TestSuite",
      "suiteRunId": "p6awv89nre6v",
      "createdAt": "2022-12-01T16:33:14.212000-05:00",
      "startedAt": "2022-12-01T16:33:15.710000-05:00",
      "endAt": "2022-12-01T16:42:03.323000-05:00",
      "status": "PASS",
      "passed": 6,
      "failed": 0
    }
  ]
}
```

Ejemplo 2: Para mostrar información sobre el estado de ejecución del conjunto de pruebas de IoT Device Advisor especificado con la configuración especificada

En el siguiente `list-suite-runs` ejemplo, se muestra información sobre el estado de ejecución de un conjunto de pruebas de Device Advisor con el ID de definición del conjunto especificado y el número máximo de resultado especificado. Si tiene más ejecuciones de conjuntos de pruebas que el número máximo, en el resultado se mostrará el símbolo `nextToken` «». Si tiene "nextToken«, puede usar "nextToken "para mostrar las ejecuciones del conjunto de pruebas que no se mostraron antes.

```
aws iotdeviceadvisor list-suite-runs \
  --suite-definition-id qqcsmtyyjaml \
  --max-result 1 \
  --next-token "nextTokenValue"
```

Salida:

```
{
  "suiteRunsList": [
    {
      "suiteDefinitionId": "qqcsmtyyjaml",
      "suiteDefinitionVersion": "v1",
```

```

        "suiteDefinitionName": "MQTT connection",
        "suiteRunId": "gz9vm2s6d2jy",
        "createdAt": "2022-12-01T20:10:27.079000-05:00",
        "startedAt": "2022-12-01T20:10:28.003000-05:00",
        "endAt": "2022-12-01T20:10:45.084000-05:00",
        "status": "STOPPED",
        "passed": 0,
        "failed": 0
    }
],
"nextToken": "nextTokenValue"
}

```

Para obtener más información, consulte [ListSuiteRuns](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [ListSuiteRuns](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas adjuntas a un recurso de IoT Device Advisor

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas adjuntas a un recurso de Device Advisor. El recurso del asesor de dispositivos puede ser un `SuiteDefinition-Arn` o un `SuiteRun-Arn`.

```

aws iotdeviceadvisor list-tags-for-resource \
  --resource-arn arn:aws:iotdeviceadvisor:us-east-1:123456789012:suitedefinition/
ba0uyjpg38ny

```

Salida:

```

{
  "tags": {
    "TestTagKey": "TestTagValue"
  }
}

```



Para obtener más información, consulte [ListTagsForResource](#) los [tipos de API](#) [referencia y recursos de AWS IoT definidos por AWS IoT Core Device Advisor](#) en la Referencia de autorización de servicios.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## start-suite-run

En el siguiente ejemplo de código, se muestra cómo usar `start-suite-run`.

### AWS CLI

Para iniciar un conjunto de pruebas de IoT Device Advisor, ejecute

El siguiente `start-suite-run` ejemplo muestra los widgets disponibles en su AWS cuenta.

```
aws iotdeviceadvisor start-suite-run \  
  --suite-definition-id qqcsmtyyjabl \  
  --suite-definition-version v1 \  
  --suite-run-configuration '{"primaryDevice":{"thingArn": "arn:aws:iot:us-  
east-1:123456789012:thing/MyIotThing", "certificateArn": "arn:aws:iot:us-  
east-1:123456789012:cert/certFile"}}'
```

Salida:

```
{  
  "suiteRunId": "pwmucgw71t9s",  
  "suiteRunArn": "arn:aws:iotdeviceadvisor:us-east-1:123456789012:suiterun/  
qqcsmtyyjabl/pwmucgw71k9s",  
  "createdAt": "2022-12-02T15:43:05.581000-05:00"  
}
```

Para obtener más información, consulte [Iniciar la ejecución de un conjunto de pruebas](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [StartSuiteRun](#) la Referencia de AWS CLI comandos.

## stop-suite-run

En el siguiente ejemplo de código, se muestra cómo usar `stop-suite-run`.

## AWS CLI

Para detener un conjunto de pruebas de IoT Device Advisor que se está ejecutando actualmente

El siguiente `stop-suite-run` ejemplo detiene un conjunto de pruebas de Device Advisor que se está ejecutando actualmente con el ID de definición de conjunto y el ID de ejecución de conjunto especificados.

```
aws iotdeviceadvisor stop-suite-run \  
  --suite-definition-id qqcsmtyyjabl \  
  --suite-run-id nzlfyhaa18oa
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Detener la ejecución de un conjunto de pruebas](#) en la Guía para desarrolladores de AWS IoT Core.

- Para API obtener más información, consulte [StopSuiteRun](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

## AWS CLI

Para agregar y modificar las etiquetas existentes de un recurso de IoT Device Advisor

El siguiente `tag-resource` ejemplo agrega y modifica las etiquetas existentes de un recurso de Device Advisor con el `arn` y las etiquetas del recurso especificados. El recurso del asesor de dispositivos puede ser un `SuiteDefinition-Arn` o un `SuiteRun-Arn`.

```
aws iotdeviceadvisor tag-resource \  
  --resource-arn arn:aws:iotdeviceadvisor:us-east-1:123456789012:suitedefinition/  
ba0uyjpg38ny \  
  --tags '{"TagKey": "TagValue"}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [TagResource](#) los [tipos de API referencia y recursos de AWS IoT definidos por AWS IoT Core Device Advisor](#) en la Referencia de autorización de servicios.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar las etiquetas existentes de un recurso de IoT Device Advisor

El siguiente `untag-resource` ejemplo elimina las etiquetas existentes de un recurso de Device Advisor con el `arn` y la clave de etiqueta del recurso especificados. El recurso del asesor de dispositivos puede ser un `SuiteDefinition-Arn` o un `SuiteRun-Arn`.

```
aws iotdeviceadvisor untag-resource \  
  --resource-arn arn:aws:iotdeviceadvisor:us-east-1:123456789012:suitedefinition/  
ba0uyjpg38ny \  
  --tag-keys "TagKey"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [UntagResource](#) los [tipos de API referencia y recursos de AWS IoT definidos por AWS IoT Core Device Advisor](#) en la Referencia de autorización de servicios.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-suite-definition

En el siguiente ejemplo de código, se muestra cómo usar `update-suite-definition`.

### AWS CLI

Ejemplo 1: Para actualizar un conjunto de pruebas de IoT Device Advisor

El siguiente `update-suite-definition` ejemplo actualiza un conjunto de pruebas de Device Advisor en el AWS IoT con el ID de definición de suite y la configuración de definición de suite especificados.

```
aws iotdeviceadvisor update-suite-definition \
  --suite-definition-id 3hsn88h4p2g5 \
  --suite-definition-configuration '{ \
    "suiteDefinitionName": "TestSuiteName", \
    "devices": [{"thingArn": "arn:aws:iot:us-east-1:123456789012:thing/MyIotThing"}], \
    "intendedForQualification": false, \
    "rootGroup": "{ \"configuration\": {}, \"tests\": [{ \"name\": \"MQTT Connect\", \
  \"configuration\": { \"EXECUTION_TIMEOUT\": 120 }, \"tests\": [{ \"name\": \"MQTT_Connect\", \
  \"configuration\": {}, \"test\": { \"id\": \"MQTT_Connect\", \"testCase\": null, \"version \
  \": \"0.0.0\" } } ] } ] }", \
    "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole" }
```

Salida:

```
{
  "suiteDefinitionId": "3hsn88h4p2g5",
  "suiteDefinitionName": "TestSuiteName",
  "suiteDefinitionVersion": "v3",
  "createdAt": "2022-11-17T14:15:56.830000-05:00",
  "lastUpdatedAt": "2022-12-02T16:02:45.857000-05:00"
}
```

Ejemplo 2: Para actualizar un conjunto de pruebas de calificación de IoT Device Advisor

El siguiente update-suite-definition ejemplo actualiza un conjunto de pruebas de calificación de Device Advisor en el AWS IoT con el ID de definición de suite y la configuración de definición de suite especificados.

```
aws iotdeviceadvisor update-suite-definition \
  --suite-definition-id txgsuolk2myj \
  --suite-definition-configuration '{ \
    "suiteDefinitionName": "TestSuiteName", \
    "devices": [{"thingArn": "arn:aws:iot:us-east-1:123456789012:thing/MyIotThing"}], \
    "intendedForQualification": true, \
    "rootGroup": "", \
    "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole" }
```

Salida:

```
{
```

```
"suiteDefinitionId": "txgsuolk2myj",
"suiteDefinitionName": "TestSuiteName",
"suiteDefinitionVersion": "v3",
"createdAt": "2022-11-17T14:15:56.830000-05:00",
"lastUpdatedAt": "2022-12-02T16:02:45.857000-05:00"
}
```

Para obtener más información, consulte [UpdateSuiteDefinition](#) la API Referencia de AWS IoT.

- Para API obtener más información, consulte [UpdateSuiteDefinition](#) la Referencia de AWS CLI comandos.

## AWS IoT data ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS IoT data.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **delete-thing-shadow**

En el siguiente ejemplo de código, se muestra cómo usar `delete-thing-shadow`.

### AWS CLI

Para eliminar el documento oculto de un dispositivo

En el siguiente `delete-thing-shadow` ejemplo, se elimina todo el documento oculto del dispositivo denominado `MyRPI`.

```
aws iot-data delete-thing-shadow \  
  --thing-name MyRPi \  
  "output.txt"
```

El comando no produce ningún resultado en la pantalla, pero `output.txt` contiene información que confirma la versión y la marca de tiempo del documento paralelo que ha eliminado.

```
{"version":2,"timestamp":1560270384}
```

Para obtener más información, consulte [Uso de sombras](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DeleteThingShadow](#) la Referencia de AWS CLI comandos.

## get-thing-shadow

En el siguiente ejemplo de código, se muestra cómo usar `get-thing-shadow`.

### AWS CLI

Para obtener un documento paralelo

El siguiente `get-thing-shadow` ejemplo obtiene el documento de sombra de la cosa de IoT especificada.

```
aws iot-data get-thing-shadow \  
  --thing-name MyRPi \  
  output.txt
```

El comando no produce ningún resultado en la pantalla, pero a continuación se muestra el contenido de `output.txt`:

```
{  
  "state":{  
    "reported":{  
      "moisture":"low"  
    }  
  },  
  "metadata":{
```

```

    "reported":{
      "moisture":{
        "timestamp":1560269319
      }
    },
    "version":1,"timestamp":1560269405
  }

```

Para obtener más información, consulte [Device Shadow Service Data Flow](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetThingShadow](#) la Referencia de AWS CLI comandos.

## update-thing-shadow

En el siguiente ejemplo de código, se muestra cómo usar `update-thing-shadow`.

### AWS CLI

Para actualizar la sombra de una cosa

El siguiente `update-thing-shadow` ejemplo modifica el estado actual de la sombra del dispositivo para el elemento especificado y lo guarda en el archivo `output.txt`.

```

aws iot-data update-thing-shadow \
  --thing-name MyRPI \
  --payload '{"state":{"reported":{"moisture":"okay"}}}' \
  output.txt

```

El comando no produce ningún resultado en la pantalla, pero a continuación se muestra el contenido de `output.txt`:

```

{
  "state": {
    "reported": {
      "moisture": "okay"
    }
  },
  "metadata": {

```

```
    "reported": {
      "moisture": {
        "timestamp": 1560270036
      }
    },
    "version": 2,
    "timestamp": 1560270036
  }
```

Para obtener más información, consulte [Device Shadow Service Data Flow](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateThingShadow](#) la Referencia de AWS CLI comandos.

## AWS IoT Events ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS IoT Events.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **batch-put-message**

En el siguiente ejemplo de código, se muestra cómo usar batch-put-message.

AWS CLI

Para enviar mensajes (entradas) a AWS IoT Events



El siguiente `batch-put-message` ejemplo envía un conjunto de mensajes al sistema AWS IoT Events. La carga útil de cada mensaje se transforma en la entrada que especifique (`inputName`) y se ingiere en cualquier detector que supervise esa entrada. Si se envían varios mensajes, no se garantiza el orden en que se procesan los mensajes. Para garantizar el pedido, debes enviar los mensajes de uno en uno y esperar a que te respondan correctamente.

```
aws iotevents-data batch-put-message \  
  --cli-input-json file://highPressureMessage.json
```

Contenidos de `highPressureMessage.json`:

```
{  
  "messages": [  
    {  
      "messageId": "00001",  
      "inputName": "PressureInput",  
      "payload": "{\"motorid\": \"Fulton-A32\", \"sensorData\": {\"pressure\":  
80, \"temperature\": 39} }"  
    }  
  ]  
}
```

Salida:

```
{  
  "BatchPutMessageErrorEntries": []  
}
```

Para obtener más información, consulte [BatchPutMessage](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [BatchPutMessage](#) la Referencia de AWS CLI comandos.

## **batch-update-detector**

En el siguiente ejemplo de código, se muestra cómo usar `batch-update-detector`.

AWS CLI

Para actualizar un detector (instancia)

El siguiente `batch-update-detector` ejemplo actualiza el estado, los valores de las variables y la configuración del temporizador de uno o más detectores (instancias) de un modelo de detector específico.

```
aws iotevents-data batch-update-detector \  
  --cli-input-json file://budFulton-A32.json
```

Contenidos de `budFulton-A32.json`:

```
{  
  "detectors": [  
    {  
      "messageId": "00001",  
      "detectorModelName": "motorDetectorModel",  
      "keyValue": "Fulton-A32",  
      "state": {  
        "stateName": "Normal",  
        "variables": [  
          {  
            "name": "pressureThresholdBreached",  
            "value": "0"  
          }  
        ],  
        "timers": [  
        ]  
      }  
    }  
  ]  
}
```

Salida:

```
{  
  "batchUpdateDetectorErrorEntries": []  
}
```

Para obtener más información, consulte [BatchUpdateDetector](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [BatchUpdateDetector](#) la Referencia de AWS CLI comandos.

## create-detector-model

En el siguiente ejemplo de código, se muestra cómo usar `create-detector-model`.

### AWS CLI

Para crear un modelo de detector

El siguiente `create-detector-model` ejemplo crea un modelo de detector con su configuración especificada en un archivo de parámetros.

```
aws iotevents create-detector-model \  
  --cli-input-json file://motorDetectorModel.json
```

Contenidos de `motorDetectorModel.json`:

```
{  
  "detectorModelName": "motorDetectorModel",  
  "detectorModelDefinition": {  
    "states": [  
      {  
        "stateName": "Normal",  
        "onEnter": {  
          "events": [  
            {  
              "eventName": "init",  
              "condition": "true",  
              "actions": [  
                {  
                  "setVariable": {  
                    "variableName": "pressureThresholdBreached",  
                    "value": "0"  
                  }  
                }  
              ]  
            }  
          ]  
        }  
      ],  
    },  
    "onInput": {  
      "transitionEvents": [  
        {  
          "eventName": "Overpressurized",
```

```

        "condition": "$input.PressureInput.sensorData.pressure
> 70",
        "actions": [
            {
                "setVariable": {
                    "variableName": "pressureThresholdBreached",
                    "value":
"$variable.pressureThresholdBreached + 3"
                }
            }
        ],
        "nextState": "Dangerous"
    }
]
},
{
    "stateName": "Dangerous",
    "onEnter": {
        "events": [
            {
                "eventName": "Pressure Threshold Breached",
                "condition": "$variable.pressureThresholdBreached >
1",
                "actions": [
                    {
                        "sns": {
                            "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                        }
                    }
                ]
            }
        ]
    },
    "onInput": {
        "events": [
            {
                "eventName": "Overpressurized",
                "condition": "$input.PressureInput.sensorData.pressure
> 70",
                "actions": [
                    {
                        "setVariable": {

```

```

        "variableName": "pressureThresholdBreach",
        "value": "3"
    }
}
],
},
{
    "eventName": "Pressure Okay",
    "condition": "$input.PressureInput.sensorData.pressure
<= 70",
    "actions": [
        {
            "setVariable": {
                "variableName": "pressureThresholdBreach",
                "value":
"$variable.pressureThresholdBreach - 1"
            }
        }
    ]
}
],
},
"transitionEvents": [
    {
        "eventName": "BackToNormal",
        "condition": "$input.PressureInput.sensorData.pressure
<= 70 && $variable.pressureThresholdBreach <= 1",
        "nextState": "Normal"
    }
]
},
"onExit": {
    "events": [
        {
            "eventName": "Normal Pressure Restored",
            "condition": "true",
            "actions": [
                {
                    "sns": {
                        "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
                    }
                }
            ]
        }
    ]
}
}

```

```

    ]
  }
},
"initialStateName": "Normal"
},
"key": "motorid",
"roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole"
}

```

Salida:

```

{
  "detectorModelConfiguration": {
    "status": "ACTIVATING",
    "lastUpdateTime": 1560796816.077,
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "creationTime": 1560796816.077,
    "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/
motorDetectorModel",
    "key": "motorid",
    "detectorModelName": "motorDetectorModel",
    "detectorModelVersion": "1"
  }
}

```

Para obtener más información, consulte [CreateDetectorModel](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [CreateDetectorModel](#) la Referencia de AWS CLI comandos.

## create-input

En el siguiente ejemplo de código, se muestra cómo usar create-input.

### AWS CLI

Para crear una entrada

En el siguiente create-input ejemplo se crea una entrada.

```
aws iotevents create-input \
```

```
--cli-input-json file://pressureInput.json
```

Contenidos de `pressureInput.json`:

```
{
  "inputName": "PressureInput",
  "inputDescription": "Pressure readings from a motor",
  "inputDefinition": {
    "attributes": [
      { "jsonPath": "sensorData.pressure" },
      { "jsonPath": "motorid" }
    ]
  }
}
```

Salida:

```
{
  "inputConfiguration": {
    "status": "ACTIVE",
    "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
    "lastUpdateTime": 1560795312.542,
    "creationTime": 1560795312.542,
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor"
  }
}
```

Para obtener más información, consulte [CreateInput](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [CreateInput](#) la Referencia de AWS CLI comandos.

## **delete-detector-model**

En el siguiente ejemplo de código, se muestra cómo usar `delete-detector-model`.

### AWS CLI

Para eliminar un modelo de detector

En el siguiente `delete-detector-model` ejemplo, se elimina el modelo de detector especificado. También se eliminan todas las instancias activas del modelo de detector.

```
aws iotevents delete-detector-model \  
  --detector-model-name motorDetectorModel
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteDetectorModel](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [DeleteDetectorModel](#) la Referencia de AWS CLI comandos.

## delete-input

En el siguiente ejemplo de código, se muestra cómo usar `delete-input`.

AWS CLI

Para eliminar una entrada

En el siguiente `delete-input` ejemplo, se elimina la entrada especificada.

```
aws iotevents delete-input \  
  --input-name PressureInput
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteInput](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [DeleteInput](#) la Referencia de AWS CLI comandos.

## describe-detector-model

En el siguiente ejemplo de código, se muestra cómo usar `describe-detector-model`.

AWS CLI

Para obtener información sobre un modelo de detector

En el siguiente `describe-detector-model` ejemplo, se muestran los detalles del modelo de detector especificado. Como el `version` parámetro no está especificado, se devuelve información sobre la última versión.



```
aws iotevents describe-detector-model \
  --detector-model-name motorDetectorModel
```

Salida:

```
{
  "detectorModel": {
    "detectorModelConfiguration": {
      "status": "ACTIVE",
      "lastUpdateTime": 1560796816.077,
      "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
      "creationTime": 1560796816.077,
      "detectorModelArn": "arn:aws:iotevents:us-
west-2:123456789012:detectorModel/motorDetectorModel",
      "key": "motorid",
      "detectorModelName": "motorDetectorModel",
      "detectorModelVersion": "1"
    },
    "detectorModelDefinition": {
      "states": [
        {
          "onInput": {
            "transitionEvents": [
              {
                "eventName": "Overpressurized",
                "actions": [
                  {
                    "setVariable": {
                      "variableName":
"pressureThresholdBreached",
                      "value":
"$variable.pressureThresholdBreached + 3"
                    }
                  ]
                },
                "condition":
"$input.PressureInput.sensorData.pressure > 70",
                "nextState": "Dangerous"
              }
            ],
            "events": []
          },
          "stateName": "Normal",

```

```

        "onEnter": {
            "events": [
                {
                    "eventName": "init",
                    "actions": [
                        {
                            "setVariable": {
                                "variableName":
"pressureThresholdBreach",
                                "value": "0"
                            }
                        }
                    ],
                    "condition": "true"
                }
            ]
        },
        "onExit": {
            "events": []
        }
    },
    {
        "onInput": {
            "transitionEvents": [
                {
                    "eventName": "BackToNormal",
                    "actions": [],
                    "condition":
"$input.PressureInput.sensorData.pressure <= 70 &&
$variable.pressureThresholdBreach <= 1",
                    "nextState": "Normal"
                }
            ],
            "events": [
                {
                    "eventName": "Overpressurized",
                    "actions": [
                        {
                            "setVariable": {
                                "variableName":
"pressureThresholdBreach",
                                "value": "3"
                            }
                        }
                    ]
                }
            ]
        }
    }
}

```

```

        ],
        "condition":
"$input.PressureInput.sensorData.pressure > 70"
    },
    {
        "eventName": "Pressure Okay",
        "actions": [
            {
                "setVariable": {
                    "variableName":
"pressureThresholdBreached",
                    "value":
"$variable.pressureThresholdBreached - 1"
                }
            }
        ],
        "condition":
"$input.PressureInput.sensorData.pressure <= 70"
    }
]
},
"stateName": "Dangerous",
"onEnter": {
    "events": [
        {
            "eventName": "Pressure Threshold Breached",
            "actions": [
                {
                    "sns": {
                        "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                    }
                }
            ]
        },
        {
            "condition": "$variable.pressureThresholdBreached >
1"
        }
    ]
},
"onExit": {
    "events": [
        {
            "eventName": "Normal Pressure Restored",
            "actions": [

```

```
        {
            "sns": {
                "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
            }
        },
        "condition": "true"
    ]
}
]
}
]
}
}
}
}
}
```

Para obtener más información, consulte [DescribeDetectorModel](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [DescribeDetectorModel](#) la Referencia de AWS CLI comandos.

## describe-detector

En el siguiente ejemplo de código, se muestra cómo usar describe-detector.

### AWS CLI

Para obtener información sobre un detector (instancia).

En el siguiente describe-detector ejemplo, se muestran los detalles del detector (instancia) especificado.

```
aws iotevents-data describe-detector \
  --detector-model-name motorDetectorModel \
  --key-value "Fulton-A32"
```

Salida:

```
{
```

```
"detector": {
  "lastUpdateTime": 1560797852.776,
  "creationTime": 1560797852.775,
  "state": {
    "variables": [
      {
        "name": "pressureThresholdBreached",
        "value": "3"
      }
    ],
    "stateName": "Dangerous",
    "timers": []
  },
  "keyValue": "Fulton-A32",
  "detectorModelName": "motorDetectorModel",
  "detectorModelVersion": "1"
}
}
```

Para obtener más información, consulte [DescribeDetector](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [DescribeDetector](#) la Referencia de AWS CLI comandos.

## describe-input

En el siguiente ejemplo de código, se muestra cómo usar `describe-input`.

AWS CLI

Para obtener información sobre una entrada

En el siguiente `describe-input` ejemplo, se muestran los detalles de la entrada especificada.

```
aws iotevents describe-input \
  --input-name PressureInput
```

Salida:

```
{
```

```
"input": {
  "inputConfiguration": {
    "status": "ACTIVE",
    "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/
PressureInput",
    "lastUpdateTime": 1560795312.542,
    "creationTime": 1560795312.542,
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor"
  },
  "inputDefinition": {
    "attributes": [
      {
        "jsonPath": "sensorData.pressure"
      },
      {
        "jsonPath": "motorid"
      }
    ]
  }
}
```

Para obtener más información, consulte [DescribeInput](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [DescribeInput](#) la Referencia de AWS CLI comandos.

## describe-logging-options

En el siguiente ejemplo de código, se muestra cómo usar `describe-logging-options`.

### AWS CLI

Para obtener información sobre la configuración de registro

El siguiente `describe-logging-options` ejemplo recupera la configuración actual de las opciones de registro de AWS IoT Events.

```
aws iotevents describe-logging-options
```

Salida:

```
{
  "loggingOptions": {
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "enabled": false,
    "level": "ERROR"
  }
}
```

Para obtener más información, consulte [DescribeLoggingOptions](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [DescribeLoggingOptions](#) la Referencia de AWS CLI comandos.

## list-detector-model-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-detector-model-versions`.

### AWS CLI

Para obtener información sobre las versiones de un modelo de detector

En el siguiente `list-detector-model-versions` ejemplo se enumeran todas las versiones de un modelo de detector. Solo se devuelven los metadatos asociados a cada versión de modelo de detector.

```
aws iotevents list-detector-model-versions \
  --detector-model-name motorDetectorModel
```

Salida:

```
{
  "detectorModelVersionSummaries": [
    {
      "status": "ACTIVE",
      "lastUpdateTime": 1560796816.077,
      "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
      "creationTime": 1560796816.077,
      "detectorModelArn": "arn:aws:iotevents:us-
west-2:123456789012:detectorModel/motorDetectorModel",
      "detectorModelName": "motorDetectorModel",
    }
  ]
}
```

```
        "detectorModelVersion": "1"
      }
    ]
  }
```

Para obtener más información, consulte [ListDetectorModelVersions](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [ListDetectorModelVersions](#) la Referencia de AWS CLI comandos.

## **list-detector-models**

En el siguiente ejemplo de código, se muestra cómo usar `list-detector-models`.

### AWS CLI

Para obtener una lista de sus modelos de detectores

El siguiente `list-detector-models` ejemplo muestra una lista de los modelos de detectores que ha creado. Solo se devuelven los metadatos asociados a cada modelo de detector.

```
aws iotevents list-detector-models
```

Salida:

```
{
  "detectorModelSummaries": [
    {
      "detectorModelName": "motorDetectorModel",
      "creationTime": 1552072424.212
      "detectorModelDescription": "Detect overpressure in a motor."
    }
  ]
}
```

Para obtener más información, consulte [ListDetectorModels](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [ListDetectorModels](#) la Referencia de AWS CLI comandos.



## list-detectors

En el siguiente ejemplo de código, se muestra cómo usar `list-detectors`.

### AWS CLI

Para obtener una lista de detectores para un modelo de detector

En el siguiente `list-detectors` ejemplo, se enumeran los detectores (las instancias de un modelo de detector) de su cuenta.

```
aws iotevents-data list-detectors \  
  --detector-model-name motorDetectorModel
```

Salida:

```
{  
  "detectorSummaries": [  
    {  
      "lastUpdateTime": 1558129925.2,  
      "creationTime": 1552073155.527,  
      "state": {  
        "stateName": "Normal"  
      },  
      "keyValue": "Fulton-A32",  
      "detectorModelName": "motorDetectorModel",  
      "detectorModelVersion": "1"  
    }  
  ]  
}
```

Para obtener más información, consulte [ListDetectors](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [ListDetectors](#) la Referencia de AWS CLI comandos.

## list-inputs

En el siguiente ejemplo de código, se muestra cómo usar `list-inputs`.

### AWS CLI

Para enumerar las entradas

El siguiente `list-inputs` ejemplo muestra las entradas que ha creado en su cuenta.

```
aws iotevents list-inputs
```

Este comando no genera ninguna salida. Salida:

```
{
  {
    "status": "ACTIVE",
    "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
    "lastUpdateTime": 1551742986.768,
    "creationTime": 1551742986.768,
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor"
  }
}
```

Para obtener más información, consulte [ListInputs](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [ListInputs](#) la Referencia de AWS CLI comandos.

## `list-tags-for-resource`

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas asignadas a un recurso.

En el siguiente `list-tags-for-resource` ejemplo, se enumeran los nombres y valores de las claves de etiquetas que ha asignado al recurso.

```
aws iotevents list-tags-for-resource \
  --resource-arn "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput"
```

Salida:

```
{
  "tags": [
    {
      "value": "motor",
```

```

        "key": "deviceType"
      }
    ]
  }

```

Para obtener más información, consulte [ListTagsForResource](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## put-logging-options

En el siguiente ejemplo de código, se muestra cómo usar put-logging-options.

### AWS CLI

Para configurar las opciones de registro

El siguiente put-logging-options ejemplo establece o actualiza las opciones de registro de AWS IoT Events. Si actualiza el valor de cualquier loggingOptions` field, it can take up to one minute for the change to take effect. Also, if you change the policy attached to the role you specified in the ``roleArn campo (por ejemplo, para corregir una política no válida), el cambio puede tardar hasta cinco minutos en surtir efecto.

```

aws iotevents put-logging-options \
  --cli-input-json file://logging-options.json

```

Contenidos de logging-options.json:

```

{
  "loggingOptions": {
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "level": "DEBUG",
    "enabled": true,
    "detectorDebugOptions": [
      {
        "detectorModelName": "motorDetectorModel",
        "keyValue": "Fulton-A32"
      }
    ]
  }
}

```

```
    ]
  }
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [PutLoggingOptions](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [PutLoggingOptions](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir etiquetas a un recurso

En el siguiente `tag-resource` ejemplo, se agrega o modifica (si la clave `deviceType` ya existe) la etiqueta adjunta al recurso especificado.

```
aws iotevents tag-resource \  
  --cli-input-json file://pressureInput.tag.json
```

Contenidos de `pressureInput.tag.json`:

```
{  
  "resourceArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",  
  "tags": [  
    {  
      "key": "deviceType",  
      "value": "motor"  
    }  
  ]  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [TagResource](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Cómo eliminar etiquetas de un recurso

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta con el nombre de clave especificado del recurso especificado.

```
aws iotevents untag-resource \  
  --resource-arn arn:aws:iotevents:us-west-2:123456789012:input/PressureInput \  
  --tagkeys deviceType
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [UntagResource](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-detector-model

En el siguiente ejemplo de código, se muestra cómo usar `update-detector-model`.

### AWS CLI

Para actualizar un modelo de detector

El siguiente `update-detector-model` ejemplo actualiza el modelo de detector especificado. Los detectores (instancias) generados por la versión anterior se eliminan y, a continuación, se vuelven a crear a medida que llegan nuevas entradas.

```
aws iotevents update-detector-model \  
  --cli-input-json file://motorDetectorModel.update.json
```

## Contenidos de motorDetectorModel.update.json:

```

{
  "detectorModelName": "motorDetectorModel",
  "detectorModelDefinition": {
    "states": [
      {
        "stateName": "Normal",
        "onEnter": {
          "events": [
            {
              "eventName": "init",
              "condition": "true",
              "actions": [
                {
                  "setVariable": {
                    "variableName": "pressureThresholdBreached",
                    "value": "0"
                  }
                }
              ]
            }
          ],
          "onInput": {
            "transitionEvents": [
              {
                "eventName": "Overpressurized",
                "condition": "$input.PressureInput.sensorData.pressure >
70",
                "actions": [
                  {
                    "setVariable": {
                      "variableName": "pressureThresholdBreached",
                      "value":
"$variable.pressureThresholdBreached + 3"
                    }
                  }
                ],
                "nextState": "Dangerous"
              }
            ]
          }
        }
      ]
    }
  },

```

```

    {
      "stateName": "Dangerous",
      "onEnter": {
        "events": [
          {
            "eventName": "Pressure Threshold Breached",
            "condition": "$variable.pressureThresholdBreach > 1",
            "actions": [
              {
                "sns": {
                  "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                }
              }
            ]
          }
        ],
      },
      "onInput": {
        "events": [
          {
            "eventName": "Overpressurized",
            "condition": "$input.PressureInput.sensorData.pressure >
70",
            "actions": [
              {
                "setVariable": {
                  "variableName": "pressureThresholdBreach",
                  "value": "3"
                }
              }
            ]
          }
        ],
      },
      {
        "eventName": "Pressure Okay",
        "condition": "$input.PressureInput.sensorData.pressure
<= 70",
        "actions": [
          {
            "setVariable": {
              "variableName": "pressureThresholdBreach",
              "value":
"$variable.pressureThresholdBreach - 1"
            }
          }
        ]
      }
    }
  ]
}

```

```

    }
  ]
},
"transitionEvents": [
  {
    "eventName": "BackToNormal",
    "condition": "$input.PressureInput.sensorData.pressure
<= 70 && $variable.pressureThresholdBreached <= 1",
    "nextState": "Normal"
  }
],
"onExit": {
  "events": [
    {
      "eventName": "Normal Pressure Restored",
      "condition": "true",
      "actions": [
        {
          "sns": {
            "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
          }
        }
      ]
    }
  ]
}
},
"initialStateName": "Normal"
},
"roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole"
}

```

**Salida:**

```

{
  "detectorModelConfiguration": {
    "status": "ACTIVATING",
    "lastUpdateTime": 1560799387.719,
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
  }
}

```



```
    "creationTime": 1560799387.719,  
    "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/  
motorDetectorModel",  
    "key": "motorid",  
    "detectorModelName": "motorDetectorModel",  
    "detectorModelVersion": "2"  
  }  
}
```

Para obtener más información, consulte [UpdateDetectorModel](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [UpdateDetectorModel](#) la Referencia de AWS CLI comandos.

## update-input

En el siguiente ejemplo de código, se muestra cómo usar `update-input`.

### AWS CLI

Para actualizar una entrada

En el siguiente `update-input` ejemplo, se actualiza la entrada especificada con una nueva descripción y definición.

```
aws iotevents update-input \  
  --cli-input-json file://pressureInput.json
```

Contenidos de `pressureInput.json`:

```
{  
  "inputName": "PressureInput",  
  "inputDescription": "Pressure readings from a motor",  
  "inputDefinition": {  
    "attributes": [  
      { "jsonPath": "sensorData.pressure" },  
      { "jsonPath": "motorid" }  
    ]  
  }  
}
```

Salida:

```
{
  "inputConfiguration": {
    "status": "ACTIVE",
    "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
    "lastUpdateTime": 1560795976.458,
    "creationTime": 1560795312.542,
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor"
  }
}
```

Para obtener más información, consulte [UpdateInput](#) la API Referencia de eventos de AWS IoT.

- Para API obtener más información, consulte [UpdateInput](#) la Referencia de AWS CLI comandos.

## AWS IoT Events-Data ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS IoT Events-Data.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **batch-put-message**

En el siguiente ejemplo de código, se muestra cómo usar `batch-put-message`.

AWS CLI

Para enviar mensajes (entradas) a AWS IoT Events

El siguiente `batch-put-message` ejemplo envía un conjunto de mensajes al sistema AWS IoT Events. La carga útil de cada mensaje se transforma en la entrada que especifique (`inputName`) y se ingiere en cualquier detector que supervise esa entrada. Si se envían varios mensajes, no se garantiza el orden en que se procesan los mensajes. Para garantizar el pedido, debes enviar los mensajes de uno en uno y esperar a que te respondan correctamente.

```
aws iotevents-data batch-put-message \  
  --cli-binary-format raw-in-base64-out \  
  --cli-input-json file://highPressureMessage.json
```

Contenidos de `highPressureMessage.json`:

```
{  
  "messages": [  
    {  
      "messageId": "00001",  
      "inputName": "PressureInput",  
      "payload": "{\"motorid\": \"Fulton-A32\", \"sensorData\": {\"pressure\":  
80, \"temperature\": 39} }"  
    }  
  ]  
}
```

Salida:

```
{  
  "BatchPutMessageErrorEntries": []  
}
```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([BatchPutMessage](#) en inglés) \*.

- Para API obtener más información, consulte [BatchPutMessage](#) la Referencia de AWS CLI comandos.

## **batch-update-detector**

En el siguiente ejemplo de código, se muestra cómo usar `batch-update-detector`.

## AWS CLI

Para actualizar un detector (instancia)

El siguiente `batch-update-detector` ejemplo actualiza el estado, los valores de las variables y la configuración del temporizador de uno o más detectores (instancias) de un modelo de detector específico.

```
aws iotevents-data batch-update-detector \  
  --cli-input-json file://budFulton-A32.json
```

Contenidos de `budFulton-A32.json`:

```
{  
  "detectors": [  
    {  
      "messageId": "00001",  
      "detectorModelName": "motorDetectorModel",  
      "keyValue": "Fulton-A32",  
      "state": {  
        "stateName": "Normal",  
        "variables": [  
          {  
            "name": "pressureThresholdBreach",  
            "value": "0"  
          }  
        ],  
        "timers": [  
        ]  
      }  
    }  
  ]  
}
```

Salida:

```
{  
  "batchUpdateDetectorErrorEntries": []  
}
```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([BatchUpdateDetector](#) en inglés) \*.

- Para API obtener más información, consulte [BatchUpdateDetector](#) la Referencia de AWS CLI comandos.

## create-detector-model

En el siguiente ejemplo de código, se muestra cómo usar `create-detector-model`.

### AWS CLI

Para crear un modelo de detector

El siguiente `create-detector-model` ejemplo crea un modelo de detector.

```
aws iotevents create-detector-model \  
  --cli-input-json file://motorDetectorModel.json
```

Contenidos de `motorDetectorModel.json`:

```
{  
  "detectorModelName": "motorDetectorModel",  
  "detectorModelDefinition": {  
    "states": [  
      {  
        "stateName": "Normal",  
        "onEnter": {  
          "events": [  
            {  
              "eventName": "init",  
              "condition": "true",  
              "actions": [  
                {  
                  "setVariable": {  
                    "variableName": "pressureThresholdBreach",  
                    "value": "0"  
                  }  
                }  
              ]  
            }  
          ]  
        }  
      ],  
    },  
    "onInput": {  
      "transitionEvents": [  
        {
```

```

        "eventName": "Overpressurized",
        "condition": "$input.PressureInput.sensorData.pressure
> 70",
        "actions": [
            {
                "setVariable": {
                    "variableName": "pressureThresholdBreach",
                    "value":
"$variable.pressureThresholdBreach + 3"
                }
            }
        ],
        "nextState": "Dangerous"
    }
]
},
{
    "stateName": "Dangerous",
    "onEnter": {
        "events": [
            {
                "eventName": "Pressure Threshold Breached",
                "condition": "$variable.pressureThresholdBreach >
1",
                "actions": [
                    {
                        "sns": {
                            "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                        }
                    }
                ]
            }
        ]
    },
    "onInput": {
        "events": [
            {
                "eventName": "Overpressurized",
                "condition": "$input.PressureInput.sensorData.pressure
> 70",
                "actions": [
                    {

```

```

        "setVariable": {
            "variableName": "pressureThresholdBreached",
            "value": "3"
        }
    ]
},
{
    "eventName": "Pressure Okay",
    "condition": "$input.PressureInput.sensorData.pressure
<= 70",
    "actions": [
        {
            "setVariable": {
                "variableName": "pressureThresholdBreached",
                "value":
"$variable.pressureThresholdBreached - 1"
            }
        }
    ]
},
"transitionEvents": [
    {
        "eventName": "BackToNormal",
        "condition": "$input.PressureInput.sensorData.pressure
<= 70 && $variable.pressureThresholdBreached <= 1",
        "nextState": "Normal"
    }
],
"onExit": {
    "events": [
        {
            "eventName": "Normal Pressure Restored",
            "condition": "true",
            "actions": [
                {
                    "sns": {
                        "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
                    }
                }
            ]
        }
    ]
}
]

```

```

    }
  ]
}
},
"initialStateName": "Normal"
},
"key": "motorid",
"roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole"
}

```

Salida:

```

{
  "detectorModelConfiguration": {
    "status": "ACTIVATING",
    "lastUpdateTime": 1560796816.077,
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "creationTime": 1560796816.077,
    "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/
motorDetectorModel",
    "key": "motorid",
    "detectorModelName": "motorDetectorModel",
    "detectorModelVersion": "1"
  }
}

```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([CreateDetectorModel](#) en inglés) \*.

- Para API obtener más información, consulte [CreateDetectorModel](#) la Referencia de AWS CLI comandos.

## create-input

En el siguiente ejemplo de código, se muestra cómo usar `create-input`.

### AWS CLI

Para crear una entrada

El siguiente `create-input` ejemplo crea una entrada.



```
aws iotevents create-input \  
  --cli-input-json file://pressureInput.json
```

Contenidos de `pressureInput.json`:

```
{  
  "inputName": "PressureInput",  
  "inputDescription": "Pressure readings from a motor",  
  "inputDefinition": {  
    "attributes": [  
      { "jsonPath": "sensorData.pressure" },  
      { "jsonPath": "motorid" }  
    ]  
  }  
}
```

Salida:

```
{  
  "inputConfiguration": {  
    "status": "ACTIVE",  
    "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",  
    "lastUpdateTime": 1560795312.542,  
    "creationTime": 1560795312.542,  
    "inputName": "PressureInput",  
    "inputDescription": "Pressure readings from a motor"  
  }  
}
```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([CreateInput](#) en inglés) \*.

- Para API obtener más información, consulte [CreateInput](#) la Referencia de AWS CLI comandos.

## **delete-detector-model**

En el siguiente ejemplo de código, se muestra cómo usar `delete-detector-model`.

### AWS CLI

Para eliminar un modelo de detector

El siguiente `delete-detector-model` ejemplo elimina un modelo de detector. También se eliminan todas las instancias activas del modelo de detector.

```
aws iotevents delete-detector-model \  
  --detector-model-name motorDetectorModel*
```

Este comando no genera ninguna salida.

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([DeleteDetectorModel](#) en inglés) \*.

- Para API obtener más información, consulte [DeleteDetectorModel](#) la Referencia de AWS CLI comandos.

## **delete-input**

En el siguiente ejemplo de código, se muestra cómo usar `delete-input`.

AWS CLI

Para eliminar una entrada

En el siguiente `delete-input` ejemplo, se elimina una entrada.

```
aws iotevents delete-input \  
  --input-name PressureInput
```

Este comando no genera ninguna salida.

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([DeleteInput](#) en inglés) \*.

- Para API obtener más información, consulte [DeleteInput](#) la Referencia de AWS CLI comandos.

## **describe-detector-model**

En el siguiente ejemplo de código, se muestra cómo usar `describe-detector-model`.

AWS CLI

Para obtener información sobre un modelo de detector

El siguiente describe-detector-model ejemplo describe un modelo de detector. Si no se especifica el version parámetro, el comando devuelve información sobre la versión más reciente.

```
aws iotevents describe-detector-model \  
  --detector-model-name motorDetectorModel
```

Salida:

```
{  
  "detectorModel": {  
    "detectorModelConfiguration": {  
      "status": "ACTIVE",  
      "lastUpdateTime": 1560796816.077,  
      "roleArn": "arn:aws:iam:123456789012:role/IoTEventsRole",  
      "creationTime": 1560796816.077,  
      "detectorModelArn": "arn:aws:iotevents:us-  
west-2:123456789012:detectorModel/motorDetectorModel",  
      "key": "motorid",  
      "detectorModelName": "motorDetectorModel",  
      "detectorModelVersion": "1"  
    },  
    "detectorModelDefinition": {  
      "states": [  
        {  
          "onInput": {  
            "transitionEvents": [  
              {  
                "eventName": "Overpressurized",  
                "actions": [  
                  {  
                    "setVariable": {  
                      "variableName":  
"pressureThresholdBreached",  
                      "value":  
"$variable.pressureThresholdBreached + 3"  
                    }  
                  }  
                ],  
                "condition":  
"$input.PressureInput.sensorData.pressure > 70",  
                "nextState": "Dangerous"  
              }  
            ]  
          }  
        ]  
      }  
    }  
  }  
}
```

```

    ],
    "events": []
  },
  "stateName": "Normal",
  "onEnter": {
    "events": [
      {
        "eventName": "init",
        "actions": [
          {
            "setVariable": {
              "variableName":
"pressureThresholdBreached",
              "value": "0"
            }
          }
        ],
        "condition": "true"
      }
    ]
  },
  "onExit": {
    "events": []
  }
},
{
  "onInput": {
    "transitionEvents": [
      {
        "eventName": "BackToNormal",
        "actions": [],
        "condition":
"$input.PressureInput.sensorData.pressure <= 70 &&
$variable.pressureThresholdBreached <= 1",
        "nextState": "Normal"
      }
    ],
    "events": [
      {
        "eventName": "Overpressurized",
        "actions": [
          {
            "setVariable": {

```

```

        "variableName":
"pressureThresholdBreached",
        "value": "3"
    }
    ],
    "condition":
"$input.PressureInput.sensorData.pressure > 70"
    },
    {
        "eventName": "Pressure Okay",
        "actions": [
            {
                "setVariable": {
                    "variableName":
"pressureThresholdBreached",
                    "value":
"$variable.pressureThresholdBreached - 1"
                }
            }
        ],
        "condition":
"$input.PressureInput.sensorData.pressure <= 70"
    }
    ],
    },
    "stateName": "Dangerous",
    "onEnter": {
        "events": [
            {
                "eventName": "Pressure Threshold Breached",
                "actions": [
                    {
                        "sns": {
                            "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                        }
                    }
                ],
                "condition": "$variable.pressureThresholdBreached >
1"
            }
        ]
    },

```

```

        "onExit": {
            "events": [
                {
                    "eventName": "Normal Pressure Restored",
                    "actions": [
                        {
                            "sns": {
                                "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
                            }
                        }
                    ],
                    "condition": "true"
                }
            ]
        }
    ],
    "initialStateName": "Normal"
}
}
}
}

```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([DescribeDetectorModel](#) en inglés) \*.

- Para API obtener más información, consulte [DescribeDetectorModel](#) la Referencia de AWS CLI comandos.

## describe-detector

En el siguiente ejemplo de código, se muestra cómo usar `describe-detector`.

### AWS CLI

Para obtener información sobre un detector (instancia)

El siguiente `describe-detector` ejemplo devuelve información sobre el detector (instancia) especificado.

```

aws iotevents-data describe-detector \
  --detector-model-name motorDetectorModel \

```

```
--key-value Fulton-A32
```

Salida:

```
{
  "detector": {
    "lastUpdateTime": 1560797852.776,
    "creationTime": 1560797852.775,
    "state": {
      "variables": [
        {
          "name": "pressureThresholdBreach",
          "value": "3"
        }
      ],
      "stateName": "Dangerous",
      "timers": []
    },
    "keyValue": "Fulton-A32",
    "detectorModelName": "motorDetectorModel",
    "detectorModelVersion": "1"
  }
}
```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([DescribeDetector](#) en inglés) \*.

- Para API obtener más información, consulte [DescribeDetector](#) la Referencia de AWS CLI comandos.

## describe-input

En el siguiente ejemplo de código, se muestra cómo usar describe-input.

AWS CLI

Para obtener información sobre una entrada

El siguiente describe-input ejemplo recupera los detalles de una entrada.

```
aws iotevents describe-input \  
  --input-name PressureInput
```

**Salida:**

```
{
  "input": {
    "inputConfiguration": {
      "status": "ACTIVE",
      "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/
PressureInput",
      "lastUpdateTime": 1560795312.542,
      "creationTime": 1560795312.542,
      "inputName": "PressureInput",
      "inputDescription": "Pressure readings from a motor"
    },
    "inputDefinition": {
      "attributes": [
        {
          "jsonPath": "sensorData.pressure"
        },
        {
          "jsonPath": "motorid"
        }
      ]
    }
  }
}
```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([DescribeInput](#) en inglés) \*.

- Para API obtener más información, consulte [DescribeInput](#) la Referencia de AWS CLI comandos.

**describe-logging-options**

En el siguiente ejemplo de código, se muestra cómo usar `describe-logging-options`.

**AWS CLI**

Para obtener información sobre la configuración de registro

El siguiente `describe-logging-options` ejemplo recupera las opciones de registro actuales de AWS IoT Events.



```
aws iotevents describe-logging-options
```

Salida:

```
{
  "loggingOptions": {
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "enabled": false,
    "level": "ERROR"
  }
}
```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([DescribeLoggingOptions](#) en inglés) \*.

- Para API obtener más información, consulte [DescribeLoggingOptions](#) la Referencia de AWS CLI comandos.

## list-detector-model-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-detector-model-versions`.

AWS CLI

Para obtener información sobre las versiones de un modelo de detector

En el siguiente `list-detector-model-versions` ejemplo se enumeran todas las versiones de un modelo de detector. Solo se devuelven los metadatos asociados a cada versión de modelo de detector.

```
aws iotevents list-detector-model-versions \
  --detector-model-name motorDetectorModel
```

Salida:

```
{
  "detectorModelVersionSummaries": [
    {
      "status": "ACTIVE",
      "lastUpdateTime": 1560796816.077,
      "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    }
  ]
}
```

```
        "creationTime": 1560796816.077,  
        "detectorModelArn": "arn:aws:iotevents:us-  
west-2:123456789012:detectorModel/motorDetectorModel",  
        "detectorModelName": "motorDetectorModel",  
        "detectorModelVersion": "1"  
    }  
]  
}
```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([ListDetectorModelVersions](#) en inglés) \*.

- Para API obtener más información, consulte [ListDetectorModelVersions](#) la Referencia de AWS CLI comandos.

## **list-detector-models**

En el siguiente ejemplo de código, se muestra cómo usar `list-detector-models`.

### AWS CLI

Para obtener una lista de sus modelos de detectores

El siguiente `list-detector-models` ejemplo muestra los modelos de detectores que ha creado. Solo se devuelven los metadatos asociados a cada modelo de detector.

```
aws iotevents list-detector-models
```

Salida:

```
{  
  "detectorModelSummaries": [  
    {  
      "detectorModelName": "motorDetectorModel",  
      "creationTime": 1552072424.212  
      "detectorModelDescription": "Detect overpressure in a motor."  
    }  
  ]  
}
```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([ListDetectorModels](#) en inglés) \*.

- Para API obtener más información, consulte [ListDetectorModels](#) la Referencia de AWS CLI comandos.

## list-detectors

En el siguiente ejemplo de código, se muestra cómo usar `list-detectors`.

### AWS CLI

Para obtener una lista de detectores para un modelo de detector

El siguiente `list-detectors` ejemplo muestra los detectores (las instancias de un modelo de detector).

```
aws iotevents-data list-detectors \  
  --detector-model-name motorDetectorModel
```

Salida:

```
{  
  "detectorSummaries": [  
    {  
      "lastUpdateTime": 1558129925.2,  
      "creationTime": 1552073155.527,  
      "state": {  
        "stateName": "Normal"  
      },  
      "keyValue": "Fulton-A32",  
      "detectorModelName": "motorDetectorModel",  
      "detectorModelVersion": "1"  
    }  
  ]  
}
```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([ListDetectors](#) en inglés) \*.

- Para API obtener más información, consulte [ListDetectors](#) la Referencia de AWS CLI comandos.

## list-inputs

En el siguiente ejemplo de código, se muestra cómo usar `list-inputs`.

## AWS CLI

Para enumerar las entradas

El siguiente `list-inputs` ejemplo muestra las entradas que ha creado.

```
aws iotevents list-inputs
```

Salida:

```
{
  "status": "ACTIVE",
  "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
  "lastUpdateTime": 1551742986.768,
  "creationTime": 1551742986.768,
  "inputName": "PressureInput",
  "inputDescription": "Pressure readings from a motor"
}
```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([ListInputs](#) en inglés) \*.

- Para API obtener más información, consulte [ListInputs](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas asignadas a un recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas (metadatos) que ha asignado al recurso.

```
aws iotevents list-tags-for-resource \
  --resource-arn "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput"
```

Salida:

```
{
  "tags": [
```

```
{
  "value": "motor",
  "key": "deviceType"
}
]
```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([ListTagsForResource](#) en inglés) \*.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## put-logging-options

En el siguiente ejemplo de código, se muestra cómo usar `put-logging-options`.

### AWS CLI

Para configurar las opciones de registro

El siguiente `list-tags-for-resource` ejemplo establece o actualiza las opciones de registro de AWS IoT Events. Si actualiza el valor de cualquier campo `loggingOptions`, los cambios tardarán hasta un minuto en surtir efecto. Además, si cambias la política asociada a la función que especificaste en el `roleArn` campo (por ejemplo, para corregir una política no válida), el cambio tardará hasta cinco minutos en surtir efecto.

```
aws iotevents put-logging-options \
  --cli-input-json file://logging-options.json
```

Contenidos de `logging-options.json`:

```
{
  "loggingOptions": {
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "level": "DEBUG",
    "enabled": true,
    "detectorDebugOptions": [
      {
        "detectorModelName": "motorDetectorModel",
        "keyValue": "Fulton-A32"
      }
    ]
  }
}
```

```
    ]
  }
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([PutLoggingOptions](#) en inglés) \*.

- Para API obtener más información, consulte [PutLoggingOptions](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir etiquetas a un recurso

El siguiente `tag-resource` ejemplo agrega o modifica las etiquetas del recurso dado. Las etiquetas son metadatos que se pueden usar para administrar un recurso.

```
aws iotevents tag-resource \  
  --cli-input-json file://pressureInput.tag.json
```

Contenidos de `pressureInput.tag.json`:

```
{  
  "resourceArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",  
  "tags": [  
    {  
      "key": "deviceType",  
      "value": "motor"  
    }  
  ]  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([TagResource](#) en inglés) \*.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Cómo eliminar etiquetas de un recurso

En el siguiente `untag-resource` ejemplo, se eliminan las etiquetas especificadas del recurso.

```
aws iotevents untag-resource \  
  --cli-input-json file://pressureInput.untag.json
```

Contenidos de `pressureInput.untag.json`:

```
{  
  "resourceArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",  
  "tagKeys": [  
    "deviceType"  
  ]  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([UntagResource](#) en inglés) \*.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-detector-model

En el siguiente ejemplo de código, se muestra cómo usar `update-detector-model`.

### AWS CLI

Para actualizar un modelo de detector

El siguiente `update-detector-model` ejemplo actualiza un modelo de detector. Los detectores (instancias) generados por la versión anterior se eliminan y, a continuación, se vuelven a crear a medida que llegan nuevas entradas.

```
aws iotevents update-detector-model \  
  --cli-input-json file://motorDetectorModel.update.json
```

Contenido de `.update.json` `motorDetectorModel`:

```
{  
  "detectorModelName": "motorDetectorModel",  
  "detectorModelDefinition": {  
    "states": [  
      {  
        "stateName": "Normal",  
        "onEnter": {  
          "events": [  
            {  
              "eventName": "init",  
              "condition": "true",  
              "actions": [  
                {  
                  "setVariable": {  
                    "variableName": "pressureThresholdBreach",  
                    "value": "0"  
                  }  
                }  
              ]  
            }  
          ]  
        },  
        "onInput": {  
          "transitionEvents": [  
            {  
              "eventName": "Overpressurized",  
              "condition": "$input.PressureInput.sensorData.pressure > 70",  
              "actions": [  
                {  
                  "setVariable": {  
                    "variableName": "pressureThresholdBreach",  
                    "value": "$variable.pressureThresholdBreach + 3"  
                  }  
                }  
              ]  
            }  
          ]  
        }  
      ]  
    }  
  }  
}
```



```
    ],
    "nextState": "Dangerous"
  }
]
},
{
  "stateName": "Dangerous",
  "onEnter": {
    "events": [
      {
        "eventName": "Pressure Threshold Breached",
        "condition": "$variable.pressureThresholdBreach > 1",
        "actions": [
          {
            "sns": {
              "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
            }
          }
        ]
      }
    ]
  },
  "onInput": {
    "events": [
      {
        "eventName": "Overpressurized",
        "condition": "$input.PressureInput.sensorData.pressure > 70",
        "actions": [
          {
            "setVariable": {
              "variableName": "pressureThresholdBreach",
              "value": "3"
            }
          }
        ]
      }
    ],
    {
      "eventName": "Pressure Okay",
      "condition": "$input.PressureInput.sensorData.pressure <= 70",
      "actions": [
        {
          "setVariable": {
```

```

        "variableName": "pressureThresholdBreached",
        "value": "$variable.pressureThresholdBreached - 1"
      }
    ]
  },
  "transitionEvents": [
    {
      "eventName": "BackToNormal",
      "condition": "$input.PressureInput.sensorData.pressure <= 70 &&
$variable.pressureThresholdBreached <= 1",
      "nextState": "Normal"
    }
  ],
  "onExit": {
    "events": [
      {
        "eventName": "Normal Pressure Restored",
        "condition": "true",
        "actions": [
          {
            "sns": {
              "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
            }
          }
        ]
      }
    ]
  },
  "initialStateName": "Normal"
},
"roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole"
}

```

Salida:

```

{
  "detectorModelConfiguration": {

```

```

    "status": "ACTIVATING",
    "lastUpdateTime": 1560799387.719,
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
    "creationTime": 1560799387.719,
    "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/
motorDetectorModel",
    "key": "motorid",
    "detectorModelName": "motorDetectorModel",
    "detectorModelVersion": "2"
  }
}

```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([UpdateDetectorModel](#) en inglés) \*.

- Para API obtener más información, consulte [UpdateDetectorModel](#) la Referencia de AWS CLI comandos.

## update-input

En el siguiente ejemplo de código, se muestra cómo usar `update-input`.

### AWS CLI

Para actualizar una entrada

El siguiente `update-input` ejemplo actualiza una entrada.

```

aws iotevents update-input \
  --cli-input-json file://pressureInput.json

```

Contenidos de `pressureInput.json`:

```

{
  "inputName": "PressureInput",
  "inputDescription": "Pressure readings from a motor",
  "inputDefinition": {
    "attributes": [
      { "jsonPath": "sensorData.pressure" },
      { "jsonPath": "motorid" }
    ]
  }
}

```

```
}
```

Salida:

```
{
  "inputConfiguration": {
    "status": "ACTIVE",
    "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
    "lastUpdateTime": 1560795976.458,
    "creationTime": 1560795312.542,
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor"
  }
}
```

Para obtener más información, consulte la Guía para desarrolladores de AWS IoT Events ([UpdateInput](#) en inglés) \*.

- Para API obtener más información, consulte [UpdateInput](#) la Referencia de AWS CLI comandos.

## AWS IoT Greengrass ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS IoT Greengrass.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

Acciones

### **associate-role-to-group**

En el siguiente ejemplo de código, se muestra cómo usar `associate-role-to-group`.

## AWS CLI

Para asociar un rol a un grupo de Greengrass

El siguiente `associate-role-to-group` ejemplo asocia el IAM rol especificado a un grupo de Greengrass. Las funciones y conectores de Lambda locales utilizan el rol de grupo para acceder AWS a los servicios. Por ejemplo, su rol de grupo puede conceder los permisos necesarios para la integración de CloudWatch los registros.

```
aws greengrass associate-role-to-group \  
  --group-id 2494ee3f-7f8a-4e92-a78b-d205f808b84b \  
  --role-arn arn:aws:iam::123456789012:role/GG-Group-Role
```

Salida:

```
{  
  "AssociatedAt": "2019-09-10T20:03:30Z"  
}
```

Para obtener más información, consulte [Configurar el rol de grupo](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte [AssociateRoleToGroup](#) la Referencia de AWS CLI comandos.

## `associate-service-role-to-account`

En el siguiente ejemplo de código, se muestra cómo usar `associate-service-role-to-account`.

## AWS CLI

Para asociar un rol de servicio a tu AWS cuenta

El siguiente `associate-service-role-to-account` ejemplo asocia un rol de IAM servicio, especificado por su ARN, con AWS IoT Greengrass en su AWS cuenta. Debe haber creado previamente el rol de servicio en IAM y debe asociarle un documento de política que permita a AWS IoT Greengrass asumir este rol.

```
aws greengrass associate-service-role-to-account \  
  --role-arn "arn:aws:iam::123456789012:role/service-role/Greengrass_ServiceRole"
```

Salida:

```
{
  "AssociatedAt": "2019-06-25T18:12:45Z"
}
```

Para obtener más información, consulte el [rol de servicio de Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [AssociateServiceRoleToAccount](#) de AWS CLI comandos.

## create-connector-definition-version

En el siguiente ejemplo de código, se muestra cómo usar `create-connector-definition-version`.

### AWS CLI

Para crear una versión de definición de conector

El siguiente `create-connector-definition-version` ejemplo crea una versión de definición de conector y la asocia a la definición de conector especificada. Todos los conectores de una versión definen valores para sus parámetros.

```
aws greengrass create-connector-definition-version \
  --connector-definition-id "55d0052b-0d7d-44d6-b56f-21867215e118" \
  --connectors "[{\\"Id\\": \\"MyTwilioNotificationsConnector\\",
  \\"ConnectorArn\\": \\"arn:aws:greengrass:us-west-2::/connectors/
  TwilioNotifications/versions/2\\", \\"Parameters\\": {\\"TWILIO_ACCOUNT_SID
  \": \\"AC1a8d4204890840d7fc482aab38090d57\\", \\"TwilioAuthTokenSecretArn\\":
  \\"arn:aws:secretsmanager:us-west-2:123456789012:secret:greengrass-TwilioAuthToken-
  ntSlp6\\", \\"TwilioAuthTokenSecretArn-ResourceId\\": \\"TwilioAuthToken\\",
  \\"DefaultFromPhoneNumber\\": \\"4254492999\\"}]]]"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
  connectors/55d0052b-0d7d-44d6-b56f-21867215e118/versions/33f709a0-c825-49cb-9eea-
  dc8964fbd635",
```

```

    "CreationTimestamp": "2019-06-24T20:46:30.134Z",
    "Id": "55d0052b-0d7d-44d6-b56f-21867215e118",
    "Version": "33f709a0-c825-49cb-9eea-dc8964fbd635"
}

```

- Para API obtener más información, consulte [CreateConnectorDefinitionVersion](#) la Referencia de AWS CLI comandos.

## create-connector-definition

En el siguiente ejemplo de código, se muestra cómo usar `create-connector-definition`.

### AWS CLI

Para crear una definición de conector

En el siguiente `create-connector-definition` ejemplo, se crea una definición de conector y una versión inicial de la definición de conector. La versión inicial contiene un conector. Todos los conectores de una versión definen valores para sus parámetros.

```

aws greengrass create-connector-definition \
  --name MySNSConnector \
  --initial-version '{"Connectors\": [{\"Id\": \"MySNSConnector\", \"ConnectorArn\": \"arn:aws:greengrass:us-west-2:/connectors/SNS/versions/1\", \"Parameters\": {\"DefaultSNSArn\": \"arn:aws:sns:us-west-2:123456789012:GGConnectorTopic\"}}]}'

```

Salida:

```

{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
  "CreationTimestamp": "2019-06-19T19:30:01.300Z",
  "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
  "LastUpdatedTimestamp": "2019-06-19T19:30:01.300Z",
  "LatestVersion": "63c57963-c7c2-4a26-a7e2-7bf478ea2623",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/versions/63c57963-c7c2-4a26-a7e2-7bf478ea2623",
  "Name": "MySNSConnector"
}

```

Para obtener más información, consulte [Introducción a los conectores Greengrass \(CLI\)](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [CreateConnectorDefinition](#) de AWS CLI comandos.

## **create-core-definition-version**

En el siguiente ejemplo de código, se muestra cómo usar `create-core-definition-version`.

### AWS CLI

Para crear una versión de definición básica

El siguiente `create-core-definition-version` ejemplo crea una versión de definición básica y la asocia a la definición básica especificada. La versión solo puede contener un núcleo. Antes de poder crear un núcleo, primero debe crear y aprovisionar el elemento de AWS IoT correspondiente. Este proceso incluye los siguientes `iot` comandos, que devuelven el `ThingArn` y `CertificateArn` necesario para el `create-core-definition-version` comando.

Cree el AWS elemento de IoT que corresponda al dispositivo principal:

```
aws iot create-thing \  
  --thing-name "MyCoreDevice"
```

Salida:

```
{  
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyCoreDevice",  
  "thingName": "MyCoreDevice",  
  "thingId": "cb419a19-9099-4515-9cec-e9b0e760608a"  
}
```

Cree las claves públicas y privadas y el certificado del dispositivo principal para el dispositivo. En este ejemplo, se usa el `create-keys-and-certificate` comando y se requieren permisos de escritura en el directorio actual. También puede utilizar el comando `create-certificate-from-csr`.

```
aws iot create-keys-and-certificate \  
  --set-as-active \  
  --certificate-key-pem-path <path>
```



```
--certificate-pem-outfile "myCore.cert.pem" \
--public-key-outfile "myCore.public.key" \
--private-key-outfile "myCore.private.key"
```

Salida:

```
{
  "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz",
  "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCAKgGawIBATgIUCgq6EGqou6zFqWgIZRndgQEFW+gwDQYJKoZIhvc...KdGewQS\n-----END
CERTIFICATE-----\n",
  "keyPair": {
    "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBzrqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAAqKpRgnn6yq26U3y...wIDAQAB\n-----END
PUBLIC KEY-----\n",
    "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIABAKCAQEAAqKpRgnn6yq26U3yt5YFZquyukfRjBMXDcNOK4rMCxDR...fvY4+te\n-----END
RSA PRIVATE KEY-----\n"
  },
  "certificateId":
  "123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
}
```

Cree una política de AWS IoT que permita `iot` y `greengrass` actuar. Para simplificar, la siguiente política permite realizar acciones en todos los recursos, pero su política debería ser más restrictiva.

```
aws iot create-policy \
  --policy-name "Core_Devices" \
  --policy-document "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\": \"Allow\", \"Action\": [\"iot:Publish\", \"iot:Subscribe\", \"iot:Connect
\", \"iot:Receive\"], \"Resource\": [\"*\"]}, {\"Effect\": \"Allow\", \"Action\":
[\"iot:GetThingShadow\", \"iot:UpdateThingShadow\", \"iot>DeleteThingShadow\"],
\"Resource\": [\"*\"]}, {\"Effect\": \"Allow\", \"Action\": [\"greengrass:*\"], \"Resource
\": [\"*\"]}]}\""
```

Salida:

```
{
  "policyName": "Core_Devices",
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/Core_Devices",
```

```

    "policyDocument": "{ \"Version\": \"2012-10-17\", \"Statement\": [{ \"Effect
  \": \"Allow\", \"Action\": [ \"iot:Publish\", \"iot:Subscribe\", \"iot:Connect
  \", \"iot:Receive\" ], \"Resource\": [ \"*\" ] }, { \"Effect\": \"Allow\", \"Action\":
  [ \"iot:GetThingShadow\", \"iot:UpdateThingShadow\", \"iot:DeleteThingShadow\" ],
  \"Resource\": [ \"*\" ] }, { \"Effect\": \"Allow\", \"Action\": [ \"greengrass:*\" ], \"Resource
  \": [ \"*\" ] } ] }",
    "policyVersionId": "1"
  }

```

Adjunte la política al certificado:

```

aws iot attach-policy \
  --policy-name "Core_Devices" \
  --target "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"

```

Este comando no genera ninguna salida.

Adjunte lo siguiente al certificado:

```

aws iot attach-thing-principal \
  --thing-name "MyCoreDevice" \
  --principal "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"

```

Este comando no genera ninguna salida.

Cree la versión de definición básica:

```

aws greengrass create-core-definition-version \
  --core-definition-id "582efe12-b05a-409e-9a24-a2ba1bcc4a12" \
  --cores "[ { \"Id\": \"MyCoreDevice\", \"ThingArn\": \"arn:aws:iot:us-
west-2:123456789012:thing/MyCoreDevice\", \"CertificateArn\": \"arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz
\", \"SyncShadow\": true } ]"

```

Salida:

```

{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
cores/582efe12-b05a-409e-9a24-a2ba1bcc4a12/versions/3fdc1190-2ce5-44de-b98b-
eec8f9571014",

```

```
"Version": "3fdc1190-2ce5-44de-b98b-eec8f9571014",  
"CreationTimestamp": "2019-09-18T00:15:09.838Z",  
"Id": "582efe12-b05a-409e-9a24-a2ba1bcc4a12"  
}
```

Para obtener más información, consulte [Configurar el núcleo de AWS IoT Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [CreateCoreDefinitionVersion](#) de AWS CLI comandos.

## create-core-definition

En el siguiente ejemplo de código, se muestra cómo usar `create-core-definition`.

### AWS CLI

Ejemplo 1: Para crear una definición básica vacía

El siguiente `create-core-definition` ejemplo crea una definición básica de Greengrass vacía (sin versión inicial). Antes de poder utilizar el núcleo, debe utilizar el `create-core-definition-version` comando para proporcionar los demás parámetros del núcleo.

```
aws greengrass create-core-definition \  
  --name cliGroup_Core
```

Salida:

```
{  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/  
b5c08008-54cb-44bd-9eec-c121b04283b5",  
  "CreationTimestamp": "2019-06-25T18:23:22.106Z",  
  "Id": "b5c08008-54cb-44bd-9eec-c121b04283b5",  
  "LastUpdatedTimestamp": "2019-06-25T18:23:22.106Z",  
  "Name": "cliGroup_Core"  
}
```

Ejemplo 2: Para crear una definición básica con una versión inicial

El siguiente `create-core-definition` ejemplo crea una definición básica que contiene una versión inicial de la definición básica. La versión solo puede contener un núcleo. Antes de poder

crear un núcleo, primero debe crear y aprovisionar el elemento de AWS IoT correspondiente. Este proceso incluye los siguientes `iot` comandos, que devuelven el `ThingArn` y `CertificateArn` necesario para el `create-core-definition` comando.

Cree el AWS elemento de IoT que corresponda al dispositivo principal:

```
aws iot create-thing \
  --thing-name "MyCoreDevice"
```

Salida:

```
{
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyCoreDevice",
  "thingName": "MyCoreDevice",
  "thingId": "cb419a19-9099-4515-9cec-e9b0e760608a"
}
```

Cree las claves públicas y privadas y el certificado del dispositivo principal para el dispositivo. En este ejemplo, se usa el `create-keys-and-certificate` comando y se requieren permisos de escritura en el directorio actual. También puede utilizar el comando `create-certificate-from-csr`.

```
aws iot create-keys-and-certificate \
  --set-as-active \
  --certificate-pem-outfile "myCore.cert.pem" \
  --public-key-outfile "myCore.public.key" \
  --private-key-outfile "myCore.private.key"
```

Salida:

```
{
  "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz",
  "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCakGgAwIBATgIUCgq6EGqou6zFqWgIZRndgQEFW+gwDQYJKoZIhvc...KdGewQS\n-----END
CERTIFICATE-----\n",
  "keyPair": {
    "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBzrqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAqKpRgnn6yq26U3y...wIDAQAB\n-----END
PUBLIC KEY-----\n",
```

```

    "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIIEowIABAKCAQEAqKpRgnn6yq26U3yt5YFZquyukfRjBMXDcNOK4rMCxDR...fvY4+te\n-----END
RSA PRIVATE KEY-----\n"
  },
  "certificateId":
    "123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
}

```

Cree una política de AWS IoT que permita `iot` y `greengrass` actúe. Para simplificar, la siguiente política permite realizar acciones en todos los recursos, pero su política debería ser más restrictiva.

```

aws iot create-policy \
  --policy-name "Core_Devices" \
  --policy-document "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\n:\n\"Allow\", \"Action\":[\"iot:Publish\", \"iot:Subscribe\", \"iot:Connect
\n\", \"iot:Receive\"], \"Resource\":[\"*\"]}, {\"Effect\":"Allow\", \"Action\":
[\"iot:GetThingShadow\", \"iot:UpdateThingShadow\", \"iot>DeleteThingShadow\"],
\n\"Resource\":[\"*\"]}, {\"Effect\":"Allow\", \"Action\":[\"greengrass:*\"], \"Resource
\n\":[\"*\"]}]}"

```

Salida:

```

{
  "policyName": "Core_Devices",
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/Core_Devices",
  "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\n:\n\"Allow\", \"Action\":[\"iot:Publish\", \"iot:Subscribe\", \"iot:Connect
\n\", \"iot:Receive\"], \"Resource\":[\"*\"]}, {\"Effect\":"Allow\", \"Action\":
[\"iot:GetThingShadow\", \"iot:UpdateThingShadow\", \"iot>DeleteThingShadow\"],
\n\"Resource\":[\"*\"]}, {\"Effect\":"Allow\", \"Action\":[\"greengrass:*\"], \"Resource
\n\":[\"*\"]}]}",
  "policyVersionId": "1"
}

```

Adjunte la política al certificado:

```

aws iot attach-policy \
  --policy-name "Core_Devices" \
  --target "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"

```

Este comando no genera ninguna salida.

Adjunte lo siguiente al certificado:

```
aws iot attach-thing-principal \
  --thing-name "MyCoreDevice" \
  --principal "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
```

Este comando no genera ninguna salida.

Cree la definición básica:

```
aws greengrass create-core-definition \
  --name "MyCores" \
  --initial-version "{ \"Cores\": [{ \"Id\": \"MyCoreDevice\", \"ThingArn\":
\"arn:aws:iot:us-west-2:123456789012:thing/MyCoreDevice\", \"CertificateArn\":
\"arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz
\", \"SyncShadow\": true} ] }"
```

Salida:

```
{
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/582efe12-b05a-409e-9a24-a2ba1bcc4a12/versions/
cc87b5b3-8f4b-465d-944c-1d6de5dbfcdb",
  "Name": "MyCores",
  "LastUpdatedTimestamp": "2019-09-18T00:11:06.197Z",
  "LatestVersion": "cc87b5b3-8f4b-465d-944c-1d6de5dbfcdb",
  "CreationTimestamp": "2019-09-18T00:11:06.197Z",
  "Id": "582efe12-b05a-409e-9a24-a2ba1bcc4a12",
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
cores/582efe12-b05a-409e-9a24-a2ba1bcc4a12"
}
```

Para obtener más información, consulte [Configurar el núcleo de AWS IoT Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [CreateCoreDefinition](#) de AWS CLI comandos.

## create-deployment

En el siguiente ejemplo de código, se muestra cómo usar `create-deployment`.

### AWS CLI

Para crear una implementación para una versión de un grupo de Greengrass

El siguiente `create-deployment` ejemplo implementa la versión especificada de un grupo de Greengrass.

```
aws greengrass create-deployment \  
  --deployment-type NewDeployment \  
  --group-id "ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca" \  
  --group-version-id "dc40c1e9-e8c8-4d28-a84d-a9cad5f599c9"
```

Salida:

```
{  
  "DeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
groups/ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca/deployments/bfceb608-4e97-45bc-  
af5c-460144270308",  
  "DeploymentId": "bfceb608-4e97-45bc-af5c-460144270308"  
}
```

Para obtener más información, consulte [Introducción a los conectores \(CLI\)](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte [CreateDeployment](#) la Referencia de AWS CLI comandos.

## create-device-definition-version

En el siguiente ejemplo de código, se muestra cómo usar `create-device-definition-version`.

### AWS CLI

Para crear una versión de definición de dispositivo

El siguiente `create-device-definition-version` ejemplo crea una versión de definición de dispositivo y la asocia a la definición de dispositivo especificada. La versión define dos

dispositivos. Antes de poder crear un dispositivo Greengrass, primero debe crear y aprovisionar el elemento de AWS IoT correspondiente. Este proceso incluye los siguientes `iot` comandos que debe ejecutar para obtener la información necesaria para el comando Greengrass:

Cree la cosa de AWS IoT que corresponda al dispositivo:

```
aws iot create-thing \
  --thing-name "InteriorTherm"
```

Salida:

```
{
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/InteriorTherm",
  "thingName": "InteriorTherm",
  "thingId": "01d4763c-78a6-46c6-92be-7add080394bf"
}
```

Cree las claves públicas y privadas y el certificado del dispositivo para el dispositivo. En este ejemplo se usa el `create-keys-and-certificate` comando y se requieren permisos de escritura en el directorio actual. Como alternativa, puede usar el `create-certificate-from-csr` comando:

```
aws iot create-keys-and-certificate \
  --set-as-active \
  --certificate-pem-outfile "myDevice.cert.pem" \
  --public-key-outfile "myDevice.public.key" \
  --private-key-outfile "myDevice.private.key"
```

Salida:

```
{
  "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92",
  "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCAkGgAwIBATgIUCgq6EGqou6zFqWgIZRndgQEFW+gwDQYJKoZIhvc...KdGewQS\n-----END
CERTIFICATE-----\n",
  "keyPair": {
    "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBzrqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAqKpRgnn6yq26U3y...wIDAQAB\n-----END
PUBLIC KEY-----\n",
```



```

    "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIIEowIABAKCAQEAqKpRgnn6yq26U3yt5YFZquyukfRjBMXDcNOK4rMCxDR...fvY4+te\n-----END
RSA PRIVATE KEY-----\n"
  },
  "certificateId":
    "66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
}

```

Cree una política de AWS IoT que permita `iot` y `greengrass` actuar. Para simplificar, la siguiente política permite realizar acciones en todos los recursos, pero su política puede ser más restrictiva:

```

aws iot create-policy \
  --policy-name "GG_Devices" \
  --policy-document "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":"
  "\":\"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect
  \",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":"
  "\":[\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot>DeleteThingShadow\"],
  \"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource
  \":[\"*\"]}]}"

```

Salida:

```

{
  "policyName": "GG_Devices",
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/GG_Devices",
  "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":"
  "\":\"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect
  \",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":"
  "\":[\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot>DeleteThingShadow\"],
  \"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource
  \":[\"*\"]}]}",
  "policyVersionId": "1"
}

```

Adjunte la política al certificado:

```

aws iot attach-policy \
  --policy-name "GG_Devices" \
  --target "arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"

```

## Adjunte la cosa al certificado

```
aws iot attach-thing-principal \
  --thing-name "InteriorTherm" \
  --principal "arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
```

Después de crear y configurar el IoT como se muestra arriba, usa los dos primeros comandos ThingArn y CertificateArn desde el siguiente ejemplo.

```
aws greengrass create-device-definition-version \
  --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd" \
  --devices "[{\\"Id\\":\\"InteriorTherm\\",\\"ThingArn\\":\\"arn:aws:iot:us-
west-2:123456789012:thing/InteriorTherm\\",\\"CertificateArn\\":\\"arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92\\",
\\"SyncShadow\\":true},{\\"Id\\":\\"ExteriorTherm\\",\\"ThingArn\\":\\"arn:aws:iot:us-
west-2:123456789012:thing/ExteriorTherm\\",\\"CertificateArn\\":\\"arn:aws:iot:us-
west-2:123456789012:cert/6c52ce1b47bde88a637e9ccdd45fe4e4c2c0a75a6866f8f63d980ee22fa51e02\\",
\\"SyncShadow\\":true}]"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/83c13984-6fed-447e-84d5-5b8aa45d5f71",
  "Version": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
  "CreationTimestamp": "2019-09-11T00:15:09.838Z",
  "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd"
}
```

- Para API obtener más información, consulte [CreateDeviceDefinitionVersion](#) la Referencia de AWS CLI comandos.

## create-device-definition

En el siguiente ejemplo de código, se muestra cómo usar create-device-definition.

### AWS CLI

Para crear una definición de dispositivo

El siguiente `create-device-definition` ejemplo crea una definición de dispositivo que contiene una versión inicial de la definición de dispositivo. La versión inicial define dos dispositivos. Antes de poder crear un dispositivo Greengrass, primero debe crear y aprovisionar el elemento de AWS IoT correspondiente. Este proceso incluye los siguientes `iot` comandos que debe ejecutar para obtener la información necesaria para el comando Greengrass:

Cree la cosa de AWS IoT que corresponda al dispositivo:

```
aws iot create-thing \  
  --thing-name "InteriorTherm"
```

Salida:

```
{  
  "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/InteriorTherm",  
  "thingName": "InteriorTherm",  
  "thingId": "01d4763c-78a6-46c6-92be-7add080394bf"  
}
```

Cree las claves públicas y privadas y el certificado del dispositivo para el dispositivo. En este ejemplo se usa el `create-keys-and-certificate` comando y se requieren permisos de escritura en el directorio actual. Como alternativa, puede usar el `create-certificate-from-csr` comando:

```
aws iot create-keys-and-certificate \  
  --set-as-active \  
  --certificate-pem-outfile "myDevice.cert.pem" \  
  --public-key-outfile "myDevice.public.key" \  
  --private-key-outfile "myDevice.private.key"
```

Salida:

```
{  
  "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92",  
  "certificatePem": "-----BEGIN CERTIFICATE-----  
  \nMIIDWTCAkGgAwIBATgIUCGq6EGqou6zFqWgIZRndgQEFW+gwDQYJKoZIhvc...KdGewQS\n-----END  
  CERTIFICATE-----\n",  
  "keyPair": {
```

```

    "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBzrqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEAqKpRgnn6yq26U3y...wIDAQAB\n-----END
PUBLIC KEY-----\n",
    "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIABAKCAQEAqKpRgnn6yq26U3yt5YFZquyukfRjBMXDcNOK4rMCxDR...fvY4+te\n-----END
RSA PRIVATE KEY-----\n"
  },
  "certificateId":
  "66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
}

```

Cree una política de AWS IoT que permita `iot` y `greengrass` actúe. Para simplificar, la siguiente política permite realizar acciones en todos los recursos, pero su política puede ser más restrictiva:

```

aws iot create-policy \
  --policy-name "GG_Devices" \
  --policy-document "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\n:\n\"Allow\n\", \"Action\":[\"iot:Publish\n\", \"iot:Subscribe\n\", \"iot:Connect
\n\", \"iot:Receive\n\"], \"Resource\":[\"*\n\"]}, {\"Effect\":[\"Allow\n\", \"Action\":[
\n\"iot:GetThingShadow\n\", \"iot:UpdateThingShadow\n\", \"iot>DeleteThingShadow\n\"],
\n\"Resource\":[\"*\n\"]}, {\"Effect\":[\"Allow\n\", \"Action\":[\"greengrass:*\"], \"Resource
\n\":[\"*\n\"]}]}\"

```

Salida:

```

{
  "policyName": "GG_Devices",
  "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/GG_Devices",
  "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\n:\n\"Allow\n\", \"Action\":[\"iot:Publish\n\", \"iot:Subscribe\n\", \"iot:Connect
\n\", \"iot:Receive\n\"], \"Resource\":[\"*\n\"]}, {\"Effect\":[\"Allow\n\", \"Action\":[
\n\"iot:GetThingShadow\n\", \"iot:UpdateThingShadow\n\", \"iot>DeleteThingShadow\n\"],
\n\"Resource\":[\"*\n\"]}, {\"Effect\":[\"Allow\n\", \"Action\":[\"greengrass:*\"], \"Resource
\n\":[\"*\n\"]}]}\",
  "policyVersionId": "1"
}

```

Adjunte la política al certificado:

```

aws iot attach-policy \
  --policy-name "GG_Devices" \

```

```
--target "arn:aws:iot:us-west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
```

### Adjunte la cosa al certificado

```
aws iot attach-thing-principal \
  --thing-name "InteriorTherm" \
  --principal "arn:aws:iot:us-west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
```

Después de crear y configurar el IoT como se muestra arriba, usa los dos primeros comandos ThingArn y CertificateArn desde el siguiente ejemplo.

```
aws greengrass create-device-definition \
  --name "Sensors" \
  --initial-version "{ \"Devices\": [ { \"Id\": \"InteriorTherm\", \"ThingArn\": \"arn:aws:iot:us-west-2:123456789012:thing/InteriorTherm\", \"CertificateArn\": \"arn:aws:iot:us-west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92\", \"SyncShadow\": true }, { \"Id\": \"ExteriorTherm\", \"ThingArn\": \"arn:aws:iot:us-west-2:123456789012:thing/ExteriorTherm\", \"CertificateArn\": \"arn:aws:iot:us-west-2:123456789012:cert/6c52ce1b47bde88a637e9ccdd45fe4e4c2c0a75a6866f8f63d980ee22fa51e02\", \"SyncShadow\": true } ] }"
```

### Salida:

```
{
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/versions/3b5cc510-58c1-44b5-9d98-4ad858ffa795",
  "Name": "Sensors",
  "LastUpdatedTimestamp": "2019-09-11T00:11:06.197Z",
  "LatestVersion": "3b5cc510-58c1-44b5-9d98-4ad858ffa795",
  "CreationTimestamp": "2019-09-11T00:11:06.197Z",
  "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd"
}
```

- Para API obtener más información, consulte [CreateDeviceDefinition](#) la Referencia de AWS CLI comandos.

## create-function-definition-version

En el siguiente ejemplo de código, se muestra cómo usar `create-function-definition-version`.

### AWS CLI

Para crear una versión de la definición de la función

El siguiente `create-function-definition-version` ejemplo crea una nueva versión de la definición de función especificada. Esta versión especifica una sola función cuyo identificador es `Hello-World-function`, permite el acceso al sistema de archivos y especifica un tamaño máximo de memoria y un período de tiempo de espera.

```
aws greengrass create-function-definition-version \
  --cli-input-json "{\"FunctionDefinitionId\": \"e626e8c9-3b8f-4bf3-9cdc-
d26ecdeb9fa3\", \"Functions\": [{\"Id\": \"Hello-World-function\", \"FunctionArn\":
  \"arn:aws:lambda:us-
west-2:123456789012:function:Greengrass_HelloWorld_Counter:gghw-alias\"},
  {\"FunctionConfiguration\": {\"Environment\": {\"AccessSysfs\": true}, \"Executable\":
  \"greengrassHelloWorldCounter.function_handler\", \"MemorySize\": 16000, \"Pinned\":
  false, \"Timeout\": 25}}}]}"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/e626e8c9-3b8f-4bf3-9cdc-d26ecdeb9fa3/
versions/74abd1cc-637e-4abe-8684-9a67890f4043",
  "CreationTimestamp": "2019-06-25T22:03:43.376Z",
  "Id": "e626e8c9-3b8f-4bf3-9cdc-d26ecdeb9fa3",
  "Version": "74abd1cc-637e-4abe-8684-9a67890f4043"
}
```

- Para API obtener más información, consulte [CreateFunctionDefinitionVersion](#) la Referencia de AWS CLI comandos.

## create-function-definition

En el siguiente ejemplo de código, se muestra cómo usar `create-function-definition`.

## AWS CLI

Para crear una definición de función Lambda

El siguiente `create-function-definition` ejemplo crea una definición de función Lambda y una versión inicial proporcionando una lista de funciones Lambda (en este caso, una lista con solo una función nombrada `TempMonitorFunction`) y sus configuraciones. Para poder crear la definición de la función, necesita la función Lambda. ARN Para crear la función y su alias, utilice los comandos `create-function` y `publish-version` Lambda. El `create-function` comando de Lambda requiere la función ARN de ejecución, aunque AWS IoT Greengrass no utilice esa función porque los permisos se especifican en la función de grupo Greengrass. Puede usar el IAM `create-role` comando para crear un rol vacío y usarlo con el de Lambda, `create-function` o puede usar un rol de ejecución existente. ARN

```
aws greengrass create-function-definition \  
  --name MyGreengrassFunctions \  
  --initial-version "{\"Functions\": [{\"Id\": \"TempMonitorFunction\",  
  \"FunctionArn\": \"arn:aws:lambda:us-  
west-2:123456789012:function:TempMonitor:GG_TempMonitor\", \"FunctionConfiguration  
\": {\"Executable\": \"temp_monitor.function_handler\", \"MemorySize\": 16000,  
\"Timeout\": 5}}]}\"
```

Salida:

```
{  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/  
functions/3b0d0080-87e7-48c6-b182-503ec743a08b",  
  "CreationTimestamp": "2019-06-19T22:24:44.585Z",  
  "Id": "3b0d0080-87e7-48c6-b182-503ec743a08b",  
  "LastUpdatedTimestamp": "2019-06-19T22:24:44.585Z",  
  "LatestVersion": "67f918b9-efb4-40b0-b87c-de8c9faf085b",  
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
definition/functions/3b0d0080-87e7-48c6-b182-503ec743a08b/versions/67f918b9-  
efb4-40b0-b87c-de8c9faf085b",  
  "Name": "MyGreengrassFunctions"  
}
```

Para obtener más información, consulte [Cómo configurar el acceso a los recursos locales mediante la interfaz de línea de AWS comandos](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte [CreateFunctionDefinition](#) la Referencia de AWS CLI comandos.

## create-group-certificate-authority

En el siguiente ejemplo de código, se muestra cómo usar `create-group-certificate-authority`.

### AWS CLI

Para crear una autoridad de certificación (CA) para un grupo

En el siguiente `create-group-certificate-authority` ejemplo, se crea o gira una CA para el grupo especificado.

```
aws greengrass create-group-certificate-authority \  
  --group-id "8eaadd72-ce4b-4f15-892a-0cc4f3a343f1"
```

Salida:

```
{  
  "GroupCertificateAuthorityArn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/groups/8eaadd72-ce4b-4f15-892a-0cc4f3a343f1/certificateauthorities/  
d31630d674c4437f6c5dbc0dca56312a902171ce2d086c38e509c8EXAMPLEecc5"  
}
```

Para obtener más información, consulte [Seguridad de AWS IoT Greengrass en la Guía](#) para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [CreateGroupCertificateAuthority](#) de AWS CLI comandos.

## create-group-version

En el siguiente ejemplo de código, se muestra cómo usar `create-group-version`.

### AWS CLI

Para crear una versión de un grupo de Greengrass



El siguiente `create-group-version` ejemplo crea una versión de grupo y la asocia al grupo especificado. La versión hace referencia a las versiones principal, de recursos, de conector, función y suscripción que contienen las entidades que se van a incluir en esta versión de grupo. Debe crear estas entidades antes de poder crear la versión de grupo.

Para crear una definición de recurso con una versión inicial, utilice el `create-resource-definition` comando. Para crear una definición de conector con una versión inicial, utilice el `create-connector-definition` comando. Para crear una definición de función con una versión inicial, utilice el `create-function-definition` comando. Para crear una definición de suscripción con una versión inicial, utilice el `create-subscription-definition` comando. Para recuperar la última versión ARN de la definición básica, utilice el `get-group-version` comando y especifique el ID de la última versión del grupo.

```
aws greengrass create-group-version \  
  --group-id "ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca" \  
  --core-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/6a630442-8708-4838-ad36-eb98849d975e/versions/6c87151b-1fb4-4cb2-8b31-6ee715d8f8ba" \  
  --resource-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38/versions/a5f94d0b-f6bc-40f4-bb78-7a1c5fe13ba1" \  
  --connector-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/connectors/55d0052b-0d7d-44d6-b56f-21867215e118/versions/78a3331b-895d-489b-8823-17b4f9f418a0" \  
  --function-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/functions/3b0d0080-87e7-48c6-b182-503ec743a08b/versions/67f918b9-efb4-40b0-b87c-de8c9faf085b" \  
  --subscription-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/subscriptions/9d611d57-5d5d-44bd-a3b4-fecbbdd69112/versions/aa645c47-ac90-420d-9091-8c7ffa4f103f"
```

Salida:

```
{  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca/versions/e10b0459-4345-4a09-88a4-1af1f5d34638",  
  "CreationTimestamp": "2019-06-20T18:42:47.020Z",  
  "Id": "ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca",  
  "Version": "e10b0459-4345-4a09-88a4-1af1f5d34638"  
}
```

Para obtener más información, consulte [Descripción general del modelo de objetos del grupo Greengrass de AWS IoT](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [CreateGroupVersion](#) de AWS CLI comandos.

## create-group

En el siguiente ejemplo de código, se muestra cómo usar `create-group`.

### AWS CLI

Para crear un grupo de Greengrass

En el siguiente `create-group` ejemplo se crea un grupo denominado. `cli-created-group`

```
aws greengrass create-group \  
  --name cli-created-group
```

Salida:

```
{  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
groups/4e22bd92-898c-436b-ade5-434d883ff749",  
  "CreationTimestamp": "2019-06-25T18:07:17.688Z",  
  "Id": "4e22bd92-898c-436b-ade5-434d883ff749",  
  "LastUpdatedTimestamp": "2019-06-25T18:07:17.688Z",  
  "Name": "cli-created-group"  
}
```

Para obtener más información, consulte [Descripción general del modelo de objetos del grupo Greengrass de AWS IoT](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [CreateGroup](#) de AWS CLI comandos.

## create-logger-definition-version

En el siguiente ejemplo de código, se muestra cómo usar `create-logger-definition-version`.

## AWS CLI

Para crear una versión de definición de registrador

El siguiente `create-logger-definition-version` ejemplo crea una versión de definición de registrador y la asocia a una definición de registrador. La versión define cuatro configuraciones de registro: 1) registros de componentes del sistema en el sistema de archivos del dispositivo principal, 2) registros de funciones Lambda definidas por el usuario en el sistema de archivos del dispositivo principal, 3) registros de componentes del sistema en CloudWatch Amazon Logs y 4) registros de funciones Lambda definidas por el usuario en Amazon Logs. CloudWatch Nota: Para la integración de CloudWatch los registros, su rol de grupo debe conceder los permisos adecuados.

```
aws greengrass create-logger-definition-version \
  --logger-definition-id "a454b62a-5d56-4ca9-bdc4-8254e1662cb0" \
  --loggers "[{"Id":"1","Component":"GreengrassSystem","Level":"ERROR",
  "Space":10240,"Type":"FileSystem"}, {"Id":"2","Component":"Lambda",
  "Level":"INFO","Space":10240,"Type":"FileSystem"}, {"Id":"3",
  "Component":"GreengrassSystem","Level":"WARN","Type":"AWSCloudWatch"},
  {"Id":"4","Component":"Lambda","Level":"INFO","Type":"AWSCloudWatch"}]"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/loggers/a454b62a-5d56-4ca9-bdc4-8254e1662cb0/versions/49aedb1e-01a3-4d39-9871-3a052573f1ea",
  "Version": "49aedb1e-01a3-4d39-9871-3a052573f1ea",
  "CreationTimestamp": "2019-07-24T00:04:48.523Z",
  "Id": "a454b62a-5d56-4ca9-bdc4-8254e1662cb0"
}
```

Para obtener más información, consulte [Supervisión con registros de AWS IoT Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [CreateLoggerDefinitionVersion](#) de AWS CLI comandos.

### **create-logger-definition**

En el siguiente ejemplo de código, se muestra cómo usar `create-logger-definition`.

## AWS CLI

Para crear una definición de registrador

El siguiente `create-logger-definition` ejemplo crea una definición de registrador que contiene una versión inicial de la definición de registrador. La versión inicial define tres configuraciones de registro: 1) registros de componentes del sistema en el sistema de archivos del dispositivo principal, 2) registros de funciones Lambda definidas por el usuario en el sistema de archivos del dispositivo principal y 3) registros de funciones Lambda definidas por el usuario en Amazon Logs. CloudWatch Nota: Para la integración de CloudWatch Logs, su rol de grupo debe conceder los permisos adecuados.

```
aws greengrass create-logger-definition \
  --name "LoggingConfigs" \
  --initial-version "{\"Loggers\": [{\"Id\": \"1\", \"Component\": \"GreengrassSystem\", \"Level\": \"ERROR\", \"Space\": 10240, \"Type\": \"FileSystem\"}, {\"Id\": \"2\", \"Component\": \"Lambda\", \"Level\": \"INFO\", \"Space\": 10240, \"Type\": \"FileSystem\"}, {\"Id\": \"3\", \"Component\": \"Lambda\", \"Level\": \"INFO\", \"Type\": \"AWSCloudWatch\"}]}"
```

Salida:

```
{
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/loggers/a454b62a-5d56-4ca9-bdc4-8254e1662cb0/versions/de1d9854-1588-4525-b25e-b378f60f2322",
  "Name": "LoggingConfigs",
  "LastUpdatedTimestamp": "2019-07-23T23:52:17.165Z",
  "LatestVersion": "de1d9854-1588-4525-b25e-b378f60f2322",
  "CreationTimestamp": "2019-07-23T23:52:17.165Z",
  "Id": "a454b62a-5d56-4ca9-bdc4-8254e1662cb0",
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/loggers/a454b62a-5d56-4ca9-bdc4-8254e1662cb0"
}
```

Para obtener más información, consulte [Supervisión con registros de AWS IoT Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [CreateLoggerDefinition](#) de AWS CLI comandos.

## create-resource-definition-version

En el siguiente ejemplo de código, se muestra cómo usar `create-resource-definition-version`.

### AWS CLI

Para crear una versión de una definición de recurso

En el siguiente `create-resource-definition-version` ejemplo, se crea una nueva versión de un `TwilioAuthToken`.

```
aws greengrass create-resource-definition-version \
  --resource-definition-id "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38" \
  --resources "[{"Id": "TwilioAuthToken"}, {"Name": "MyTwilioAuthToken"}, {"ResourceDataContainer": {"SecretsManagerSecretResourceData": {"ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:greenrass-TwilioAuthToken-ntS1p6"}}}]"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38/versions/b3bcada0-5fb6-42df-bf0b-1ee4f15e769e",
  "CreationTimestamp": "2019-06-24T21:17:25.623Z",
  "Id": "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
  "Version": "b3bcada0-5fb6-42df-bf0b-1ee4f15e769e"
}
```

- Para API obtener más información, consulte [CreateResourceDefinitionVersion](#) la Referencia de AWS CLI comandos.

## create-resource-definition

En el siguiente ejemplo de código, se muestra cómo usar `create-resource-definition`.

### AWS CLI

Para crear una definición de recurso

El siguiente `create-resource-definition` ejemplo crea una definición de recurso que contiene una lista de recursos que se utilizarán en un grupo de Greengrass. En este ejemplo, se incluye una versión inicial de la definición de recurso proporcionando una lista de recursos. La lista incluye un recurso para un token de autorización de Twilio y otro ARN para un secreto almacenado en AWS Secrets Manager. Debes crear el secreto antes de poder crear la definición del recurso.

```
aws greengrass create-resource-definition \
  --name MyGreengrassResources \
  --initial-version "{\"Resources\": [{\"Id\": \"TwilioAuthToken
  \",\"Name\": \"MyTwilioAuthToken\", \"ResourceDataContainer\":
  {\"SecretsManagerSecretResourceData\": {\"ARN\": \"arn:aws:secretsmanager:us-
  west-2:123456789012:secret:greengrass-TwilioAuthToken-ntSlp6\"}}}]}"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
  resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
  "CreationTimestamp": "2019-06-19T21:51:28.212Z",
  "Id": "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
  "LastUpdatedTimestamp": "2019-06-19T21:51:28.212Z",
  "LatestVersion": "a5f94d0b-f6bc-40f4-bb78-7a1c5fe13ba1",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
  definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38/versions/a5f94d0b-
  f6bc-40f4-bb78-7a1c5fe13ba1",
  "Name": "MyGreengrassResources"
}
```

Para obtener más información, consulte [Cómo configurar el acceso a los recursos locales mediante la interfaz de línea de AWS comandos](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte [CreateResourceDefinition](#) la Referencia de AWS CLI comandos.

## **create-software-update-job**

En el siguiente ejemplo de código, se muestra cómo usar `create-software-update-job`.

## AWS CLI

Para crear un trabajo de actualización de software para un núcleo

El siguiente `create-software-update-job` ejemplo crea un trabajo de actualización over-the-air (OTA) para actualizar el software AWS IoT Greengrass Core en el núcleo cuyo nombre es `MyFirstGroup_Core`. Este comando requiere un IAM rol que permita el acceso a los paquetes de actualización de software en Amazon S3 y `iot.amazonaws.com` se incluye como entidad de confianza.

```
aws greengrass create-software-update-job \
  --update-targets-architecture armv7l \
  --update-targets ["arn:aws:iot:us-west-2:123456789012:thing/MyFirstGroup_Core"] \
  --update-targets-operating-system raspbian \
  --software-to-update core \
  --s3-url-signer-role arn:aws:iam::123456789012:role/OTA_signer_role \
  --update-agent-log-level WARN
```

Salida:

```
{
  "IotJobId": "GreengrassUpdateJob_30b353e3-3af7-4786-be25-4c446663c09e",
  "IotJobArn": "arn:aws:iot:us-west-2:123456789012:job/GreengrassUpdateJob_30b353e3-3af7-4786-be25-4c446663c09e",
  "PlatformSoftwareVersion": "1.9.3"
}
```

Para obtener más información, consulte [OTAActualizaciones del software principal de AWS IoT Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [CreateSoftwareUpdateJob](#) de AWS CLI comandos.

## **create-subscription-definition-version**

En el siguiente ejemplo de código, se muestra cómo usar `create-subscription-definition-version`.

## AWS CLI

Para crear una nueva versión de una definición de suscripción

El siguiente `create-subscription-definition-version` ejemplo crea una nueva versión de una definición de suscripción que contiene tres suscripciones: una notificación de activación, una entrada de temperatura y un estado de salida.

```
aws greengrass create-subscription-definition-version \
  --subscription-definition-id "9d611d57-5d5d-44bd-a3b4-feccbdd69112" \
  --subscriptions "[{\\"Id\\": \\"TriggerNotification\\", \\"Source\\":
  \\"arn:aws:lambda:us-west-2:123456789012:function:TempMonitor:GG_TempMonitor
  \", \\"Subject\\": \\"twilio/txt\\", \\"Target\\": \\"arn:aws:greengrass:us-west-2:/:
  connectors/TwilioNotifications/versions/1\\"},{\\"Id\\": \\"TemperatureInput\\", \\"Source
  \": \\"cloud\\", \\"Subject\\": \\"temperature/input\\", \\"Target\\": \\"arn:aws:lambda:us-
  west-2:123456789012:function:TempMonitor:GG_TempMonitor\\"},{\\"Id\\": \\"OutputStatus
  \", \\"Source\\": \\"arn:aws:greengrass:us-west-2:/:connectors/TwilioNotifications/
  versions/1\\", \\"Subject\\": \\"twilio/message/status\\", \\"Target\\": \\"cloud\\"}]"]
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
  subscriptions/9d611d57-5d5d-44bd-a3b4-feccbdd69112/versions/7b65dfae-50b6-4d0f-
  b3e0-27728bfb0620",
  "CreationTimestamp": "2019-06-24T21:21:33.837Z",
  "Id": "9d611d57-5d5d-44bd-a3b4-feccbdd69112",
  "Version": "7b65dfae-50b6-4d0f-b3e0-27728bfb0620"
}
```

- Para API obtener más información, consulte [CreateSubscriptionDefinitionVersion](#) la Referencia de AWS CLI comandos.

## **create-subscription-definition**

En el siguiente ejemplo de código, se muestra cómo usar `create-subscription-definition`.

### AWS CLI

Para crear una definición de suscripción

En el siguiente `create-subscription-definition` ejemplo, se crea una definición de suscripción y se especifica su versión inicial. La versión inicial contiene tres suscripciones: una para el MQTT tema al que se suscribe el conector, otra para permitir que una función reciba lecturas de temperatura del AWS IoT y otra para permitir que el AWS IoT reciba información



de estado del conector. El ejemplo proporciona el ARN alias de la función Lambda que se creó anteriormente mediante el comando de Lambda. `create-alias`

```
aws greengrass create-subscription-definition \
  --initial-version "{\"Subscriptions\": [{\"Id\":
  \"TriggerNotification\", \"Source\": \"arn:aws:lambda:us-
  west-2:123456789012:function:TempMonitor:GG_TempMonitor\", \"Subject\":
  \"twilio/txt\", \"Target\": \"arn:aws:greengrass:us-west-2:/connectors/
  TwilioNotifications/versions/1\"},{\"Id\": \"TemperatureInput\", \"Source\":
  \"cloud\", \"Subject\": \"temperature/input\", \"Target\": \"arn:aws:lambda:us-
  west-2:123456789012:function:TempMonitor:GG_TempMonitor\"},{\"Id\": \"OutputStatus
  \", \"Source\": \"arn:aws:greengrass:us-west-2:/connectors/TwilioNotifications/
  versions/1\", \"Subject\": \"twilio/message/status\", \"Target\": \"cloud\"}]}"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
  subscriptions/9d611d57-5d5d-44bd-a3b4-feccbdd69112",
  "CreationTimestamp": "2019-06-19T22:34:26.677Z",
  "Id": "9d611d57-5d5d-44bd-a3b4-feccbdd69112",
  "LastUpdatedTimestamp": "2019-06-19T22:34:26.677Z",
  "LatestVersion": "aa645c47-ac90-420d-9091-8c7ffa4f103f",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
  definition/subscriptions/9d611d57-5d5d-44bd-a3b4-feccbdd69112/versions/aa645c47-
  ac90-420d-9091-8c7ffa4f103f"
}
```

Para obtener más información, consulte [Introducción a los conectores \(CLI\)](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte [CreateSubscriptionDefinition](#) la Referencia de AWS CLI comandos.

## **delete-connector-definition**

En el siguiente ejemplo de código, se muestra cómo usar `delete-connector-definition`.

### AWS CLI

Para eliminar una definición de conector

El siguiente `delete-connector-definition` ejemplo elimina la definición de conector de Greengrass especificada. Si elimina una definición de conector utilizada por un grupo, ese grupo no se podrá implementar correctamente.

```
aws greengrass delete-connector-definition \  
  --connector-definition-id "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteConnectorDefinition](#) la Referencia de AWS CLI comandos.

## **delete-core-definition**

En el siguiente ejemplo de código, se muestra cómo usar `delete-core-definition`.

### AWS CLI

Para eliminar una definición básica

En el siguiente `delete-core-definition` ejemplo, se elimina la definición básica de Greengrass especificada, incluidas todas las versiones. Si eliminas un núcleo que está asociado a un grupo de Greengrass, ese grupo no se podrá implementar correctamente.

```
aws greengrass delete-core-definition \  
  --core-definition-id "ff36cc5f-9f98-4994-b468-9d9b6dc52abd"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteCoreDefinition](#) la Referencia de AWS CLI comandos.

## **delete-device-definition**

En el siguiente ejemplo de código, se muestra cómo usar `delete-device-definition`.

### AWS CLI

Para eliminar una definición de dispositivo

En el siguiente `delete-device-definition` ejemplo, se elimina la definición de dispositivo especificada, incluidas todas sus versiones. Si elimina una versión de definición de dispositivo utilizada por una versión de grupo, la versión de grupo no se podrá implementar correctamente.

```
aws greengrass delete-device-definition \  
  --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteDeviceDefinition](#) la Referencia de AWS CLI comandos.

## **delete-function-definition**

En el siguiente ejemplo de código, se muestra cómo usar `delete-function-definition`.

### AWS CLI

Para eliminar una definición de función

El siguiente `delete-function-definition` ejemplo elimina la definición de función de Greengrass especificada. Si elimina una definición de función utilizada por un grupo, ese grupo no se podrá implementar correctamente.

```
aws greengrass delete-function-definition \  
  --function-definition-id "fd4b906a-dff3-4c1b-96eb-52ebfcfac06a"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteFunctionDefinition](#) la Referencia de AWS CLI comandos.

## **delete-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-group`.

### AWS CLI

Para eliminar un grupo

El siguiente `delete-group` ejemplo elimina el grupo Greengrass especificado.

```
aws greengrass delete-group \  
  --group-id "4e22bd92-898c-436b-ade5-434d883ff749"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte la Referencia [DeleteGroup](#) de AWS CLI comandos.

## delete-logger-definition

En el siguiente ejemplo de código, se muestra cómo usar `delete-logger-definition`.

### AWS CLI

Para eliminar una definición de registrador

El siguiente `delete-logger-definition` ejemplo elimina la definición de registrador especificada, incluidas todas las versiones de definición de registrador. Si elimina una versión de definición de registrador utilizada por una versión de grupo, la versión de grupo no se podrá implementar correctamente.

```
aws greengrass delete-logger-definition \  
  --logger-definition-id "a454b62a-5d56-4ca9-bdc4-8254e1662cb0"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Supervisión con registros de AWS IoT Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [DeleteLoggerDefinition](#) de AWS CLI comandos.

## delete-resource-definition

En el siguiente ejemplo de código, se muestra cómo usar `delete-resource-definition`.

### AWS CLI

Para eliminar una definición de recurso

En el siguiente `delete-resource-definition` ejemplo, se elimina la definición de recurso especificada, incluidas todas las versiones del recurso. Si eliminas una definición de recurso que utiliza un grupo, ese grupo no se podrá implementar correctamente.

```
aws greengrass delete-resource-definition \  
  --resource-definition-id "ad8c101d-8109-4b0e-b97d-9cc5802ab658"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteResourceDefinition](#) la Referencia de AWS CLI comandos.

## delete-subscription-definition

En el siguiente ejemplo de código, se muestra cómo usar `delete-subscription-definition`.

### AWS CLI

Para eliminar una definición de suscripción

El siguiente `delete-subscription-definition` ejemplo elimina la definición de suscripción de Greengrass especificada. Si eliminas una suscripción que está siendo utilizada por un grupo, ese grupo no se podrá implementar correctamente.

```
aws greengrass delete-subscription-definition \  
  --subscription-definition-id "cd6f1c37-d9a4-4e90-be94-01a7404f5967"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteSubscriptionDefinition](#) la Referencia de AWS CLI comandos.

## disassociate-role-from-group

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-role-from-group`.

### AWS CLI

Para desasociar el rol de un grupo de Greengrass

El siguiente `disassociate-role-from-group` ejemplo desasocia el IAM rol del grupo de Greengrass especificado.

```
aws greengrass disassociate-role-from-group \  
  --group-id 2494ee3f-7f8a-4e92-a78b-d205f808b84b
```

Salida:

```
{
  "DisassociatedAt": "2019-09-10T20:05:49Z"
}
```

Para obtener más información, consulte [Configurar el rol de grupo](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte [DisassociateRoleFromGroup](#) la Referencia de AWS CLI comandos.

## **disassociate-service-role-from-account**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-service-role-from-account`.

AWS CLI

Para desasociar un rol de servicio de tu cuenta AWS

En el siguiente `disassociate-service-role-from-account` ejemplo, se elimina el rol de servicio que está asociado a tu AWS cuenta. Si no utiliza la función de servicio en ninguna AWS región, utilice el `delete-role-policy` comando para separar la política `AWSGreengrassResourceAccessRolePolicy` gestionada de la función y, a continuación, utilice el `delete-role` comando para eliminar la función.

```
aws greengrass disassociate-service-role-from-account
```

Salida:

```
{
  "DisassociatedAt": "2019-06-25T22:12:55Z"
}
```

Para obtener más información, consulte el [rol de servicio de Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [DisassociateServiceRoleFromAccount](#) de AWS CLI comandos.

## get-associated-role

En el siguiente ejemplo de código, se muestra cómo usar `get-associated-role`.

### AWS CLI

Para asociar el rol a un grupo de Greengrass

En el siguiente `get-associated-role` ejemplo, se obtiene el IAM rol asociado al grupo de Greengrass especificado. Las funciones y conectores de Lambda locales utilizan el rol de grupo para acceder AWS a los servicios.

```
aws greengrass get-associated-role \  
  --group-id 2494ee3f-7f8a-4e92-a78b-d205f808b84b
```

Salida:

```
{  
  "RoleArn": "arn:aws:iam::123456789012:role/GG-Group-Role",  
  "AssociatedAt": "2019-09-10T20:03:30Z"  
}
```

Para obtener más información, consulte [Configurar el rol de grupo](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte [GetAssociatedRole](#) la Referencia de AWS CLI comandos.

## get-bulk-deployment-status

En el siguiente ejemplo de código, se muestra cómo usar `get-bulk-deployment-status`.

### AWS CLI

Para comprobar el estado de su despliegue masivo

En el siguiente `get-bulk-deployment-status` ejemplo, se recupera la información de estado de la operación de despliegue masivo especificada. En este ejemplo, el archivo que especificó los grupos que se van a implementar tiene un registro de entrada no válido.

```
aws greengrass get-bulk-deployment-status \  
  --bulk-deployment-id "870fb41b-6288-4e0c-bc76-a7ba4b4d3267"
```

Salida:

```
{
  "BulkDeploymentMetrics": {
    "InvalidInputRecords": 1,
    "RecordsProcessed": 1,
    "RetryAttempts": 0
  },
  "BulkDeploymentStatus": "Completed",
  "CreatedAt": "2019-06-25T16:11:33.265Z",
  "tags": {}
}
```

Para obtener más información, consulte [Crear despliegues masivos para grupos](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte la Referencia [GetBulkDeploymentStatus](#) de AWS CLI comandos.

## get-connectivity-info

En el siguiente ejemplo de código, se muestra cómo usar `get-connectivity-info`.

AWS CLI

Para obtener la información de conectividad de un núcleo de Greengrass

El siguiente `get-connectivity-info` ejemplo muestra los puntos finales que los dispositivos pueden usar para conectarse al núcleo de Greengrass especificado. La información de conectividad es una lista de direcciones IP o nombres de dominio, con los números de puerto correspondientes y los metadatos opcionales definidos por el cliente.

```
aws greengrass get-connectivity-info \
  --thing-name "MyGroup_Core"
```

Salida:

```
{
  "ConnectivityInfo": [
    {
      "Metadata": "",

```



```

        "PortNumber": 8883,
        "HostAddress": "127.0.0.1",
        "Id": "AUTOIP_127.0.0.1_0"
    },
    {
        "Metadata": "",
        "PortNumber": 8883,
        "HostAddress": "192.168.1.3",
        "Id": "AUTOIP_192.168.1.3_1"
    },
    {
        "Metadata": "",
        "PortNumber": 8883,
        "HostAddress": "::1",
        "Id": "AUTOIP_::1_2"
    },
    {
        "Metadata": "",
        "PortNumber": 8883,
        "HostAddress": "fe80::1e69:ed93:f5b:f6d",
        "Id": "AUTOIP_fe80::1e69:ed93:f5b:f6d_3"
    }
]
}

```

- Para API obtener más información, consulte la Referencia [GetConnectivityInfo](#) de AWS CLI comandos.

## get-connector-definition-version

En el siguiente ejemplo de código, se muestra cómo usar `get-connector-definition-version`.

### AWS CLI

Para obtener información sobre una versión específica de una definición de conector

El siguiente `get-connector-definition-version` ejemplo recupera información sobre la versión especificada de la definición de conector especificada. Para recuperar todas las versiones de la definición de conector, utilice el `list-connector-definition-versions` comando. IDs Para recuperar el ID de la última versión añadida a la definición del conector, utilice el `get-connector-definition` comando y compruebe la `LatestVersion` propiedad.

```
aws greengrass get-connector-definition-version \  
  --connector-definition-id "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8" \  
  --connector-definition-version-id "63c57963-c7c2-4a26-a7e2-7bf478ea2623"
```

Salida:

```
{  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/  
connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/versions/63c57963-c7c2-4a26-  
a7e2-7bf478ea2623",  
  "CreationTimestamp": "2019-06-19T19:30:01.300Z",  
  "Definition": {  
    "Connectors": [  
      {  
        "ConnectorArn": "arn:aws:greengrass:us-west-2:./connectors/SNS/  
versions/1",  
        "Id": "MySNSConnector",  
        "Parameters": {  
          "DefaultSNSArn": "arn:aws:sns:us-  
west-2:123456789012:GGConnectorTopic"  
        }  
      }  
    ]  
  },  
  "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",  
  "Version": "63c57963-c7c2-4a26-a7e2-7bf478ea2623"  
}
```

Para obtener más información, consulte [Integración con servicios y protocolos mediante conectores Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [GetConnectorDefinitionVersion](#) de AWS CLI comandos.

## get-connector-definition

En el siguiente ejemplo de código, se muestra cómo usar `get-connector-definition`.

### AWS CLI

Para recuperar información sobre la definición de un conector

El siguiente `get-connector-definition` ejemplo recupera información sobre la definición de conector especificada. Para recuperar las definiciones IDs de conectores, utilice el `list-connector-definitions` comando.

```
aws greengrass get-connector-definition \  
  --connector-definition-id "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8"
```

Salida:

```
{  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/  
connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",  
  "CreationTimestamp": "2019-06-19T19:30:01.300Z",  
  "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",  
  "LastUpdatedTimestamp": "2019-06-19T19:30:01.300Z",  
  "LatestVersion": "63c57963-c7c2-4a26-a7e2-7bf478ea2623",  
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/versions/63c57963-  
c7c2-4a26-a7e2-7bf478ea2623",  
  "Name": "MySNSConnector",  
  "tags": {}  
}
```

Para obtener más información, consulte [Integración con servicios y protocolos mediante conectores Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [GetConnectorDefinition](#) de AWS CLI comandos.

## get-core-definition-version

En el siguiente ejemplo de código, se muestra cómo usar `get-core-definition-version`.

AWS CLI

Para recuperar detalles sobre una versión específica de la definición básica de Greengrass

El siguiente `get-core-definition-version` ejemplo recupera información sobre la versión especificada de la definición básica especificada. Para recuperar todas las versiones de la definición básica, utilice el `list-core-definition-versions` comando. IDs Para

recuperar el identificador de la última versión añadida a la definición básica, utilice el `get-core-definition` comando y compruebe la `LatestVersion` propiedad.

```
aws greengrass get-core-definition-version \
  --core-definition-id "c906ed39-a1e3-4822-a981-7b9bd57b4b46" \
  --core-definition-version-id "42aeaac3-fd9d-4312-a8fd-ffa9404a20e0"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/
c906ed39-a1e3-4822-a981-7b9bd57b4b46/versions/42aeaac3-fd9d-4312-a8fd-ffa9404a20e0",
  "CreationTimestamp": "2019-06-18T16:21:21.351Z",
  "Definition": {
    "Cores": [
      {
        "CertificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/928dea7b82331b47c3ff77b0e763fc5e64e2f7c884e6ef391baed9b6b8e21b45",
        "Id": "1a39aac7-0885-4417-91f6-23e4cea6c511",
        "SyncShadow": false,
        "ThingArn": "arn:aws:iot:us-west-2:123456789012:thing/
GGGroup4Pi3_Core"
      }
    ]
  },
  "Id": "c906ed39-a1e3-4822-a981-7b9bd57b4b46",
  "Version": "42aeaac3-fd9d-4312-a8fd-ffa9404a20e0"
}
```

- Para API obtener más información, consulte [GetCoreDefinitionVersion](#) la Referencia de AWS CLI comandos.

## get-core-definition

En el siguiente ejemplo de código, se muestra cómo usar `get-core-definition`.

### AWS CLI

Para recuperar los detalles de una definición básica de Greengrass

El siguiente `get-core-definition` ejemplo recupera información sobre la definición básica especificada. Para recuperar IDs las definiciones principales, utilice el `list-core-definitions` comando.

```
aws greengrass get-core-definition \  
  --core-definition-id "c906ed39-a1e3-4822-a981-7b9bd57b4b46"
```

Salida:

```
{  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/  
cores/237d6916-27cf-457f-ba0c-e86cfb5d25cd",  
  "CreationTimestamp": "2018-10-18T04:47:06.721Z",  
  "Id": "237d6916-27cf-457f-ba0c-e86cfb5d25cd",  
  "LastUpdatedTimestamp": "2018-10-18T04:47:06.721Z",  
  "LatestVersion": "bd2cd6d4-2bc5-468a-8962-39e071e34b68",  
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/definition/cores/237d6916-27cf-457f-ba0c-e86cfb5d25cd/versions/  
bd2cd6d4-2bc5-468a-8962-39e071e34b68",  
  "tags": {}  
}
```

- Para API obtener más información, consulte [GetCoreDefinition](#) la Referencia de AWS CLI comandos.

## get-deployment-status

En el siguiente ejemplo de código, se muestra cómo usar `get-deployment-status`.

### AWS CLI

Para recuperar el estado de un despliegue

El siguiente `get-deployment-status` ejemplo recupera el estado de la implementación especificada del grupo Greengrass especificado. Para obtener el ID de despliegue, utilice el `list-deployments` comando y especifique el ID del grupo.

```
aws greengrass get-deployment-status \  
  --group-id "1013db12-8b58-45ff-acc7-704248f66731" \  
  --deployment-id "1065b8a0-812b-4f21-9d5d-e89b232a530f"
```

Salida:

```
{
  "DeploymentStatus": "Success",
  "DeploymentType": "NewDeployment",
  "UpdatedAt": "2019-06-18T17:04:44.761Z"
}
```

- Para API obtener más información, consulte [GetDeploymentStatus](#) la Referencia de AWS CLI comandos.

## get-device-definition-version

En el siguiente ejemplo de código, se muestra cómo usar `get-device-definition-version`.

AWS CLI

Para obtener una versión de definición de dispositivo

El siguiente `get-device-definition-version` ejemplo recupera información sobre la versión especificada de la definición de dispositivo especificada. Para recuperar todas las versiones de la definición de dispositivo, utilice el `list-device-definition-versions` comando. IDs Para recuperar el ID de la última versión añadida a la definición del dispositivo, utilice el `get-device-definition` comando y compruebe la `LatestVersion` propiedad.

```
aws greengrass get-device-definition-version \
  --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd" \
  --device-definition-version-id "83c13984-6fed-447e-84d5-5b8aa45d5f71"
```

Salida:

```
{
  "Definition": {
    "Devices": [
      {
        "CertificateArn": "arn:aws:iot:us-west-2:123456789012:cert/6c52ce1b47bde88a637e9ccdd45fe4e4c2c0a75a6866f8f63d980ee22fa51e02",
        "ThingArn": "arn:aws:iot:us-west-2:123456789012:thing/ExteriorTherm",
        "SyncShadow": true,

```

```

        "Id": "ExteriorTherm"
      },
      {
        "CertificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92",
        "ThingArn": "arn:aws:iot:us-west-2:123456789012:thing/
InteriorTherm",
        "SyncShadow": true,
        "Id": "InteriorTherm"
      }
    ]
  },
  "Version": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
  "CreationTimestamp": "2019-09-11T00:15:09.838Z",
  "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/83c13984-6fed-447e-84d5-5b8aa45d5f71"
}

```

- Para API obtener más información, consulte [GetDeviceDefinitionVersion](#) la Referencia de AWS CLI comandos.

## get-device-definition

En el siguiente ejemplo de código, se muestra cómo usar `get-device-definition`.

### AWS CLI

Para obtener una definición de dispositivo

En el siguiente `get-device-definition` ejemplo, se recupera información sobre la definición de dispositivo especificada. Para recuperar las definiciones IDs de sus dispositivos, utilice el `list-device-definitions` comando.

```
aws greengrass get-device-definition \
  --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd"
```

Salida:

```
{
```

```

    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/83c13984-6fed-447e-84d5-5b8aa45d5f71",
    "Name": "TemperatureSensors",
    "tags": {},
    "LastUpdatedTimestamp": "2019-09-11T00:19:03.698Z",
    "LatestVersion": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
    "CreationTimestamp": "2019-09-11T00:11:06.197Z",
    "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd"
}

```

- Para API obtener más información, consulte [GetDeviceDefinition](#) la Referencia de AWS CLI comandos.

## get-function-definition-version

En el siguiente ejemplo de código, se muestra cómo usar `get-function-definition-version`.

### AWS CLI

Para recuperar detalles sobre una versión específica de una función Lambda

A continuación, se `get-function-definition-version` recupera información sobre la versión especificada de la definición de función especificada. Para recuperar todas las versiones de la definición de función, utilice el `list-function-definition-versions` comando. IDs Para recuperar el identificador de la última versión añadida a la definición de la función, utilice el `get-function-definition` comando y compruebe la `LatestVersion` propiedad.

```

aws greengrass get-function-definition-version \
  --function-definition-id "063f5d1a-1dd1-40b4-9b51-56f8993d0f85" \
  --function-definition-version-id "9748fda7-1589-4fcc-ac94-f5559e88678b"

```

Salida:

```

{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/versions/9748fda7-1589-4fcc-ac94-
f5559e88678b",
  "CreationTimestamp": "2019-06-18T17:04:30.776Z",

```



```

"Definition": {
  "Functions": [
    {
      "FunctionArn": "arn:aws:lambda::function:GGIPDetector:1",
      "FunctionConfiguration": {
        "Environment": {},
        "MemorySize": 32768,
        "Pinned": true,
        "Timeout": 3
      },
      "Id": "26b69bdb-e547-46bc-9812-84ec04b6cc8c"
    },
    {
      "FunctionArn": "arn:aws:lambda:us-
west-2:123456789012:function:Greengrass_HelloWorld:GG_HelloWorld",
      "FunctionConfiguration": {
        "EncodingType": "json",
        "Environment": {
          "Variables": {}
        },
        "MemorySize": 16384,
        "Pinned": true,
        "Timeout": 25
      },
      "Id": "384465a8-eedf-48c6-b793-4c35f7bfae9b"
    }
  ]
},
"Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
"Version": "9748fda7-1589-4fcc-ac94-f5559e88678b"
}

```

- Para API obtener más información, consulte [GetFunctionDefinitionVersion](#) la Referencia de AWS CLI comandos.

## get-function-definition

En el siguiente ejemplo de código, se muestra cómo usar `get-function-definition`.

### AWS CLI

Para recuperar una definición de función

En el siguiente `get-function-definition` ejemplo, se muestran los detalles de la definición de función especificada. Para recuperar las definiciones IDs de funciones, utilice el `list-function-definitions` comando.

```
aws greengrass get-function-definition \  
  --function-definition-id "063f5d1a-1dd1-40b4-9b51-56f8993d0f85"
```

Salida:

```
{  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/  
functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85",  
  "CreationTimestamp": "2019-06-18T16:21:21.431Z",  
  "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",  
  "LastUpdatedTimestamp": "2019-06-18T16:21:21.431Z",  
  "LatestVersion": "9748fda7-1589-4fcc-ac94-f5559e88678b",  
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/  
versions/9748fda7-1589-4fcc-ac94-f5559e88678b",  
  "tags": {}  
}
```

- Para API obtener más información, consulte [GetFunctionDefinition](#) la Referencia de AWS CLI comandos.

## get-group-certificate-authority

En el siguiente ejemplo de código, se muestra cómo usar `get-group-certificate-authority`.

### AWS CLI

Para recuperar la CA asociada a un grupo de Greengrass

En el siguiente `get-group-certificate-authority` ejemplo, se recupera la entidad de certificación (CA) asociada al grupo de Greengrass especificado. Para obtener el ID de la autoridad de certificación, utilice el `list-group-certificate-authorities` comando y especifique el ID del grupo.

```
aws greengrass get-group-certificate-authority \  
  --group-id "1013db12-8b58-45ff-acc7-704248f66731" \  
  --certificate-authority-id "1013db12-8b58-45ff-acc7-704248f66731"
```

```
--certificate-authority-
id "f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6"
```

Salida:

```
{
  "GroupCertificateAuthorityArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/certificateauthorities/f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6",
  "GroupCertificateAuthorityId":
  "f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6",
  "PemEncodedCertificate": "-----BEGIN CERTIFICATE-----
MIICiTCCAFICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBWEXAMPLEGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDEXAMPLEEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAEXAMPLESBDdb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWxhZAd
BgkqhkiG9w0BCQEWEG5vb25lQGFTYXpvbi5jEXAMPLENMTewNDI1MjA0NTIxWhcN
MTIwNDI0MjA0EXAMPLEBIDELMAKGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWEXAMPLEDASBgNVBAwTC01BTSBDdb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWEXAMPLEEgkqhkiG9w0BCQEWEG5vb25lQGFT
YXpvbi5EXAMPLE8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CEXAMPLE93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswYEXAMPLEEgpE
Ibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKEXAMPLEAQEFBQADgYEAtCu4
nUHVvXyUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFbjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----\n"
}
```

- Para API obtener más información, consulte [GetGroupCertificateAuthority](#) la Referencia de AWS CLI comandos.

## get-group-certificate-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-group-certificate-configuration`.

### AWS CLI

Para recuperar la configuración de la autoridad de certificación utilizada por el grupo Greengrass

El siguiente `get-group-certificate-configuration` ejemplo recupera la configuración de la autoridad de certificación (CA) utilizada por el grupo de Greengrass especificado.

```
aws greengrass get-group-certificate-configuration \  
  --group-id "1013db12-8b58-45ff-acc7-704248f66731"
```

Salida:

```
{  
  "CertificateAuthorityExpiryInMilliseconds": 2524607999000,  
  "CertificateExpiryInMilliseconds": 604800000,  
  "GroupId": "1013db12-8b58-45ff-acc7-704248f66731"  
}
```

- Para API obtener más información, consulte la Referencia [GetGroupCertificateConfiguration](#) de AWS CLI comandos.

## get-group-version

En el siguiente ejemplo de código, se muestra cómo usar `get-group-version`.

AWS CLI

Para recuperar información sobre una versión de un grupo de Greengrass

El siguiente `get-group-version` ejemplo recupera información sobre la versión especificada del grupo especificado. Para recuperar todas IDs las versiones del grupo, utilice el `list-group-versions` comando. Para recuperar el ID de la última versión agregada al grupo, utilice el `get-group` comando y compruebe la `LatestVersion` propiedad.

```
aws greengrass get-group-version \  
  --group-id "1013db12-8b58-45ff-acc7-704248f66731" \  
  --group-version-id "115136b3-cfd7-4462-b77f-8741a4b00e5e"
```

Salida:

```
{  
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-  
b77f-8741a4b00e5e",
```

```

    "CreationTimestamp": "2019-06-18T17:04:30.915Z",
    "Definition": {
      "CoreDefinitionVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/c906ed39-a1e3-4822-a981-7b9bd57b4b46/versions/42aeeac3-fd9d-4312-a8fd-ffa9404a20e0",
      "FunctionDefinitionVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/versions/9748fda7-1589-4fcc-ac94-f5559e88678b",
      "SubscriptionDefinitionVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/versions/88ae8699-12ac-4663-ba3f-4d7f0519140b"
    },
    "Id": "1013db12-8b58-45ff-acc7-704248f66731",
    "Version": "115136b3-cfd7-4462-b77f-8741a4b00e5e"
  }
}

```

- Para API obtener más información, consulte [GetGroupVersion](#) la Referencia de AWS CLI comandos.

## get-group

En el siguiente ejemplo de código, se muestra cómo usar `get-group`.

### AWS CLI

Para recuperar información sobre un grupo de Greengrass

El siguiente `get-group` ejemplo recupera información sobre el grupo Greengrass especificado. Para recuperar uno IDs de sus grupos, utilice el `list-groups` comando.

```

aws greengrass get-group \
  --group-id "1013db12-8b58-45ff-acc7-704248f66731"

```

Salida:

```

{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731",
  "CreationTimestamp": "2019-06-18T16:21:21.457Z",
  "Id": "1013db12-8b58-45ff-acc7-704248f66731",
  "LastUpdatedTimestamp": "2019-06-18T16:21:21.457Z",
}

```

```

    "LatestVersion": "115136b3-cfd7-4462-b77f-8741a4b00e5e",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-
b77f-8741a4b00e5e",
    "Name": "GGGroup4Pi3",
    "tags": {}
}

```

- Para API obtener más información, consulte [GetGroup](#) la Referencia de AWS CLI comandos.

## get-logger-definition-version

En el siguiente ejemplo de código, se muestra cómo usar `get-logger-definition-version`.

### AWS CLI

Para recuperar información sobre una versión de una definición de registrador

El siguiente `get-logger-definition-version` ejemplo recupera información sobre la versión especificada de la definición de registrador especificada. Para recuperar todas las versiones IDs de la definición del registrador, utilice el comando `list-logger-definition-versions` Para recuperar el ID de la última versión añadida a la definición del registrador, utilice el `get-logger-definition` comando y compruebe la `LatestVersion` propiedad.

```

aws greengrass get-logger-definition-version \
  --logger-definition-id "49eeeb66-f1d3-4e34-86e3-3617262abf23" \
  --logger-definition-version-id "5e3f6f64-a565-491e-8de0-3c0d8e0f2073"

```

Salida:

```

{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/versions/5e3f6f64-
a565-491e-8de0-3c0d8e0f2073",
  "CreationTimestamp": "2019-05-08T16:10:13.866Z",
  "Definition": {
    "Loggers": []
  },
  "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
  "Version": "5e3f6f64-a565-491e-8de0-3c0d8e0f2073"
}

```

- Para API obtener más información, consulte [GetLoggerDefinitionVersion](#) la Referencia de AWS CLI comandos.

## get-logger-definition

En el siguiente ejemplo de código, se muestra cómo usar `get-logger-definition`.

### AWS CLI

Para recuperar información sobre la definición de un registrador

El siguiente `get-logger-definition` ejemplo recupera información sobre la definición de registrador especificada. Para recuperar las definiciones IDs de su registrador, utilice el comando. `list-logger-definitions`

```
aws greengrass get-logger-definition \
  --logger-definition-id "49eeeb66-f1d3-4e34-86e3-3617262abf23"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23",
  "CreationTimestamp": "2019-05-08T16:10:13.809Z",
  "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
  "LastUpdatedTimestamp": "2019-05-08T16:10:13.809Z",
  "LatestVersion": "5e3f6f64-a565-491e-8de0-3c0d8e0f2073",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/versions/5e3f6f64-
a565-491e-8de0-3c0d8e0f2073",
  "tags": {}
}
```

- Para API obtener más información, consulte [GetLoggerDefinition](#) la Referencia de AWS CLI comandos.

## get-resource-definition-version

En el siguiente ejemplo de código, se muestra cómo usar `get-resource-definition-version`.

## AWS CLI

Para recuperar información sobre una versión específica de una definición de recurso

El siguiente `get-resource-definition-version` ejemplo recupera información sobre la versión especificada de la definición de recurso especificada. Para recuperar todas las versiones de la definición de recurso, utilice el `list-resource-definition-versions` comando. IDs Para recuperar el ID de la última versión agregada a la definición del recurso, utilice el `get-resource-definition` comando y compruebe la `LatestVersion` propiedad.

```
aws greengrass get-resource-definition-version \
  --resource-definition-id "ad8c101d-8109-4b0e-b97d-9cc5802ab658" \
  --resource-definition-version-id "26e8829a-491a-464d-9c87-664bf6f6f2be"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/
versions/26e8829a-491a-464d-9c87-664bf6f6f2be",
  "CreationTimestamp": "2019-06-19T16:40:59.392Z",
  "Definition": {
    "Resources": [
      {
        "Id": "26ff3f7b-839a-4217-9fdc-a218308b3963",
        "Name": "usb-port",
        "ResourceDataContainer": {
          "LocalDeviceResourceData": {
            "GroupOwnerSetting": {
              "AutoAddGroupOwner": false
            },
            "SourcePath": "/dev/bus/usb"
          }
        }
      }
    ]
  },
  "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
  "Version": "26e8829a-491a-464d-9c87-664bf6f6f2be"
}
```



- Para API obtener más información, consulte [GetResourceDefinitionVersion](#) la Referencia de AWS CLI comandos.

## get-resource-definition

En el siguiente ejemplo de código, se muestra cómo usar `get-resource-definition`.

### AWS CLI

Para recuperar información sobre una definición de recurso

El siguiente `get-resource-definition` ejemplo recupera información sobre la definición de recurso especificada. Para recuperar las definiciones IDs de recursos, utilice el `list-resource-definitions` comando.

```
aws greengrass get-resource-definition \
  --resource-definition-id "ad8c101d-8109-4b0e-b97d-9cc5802ab658"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658",
  "CreationTimestamp": "2019-06-19T16:40:59.261Z",
  "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
  "LastUpdatedTimestamp": "2019-06-19T16:40:59.261Z",
  "LatestVersion": "26e8829a-491a-464d-9c87-664bf6f6f2be",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/
versions/26e8829a-491a-464d-9c87-664bf6f6f2be",
  "tags": {}
}
```

- Para API obtener más información, consulte [GetResourceDefinition](#) la Referencia de AWS CLI comandos.

## get-service-role-for-account

En el siguiente ejemplo de código, se muestra cómo usar `get-service-role-for-account`.

## AWS CLI

Para recuperar los detalles del rol de servicio asociado a su cuenta

En el siguiente `get-service-role-for-account` ejemplo, se recupera información sobre la función de servicio asociada a su AWS cuenta.

```
aws greengrass get-service-role-for-account
```

Salida:

```
{
  "AssociatedAt": "2018-10-18T15:59:20Z",
  "RoleArn": "arn:aws:iam::123456789012:role/service-role/Greengrass_ServiceRole"
}
```

Para obtener más información, consulte el [rol de servicio de Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [GetServiceRoleForAccount](#) de AWS CLI comandos.

## get-subscription-definition-version

En el siguiente ejemplo de código, se muestra cómo usar `get-subscription-definition-version`.

### AWS CLI

Para recuperar información sobre una versión específica de una definición de suscripción

El siguiente `get-subscription-definition-version` ejemplo recupera información sobre la versión especificada de la definición de suscripción especificada. Para recuperar todas las versiones IDs de la definición de suscripción, utilice el `list-subscription-definition-versions` comando. Para recuperar el ID de la última versión agregada a la definición de suscripción, utilice el `get-subscription-definition` comando y compruebe la `LatestVersion` propiedad.

```
aws greengrass get-subscription-definition-version \
  --subscription-definition-id "70e49321-83d5-45d2-bc09-81f4917ae152" \
```

```
--subscription-definition-version-id "88ae8699-12ac-4663-ba3f-4d7f0519140b"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/versions/88ae8699-12ac-4663-
ba3f-4d7f0519140b",
  "CreationTimestamp": "2019-06-18T17:03:52.499Z",
  "Definition": {
    "Subscriptions": [
      {
        "Id": "692c4484-d89f-4f64-8edd-1a041a65e5b6",
        "Source": "arn:aws:lambda:us-
west-2:123456789012:function:Greengrass_HelloWorld:GG_HelloWorld",
        "Subject": "hello/world",
        "Target": "cloud"
      }
    ]
  },
  "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
  "Version": "88ae8699-12ac-4663-ba3f-4d7f0519140b"
}
```

- Para API obtener más información, consulte [GetSubscriptionDefinitionVersion](#) la Referencia de AWS CLI comandos.

## get-subscription-definition

En el siguiente ejemplo de código, se muestra cómo usar `get-subscription-definition`.

### AWS CLI

Para recuperar información sobre una definición de suscripción

En el siguiente `get-subscription-definition` ejemplo, se recupera información sobre la definición de suscripción especificada. Para recuperar las definiciones IDs de suscripción, utilice el `list-subscription-definitions` comando.

```
aws greengrass get-subscription-definition \
--subscription-definition-id "70e49321-83d5-45d2-bc09-81f4917ae152"
```

Salida:

```
{
  "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152",
  "CreationTimestamp": "2019-06-18T17:03:52.392Z",
  "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
  "LastUpdatedTimestamp": "2019-06-18T17:03:52.392Z",
  "LatestVersion": "88ae8699-12ac-4663-ba3f-4d7f0519140b",
  "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/
versions/88ae8699-12ac-4663-ba3f-4d7f0519140b",
  "tags": {}
}
```

- Para API obtener más información, consulte [GetSubscriptionDefinition](#) la Referencia de AWS CLI comandos.

## get-thing-runtime-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-thing-runtime-configuration`.

### AWS CLI

Para recuperar la configuración de tiempo de ejecución de un núcleo de Greengrass

El siguiente `get-thing-runtime-configuration` ejemplo recupera la configuración de tiempo de ejecución de un núcleo de Greengrass. Antes de poder recuperar la configuración de tiempo de ejecución, debe usar el `update-thing-runtime-configuration` comando para crear una configuración de tiempo de ejecución para el núcleo.

```
aws greengrass get-thing-runtime-configuration \
  --thing-name SampleGreengrassCore
```

Salida:

```
{
  "RuntimeConfiguration": {
    "TelemetryConfiguration": {
      "ConfigurationSyncStatus": "OutOfSync",
```

```

        "Telemetry": "On"
      }
    }
  }
}

```

Para obtener más información, consulte [Configuración de los ajustes de telemetría en la Guía para desarrolladores de IoT AWS Greengrass](#).

- Para API obtener más información, consulte la Referencia de [GetThingRuntimeConfiguration](#) comandos AWS CLI .

## list-bulk-deployment-detailed-reports

En el siguiente ejemplo de código, se muestra cómo usar `list-bulk-deployment-detailed-reports`.

### AWS CLI

Para incluir información sobre las implementaciones individuales en una implementación masiva

En el siguiente `list-bulk-deployment-detailed-reports` ejemplo, se muestra información sobre las implementaciones individuales de una operación de despliegue masivo, incluido su estado.

```

aws greengrass list-bulk-deployment-detailed-reports \
  --bulk-deployment-id 42ce9c42-489b-4ed4-b905-8996aa50ef9d

```

Salida:

```

{
  "Deployments": [
    {
      "DeploymentType": "NewDeployment",
      "DeploymentStatus": "Success",
      "DeploymentId": "123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "DeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333/deployments/123456789012:123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "GroupArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333/versions/123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
    }
  ]
}

```

```

        "CreatedAt": "2020-01-21T21:34:16.501Z"
    },
    {
        "DeploymentType": "NewDeployment",
        "DeploymentStatus": "InProgress",
        "DeploymentId": "123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "DeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/a1b2c3d4-5678-90ab-cdef-EXAMPLE55555/
deployments/123456789012:123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "GroupArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/a1b2c3d4-5678-90ab-cdef-EXAMPLE55555/versions/a1b2c3d4-5678-90ab-cdef-
EXAMPLE66666",
        "CreatedAt": "2020-01-21T21:34:16.486Z"
    },
    ...
]
}

```

Para obtener más información, consulte [Crear despliegues masivos para grupos](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte la Referencia [ListBulkDeploymentDetailedReports](#) de AWS CLI comandos.

## list-bulk-deployments

En el siguiente ejemplo de código, se muestra cómo usar `list-bulk-deployments`.

### AWS CLI

Para ver una lista de las implementaciones masivas

En el siguiente `list-bulk-deployments` ejemplo, se enumeran todos los despliegues masivos.

```
aws greengrass list-bulk-deployments
```

Salida:

```
{
  "BulkDeployments": [
```

```

    {
      "BulkDeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/bulk/deployments/870fb41b-6288-4e0c-bc76-a7ba4b4d3267",
      "BulkDeploymentId": "870fb41b-6288-4e0c-bc76-a7ba4b4d3267",
      "CreatedAt": "2019-06-25T16:11:33.265Z"
    }
  ]
}

```

Para obtener más información, consulte [Crear despliegues masivos para grupos](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte la Referencia [ListBulkDeployments](#) de AWS CLI comandos.

## list-connector-definition-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-connector-definition-versions`.

### AWS CLI

Para ver una lista de las versiones disponibles para una definición de conector

En el siguiente `list-connector-definition-versions` ejemplo, se enumeran las versiones disponibles para la definición de conector especificada. Utilice el `list-connector-definitions` comando para obtener el ID de definición del conector.

```

aws greengrass list-connector-definition-versions \
  --connector-definition-id "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8"

```

Salida:

```

{
  "Versions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/versions/63c57963-c7c2-4a26-a7e2-7bf478ea2623",
      "CreationTimestamp": "2019-06-19T19:30:01.300Z",
      "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
    }
  ]
}

```

```
        "Version": "63c57963-c7c2-4a26-a7e2-7bf478ea2623"
      }
    ]
  }
```

Para obtener más información, consulte [Integración con servicios y protocolos mediante conectores Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [ListConnectorDefinitionVersions](#) de AWS CLI comandos.

## list-connector-definitions

En el siguiente ejemplo de código, se muestra cómo usar `list-connector-definitions`.

AWS CLI

Para enumerar los conectores de Greengrass que están definidos

En el siguiente `list-connector-definitions` ejemplo, se enumeran todos los conectores de Greengrass que están definidos para su AWS cuenta.

```
aws greengrass list-connector-definitions
```

Salida:

```
{
  "Definitions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
      "CreationTimestamp": "2019-06-19T19:30:01.300Z",
      "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
      "LastUpdatedTimestamp": "2019-06-19T19:30:01.300Z",
      "LatestVersion": "63c57963-c7c2-4a26-a7e2-7bf478ea2623",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/
versions/63c57963-c7c2-4a26-a7e2-7bf478ea2623",
      "Name": "MySNSConnector"
    }
  ]
}
```



```
}
```

Para obtener más información, consulte [Integración con servicios y protocolos mediante conectores Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [ListConnectorDefinitions](#) de AWS CLI comandos.

## list-core-definition-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-core-definition-versions`.

### AWS CLI

Para enumerar las versiones de una definición básica de Greengrass

En el siguiente `list-core-definitions` ejemplo, se enumeran todas las versiones de la definición básica de Greengrass especificada. Puede usar el `list-core-definitions` comando para obtener el ID de la versión.

```
aws greengrass list-core-definition-versions \  
  --core-definition-id "eaf280cb-138c-4d15-af36-6f681a1348f7"
```

Salida:

```
{  
  "Versions": [  
    {  
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
definition/cores/eaf280cb-138c-4d15-af36-6f681a1348f7/versions/467c36e4-c5da-440c-  
a97b-084e62593b4c",  
      "CreationTimestamp": "2019-06-18T16:14:17.709Z",  
      "Id": "eaf280cb-138c-4d15-af36-6f681a1348f7",  
      "Version": "467c36e4-c5da-440c-a97b-084e62593b4c"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListCoreDefinitionVersions](#) la Referencia de AWS CLI comandos.

## list-core-definitions

En el siguiente ejemplo de código, se muestra cómo usar `list-core-definitions`.

### AWS CLI

Para enumerar las definiciones principales de Greengrass

En el siguiente `list-core-definitions` ejemplo, se enumeran todas las definiciones principales de Greengrass para su AWS cuenta.

```
aws greengrass list-core-definitions
```

Salida:

```
{
  "Definitions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/cores/0507843c-c1ef-4f06-b051-817030df7e7d",
      "CreationTimestamp": "2018-10-17T04:30:32.786Z",
      "Id": "0507843c-c1ef-4f06-b051-817030df7e7d",
      "LastUpdatedTimestamp": "2018-10-17T04:30:32.786Z",
      "LatestVersion": "bcdf9e86-3793-491e-93af-3cdfbf4e22b7",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/0507843c-c1ef-4f06-b051-817030df7e7d/versions/
bcdf9e86-3793-491e-93af-3cdfbf4e22b7"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/cores/31c22500-3509-4271-bafd-cf0655cda438",
      "CreationTimestamp": "2019-06-18T16:24:16.064Z",
      "Id": "31c22500-3509-4271-bafd-cf0655cda438",
      "LastUpdatedTimestamp": "2019-06-18T16:24:16.064Z",
      "LatestVersion": "2f350395-6d09-4c8a-8336-9ae5b57ace84",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/31c22500-3509-4271-bafd-cf0655cda438/
versions/2f350395-6d09-4c8a-8336-9ae5b57ace84"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/cores/c906ed39-a1e3-4822-a981-7b9bd57b4b46",
      "CreationTimestamp": "2019-06-18T16:21:21.351Z",
```

```

        "Id": "c906ed39-a1e3-4822-a981-7b9bd57b4b46",
        "LastUpdatedTimestamp": "2019-06-18T16:21:21.351Z",
        "LatestVersion": "42aeeac3-fd9d-4312-a8fd-ffa9404a20e0",
        "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/c906ed39-a1e3-4822-a981-7b9bd57b4b46/versions/42aeeac3-
fd9d-4312-a8fd-ffa9404a20e0"
    },
    {
        "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/cores/eaf280cb-138c-4d15-af36-6f681a1348f7",
        "CreationTimestamp": "2019-06-18T16:14:17.709Z",
        "Id": "eaf280cb-138c-4d15-af36-6f681a1348f7",
        "LastUpdatedTimestamp": "2019-06-18T16:14:17.709Z",
        "LatestVersion": "467c36e4-c5da-440c-a97b-084e62593b4c",
        "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/eaf280cb-138c-4d15-af36-6f681a1348f7/versions/467c36e4-
c5da-440c-a97b-084e62593b4c"
    }
]
}

```

- Para API obtener más información, consulte [ListCoreDefinitions](#) la Referencia de AWS CLI comandos.

## list-deployments

En el siguiente ejemplo de código, se muestra cómo usar `list-deployments`.

### AWS CLI

Para enumerar las implementaciones de un grupo de Greengrass

En el siguiente `list-deployments` ejemplo, se enumeran las implementaciones del grupo de Greengrass especificado. Puede usar el `list-groups` comando para buscar el ID de su grupo.

```

aws greengrass list-deployments \
  --group-id "1013db12-8b58-45ff-acc7-704248f66731"

```

Salida:

```
{
```

```

    "Deployments": [
      {
        "CreatedAt": "2019-06-18T17:04:32.702Z",
        "DeploymentId": "1065b8a0-812b-4f21-9d5d-e89b232a530f",
        "DeploymentType": "NewDeployment",
        "GroupArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-
b77f-8741a4b00e5e"
      }
    ]
  }

```

- Para API obtener más información, consulte [ListDeployments](#) la Referencia de AWS CLI comandos.

## list-device-definition-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-device-definition-versions`.

### AWS CLI

Para ver una lista de las versiones de una definición de dispositivo

En el siguiente `list-device-definition-versions` ejemplo, se muestran las versiones de la definición de dispositivo asociadas a la definición de dispositivo especificada.

```

aws greengrass list-device-definition-versions \
  --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd"

```

Salida:

```

{
  "Versions": [
    {
      "Version": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
      "CreationTimestamp": "2019-09-11T00:15:09.838Z",
      "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/83c13984-6fed-447e-84d5-5b8aa45d5f71"
    },
    {

```

```

    "Version": "3b5cc510-58c1-44b5-9d98-4ad858ffa795",
    "CreationTimestamp": "2019-09-11T00:11:06.197Z",
    "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/3b5cc510-58c1-44b5-9d98-4ad858ffa795"
  }
]
}

```

- Para API obtener más información, consulte [ListDeviceDefinitionVersions](#) la Referencia de AWS CLI comandos.

## list-device-definitions

En el siguiente ejemplo de código, se muestra cómo usar `list-device-definitions`.

### AWS CLI

Para enumerar las definiciones de sus dispositivos

En el siguiente `list-device-definitions` ejemplo, se muestran detalles sobre las definiciones de dispositivos de tu AWS cuenta en la AWS región especificada.

```

aws greengrass list-device-definitions \
  --region us-west-2

```

Salida:

```

{
  "Definitions": [
    {
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/devices/50f3274c-3f0a-4f57-b114-6f46085281ab/versions/
c777b0f5-1059-449b-beaa-f003ebc56c34",
      "LastUpdatedTimestamp": "2019-06-14T15:42:09.059Z",
      "LatestVersion": "c777b0f5-1059-449b-beaa-f003ebc56c34",
      "CreationTimestamp": "2019-06-14T15:42:09.059Z",
      "Id": "50f3274c-3f0a-4f57-b114-6f46085281ab",
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/devices/50f3274c-3f0a-4f57-b114-6f46085281ab"
    },
  ],
}

```

```

    {
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/e01951c9-6134-479a-969a-1a15cac11c40/versions/514d57aa-4ee6-401c-9fac-938a9f7a51e5",
      "Name": "TestDeviceDefinition",
      "LastUpdatedTimestamp": "2019-04-16T23:17:43.245Z",
      "LatestVersion": "514d57aa-4ee6-401c-9fac-938a9f7a51e5",
      "CreationTimestamp": "2019-04-16T23:17:43.245Z",
      "Id": "e01951c9-6134-479a-969a-1a15cac11c40",
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/e01951c9-6134-479a-969a-1a15cac11c40"
    },
    {
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/versions/83c13984-6fed-447e-84d5-5b8aa45d5f71",
      "Name": "TemperatureSensors",
      "LastUpdatedTimestamp": "2019-09-10T00:19:03.698Z",
      "LatestVersion": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
      "CreationTimestamp": "2019-09-11T00:11:06.197Z",
      "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd"
    }
  ]
}

```

- Para API obtener más información, consulte [ListDeviceDefinitions](#) la Referencia de AWS CLI comandos.

## list-function-definition-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-function-definition-versions`.

### AWS CLI

Para enumerar las versiones de una función Lambda

En el siguiente `list-function-definition-versions` ejemplo, se enumeran todas las versiones de la función Lambda especificada. Puede usar el `list-function-definitions` comando para obtener el ID.

```
aws greengrass list-function-definition-versions \  
--function-definition-id "063f5d1a-1dd1-40b4-9b51-56f8993d0f85"
```

Salida:

```
{  
  "Versions": [  
    {  
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/  
versions/9748fda7-1589-4fcc-ac94-f5559e88678b",  
      "CreationTimestamp": "2019-06-18T17:04:30.776Z",  
      "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",  
      "Version": "9748fda7-1589-4fcc-ac94-f5559e88678b"  
    },  
    {  
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/  
versions/9b08df77-26f2-4c29-93d2-769715edcfec",  
      "CreationTimestamp": "2019-06-18T17:02:44.087Z",  
      "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",  
      "Version": "9b08df77-26f2-4c29-93d2-769715edcfec"  
    },  
    {  
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/  
versions/4236239f-94f7-4b90-a2f8-2a24c829d21e",  
      "CreationTimestamp": "2019-06-18T17:01:42.284Z",  
      "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",  
      "Version": "4236239f-94f7-4b90-a2f8-2a24c829d21e"  
    },  
    {  
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/  
versions/343408bb-549a-4fbe-b043-853643179a39",  
      "CreationTimestamp": "2019-06-18T16:21:21.431Z",  
      "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",  
      "Version": "343408bb-549a-4fbe-b043-853643179a39"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListFunctionDefinitionVersions](#) la Referencia de AWS CLI comandos.

## list-function-definitions

En el siguiente ejemplo de código, se muestra cómo usar `list-function-definitions`.

### AWS CLI

Para enumerar las funciones Lambda

En el siguiente `list-function-definitions` ejemplo, se enumeran todas las funciones de Lambda definidas para su AWS cuenta.

```
aws greengrass list-function-definitions
```

Salida:

```
{
  "Definitions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/017970a5-8952-46dd-b1c1-020b3ae8e960",
      "CreationTimestamp": "2018-10-17T04:30:32.884Z",
      "Id": "017970a5-8952-46dd-b1c1-020b3ae8e960",
      "LastUpdatedTimestamp": "2018-10-17T04:30:32.884Z",
      "LatestVersion": "4380b302-790d-4ed8-92bf-02e88afecb15",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/017970a5-8952-46dd-b1c1-020b3ae8e960/
versions/4380b302-790d-4ed8-92bf-02e88afecb15"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
      "CreationTimestamp": "2019-06-18T16:21:21.431Z",
      "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
      "LastUpdatedTimestamp": "2019-06-18T16:21:21.431Z",
      "LatestVersion": "9748fda7-1589-4fcc-ac94-f5559e88678b",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/
versions/9748fda7-1589-4fcc-ac94-f5559e88678b"
    },
  ],
}
```



```

    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/6598e653-a262-440c-9967-e2697f64da7b",
      "CreationTimestamp": "2019-06-18T16:24:16.123Z",
      "Id": "6598e653-a262-440c-9967-e2697f64da7b",
      "LastUpdatedTimestamp": "2019-06-18T16:24:16.123Z",
      "LatestVersion": "38bc6ccd-98a2-4ce7-997e-16c84748fae4",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/6598e653-a262-440c-9967-e2697f64da7b/
versions/38bc6ccd-98a2-4ce7-997e-16c84748fae4"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/c668df84-fad2-491b-95f4-655d2cad7885",
      "CreationTimestamp": "2019-06-18T16:14:17.784Z",
      "Id": "c668df84-fad2-491b-95f4-655d2cad7885",
      "LastUpdatedTimestamp": "2019-06-18T16:14:17.784Z",
      "LatestVersion": "37dd68c4-a64f-40ba-aa13-71fecc3ebded",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/c668df84-fad2-491b-95f4-655d2cad7885/
versions/37dd68c4-a64f-40ba-aa13-71fecc3ebded"
    }
  ]
}

```

- Para API obtener más información, consulte [ListFunctionDefinitions](#) la Referencia de AWS CLI comandos.

## list-group-certificate-authorities

En el siguiente ejemplo de código, se muestra cómo usar `list-group-certificate-authorities`.

### AWS CLI

Para enumerar la corriente CAs de un grupo

En el siguiente `list-group-certificate-authorities` ejemplo, se enumeran las autoridades de certificación actuales (CAs) del grupo Greengrass especificado.

```

aws greengrass list-group-certificate-authorities \
  --group-id "1013db12-8b58-45ff-acc7-704248f66731"

```

**Salida:**

```
{
  "GroupCertificateAuthorities": [
    {
      "GroupCertificateAuthorityArn": "arn:aws:greengrass:us-
west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/
certificateauthorities/
f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6",
      "GroupCertificateAuthorityId":
      "f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6"
    }
  ]
}
```

- Para API obtener más información, consulte [ListGroupCertificateAuthorities](#) la Referencia de AWS CLI comandos.

**list-group-versions**

En el siguiente ejemplo de código, se muestra cómo usar `list-group-versions`.

**AWS CLI**

Para enumerar las versiones de un grupo de Greengrass

El siguiente `list-group-versions` ejemplo muestra las versiones del grupo Greengrass especificado.

```
aws greengrass list-group-versions \
  --group-id "1013db12-8b58-45ff-acc7-704248f66731"
```

**Salida:**

```
{
  "Versions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-
b77f-8741a4b00e5e",
      "CreationTimestamp": "2019-06-18T17:04:30.915Z",
      "Id": "1013db12-8b58-45ff-acc7-704248f66731",
    }
  ]
}
```

```

    "Version": "115136b3-cfd7-4462-b77f-8741a4b00e5e"
  },
  {
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/versions/4340669d-d14d-44e3-920c-46c928750750",
    "CreationTimestamp": "2019-06-18T17:03:52.663Z",
    "Id": "1013db12-8b58-45ff-acc7-704248f66731",
    "Version": "4340669d-d14d-44e3-920c-46c928750750"
  },
  {
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/versions/1b06e099-2d5b-4f10-91b9-78c4e060f5da",
    "CreationTimestamp": "2019-06-18T17:02:44.189Z",
    "Id": "1013db12-8b58-45ff-acc7-704248f66731",
    "Version": "1b06e099-2d5b-4f10-91b9-78c4e060f5da"
  },
  {
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/versions/2d3f27f1-3b43-4554-ab7a-73ec30477efe",
    "CreationTimestamp": "2019-06-18T17:01:42.401Z",
    "Id": "1013db12-8b58-45ff-acc7-704248f66731",
    "Version": "2d3f27f1-3b43-4554-ab7a-73ec30477efe"
  },
  {
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/versions/d20f7ae9-3444-4c1c-b025-e2ede23cdd31",
    "CreationTimestamp": "2019-06-18T16:21:21.457Z",
    "Id": "1013db12-8b58-45ff-acc7-704248f66731",
    "Version": "d20f7ae9-3444-4c1c-b025-e2ede23cdd31"
  }
]
}

```

- Para API obtener más información, consulte [ListGroupVersions](#) la Referencia de AWS CLI comandos.

## list-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-groups`.

## AWS CLI

Para enumerar los grupos de Greengrass

El siguiente `list-groups` ejemplo muestra todos los grupos de Greengrass que están definidos en su AWS cuenta.

```
aws greengrass list-groups
```

Salida:

```
{
  "Groups": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731",
      "CreationTimestamp": "2019-06-18T16:21:21.457Z",
      "Id": "1013db12-8b58-45ff-acc7-704248f66731",
      "LastUpdatedTimestamp": "2019-06-18T16:21:21.457Z",
      "LatestVersion": "115136b3-cfd7-4462-b77f-8741a4b00e5e",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-b77f-8741a4b00e5e",
      "Name": "GGGroup4Pi3"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1402daf9-71cf-4cfe-8be0-d5e80526d0d8",
      "CreationTimestamp": "2018-10-31T21:52:46.603Z",
      "Id": "1402daf9-71cf-4cfe-8be0-d5e80526d0d8",
      "LastUpdatedTimestamp": "2018-10-31T21:52:46.603Z",
      "LatestVersion": "749af901-60ab-456f-a096-91b12d983c29",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/1402daf9-71cf-4cfe-8be0-d5e80526d0d8/versions/749af901-60ab-456f-a096-91b12d983c29",
      "Name": "MyTestGroup"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/504b5c8d-bbed-4635-aff1-48ec5b586db5",
      "CreationTimestamp": "2018-12-31T21:39:36.771Z",
      "Id": "504b5c8d-bbed-4635-aff1-48ec5b586db5",
      "LastUpdatedTimestamp": "2018-12-31T21:39:36.771Z",

```

```

        "LatestVersion": "46911e8e-f9bc-4898-8b63-59c7653636ec",
        "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/504b5c8d-bbed-4635-aff1-48ec5b586db5/versions/46911e8e-
f9bc-4898-8b63-59c7653636ec",
        "Name": "smp-ggrass-group"
    }
]
}

```

- Para API obtener más información, consulte [ListGroups](#) la Referencia de AWS CLI comandos.

## list-logger-definition-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-logger-definition-versions`.

### AWS CLI

Para obtener una lista de las versiones de una definición de registrador

El siguiente `list-logger-definition-versions` ejemplo obtiene una lista de todas las versiones de la definición de registrador especificada.

```

aws greengrass list-logger-definition-versions \
  --logger-definition-id "49eeeb66-f1d3-4e34-86e3-3617262abf23"

```

Salida:

```

{
  "Versions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/versions/5e3f6f64-
a565-491e-8de0-3c0d8e0f2073",
      "CreationTimestamp": "2019-05-08T16:10:13.866Z",
      "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
      "Version": "5e3f6f64-a565-491e-8de0-3c0d8e0f2073"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/versions/3ec6d3af-eb85-48f9-
a16d-1c795fe696d7",
      "CreationTimestamp": "2019-05-08T16:10:13.809Z",

```

```

        "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
        "Version": "3ec6d3af-eb85-48f9-a16d-1c795fe696d7"
    }
]
}

```

- Para API obtener más información, consulte [ListLoggerDefinitionVersions](#) la Referencia de AWS CLI comandos.

## list-logger-definitions

En el siguiente ejemplo de código, se muestra cómo usar `list-logger-definitions`.

### AWS CLI

Para obtener una lista de las definiciones de los registradores

En el siguiente `list-logger-definitions` ejemplo, se enumeran todas las definiciones de registrador de su AWS cuenta.

```
aws greengrass list-logger-definitions
```

Salida:

```

{
  "Definitions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23",
      "CreationTimestamp": "2019-05-08T16:10:13.809Z",
      "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
      "LastUpdatedTimestamp": "2019-05-08T16:10:13.809Z",
      "LatestVersion": "5e3f6f64-a565-491e-8de0-3c0d8e0f2073",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/
versions/5e3f6f64-a565-491e-8de0-3c0d8e0f2073"
    }
  ]
}

```

- Para API obtener más información, consulte [ListLoggerDefinitions](#) la Referencia de AWS CLI comandos.

## list-resource-definition-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-resource-definition-versions`.

### AWS CLI

Para enumerar las versiones de una definición de recurso

El siguiente `list-resource-definition-versions` ejemplo muestra las versiones del recurso de Greengrass especificado.

```
aws greengrass list-resource-definition-versions \  
--resource-definition-id "ad8c101d-8109-4b0e-b97d-9cc5802ab658"
```

Salida:

```
{  
  "Versions": [  
    {  
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/  
versions/26e8829a-491a-464d-9c87-664bf6f6f2be",  
      "CreationTimestamp": "2019-06-19T16:40:59.392Z",  
      "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",  
      "Version": "26e8829a-491a-464d-9c87-664bf6f6f2be"  
    },  
    {  
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/  
versions/432d92f6-12de-4ec9-a704-619a942a62aa",  
      "CreationTimestamp": "2019-06-19T16:40:59.261Z",  
      "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",  
      "Version": "432d92f6-12de-4ec9-a704-619a942a62aa"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListResourceDefinitionVersions](#) la Referencia de AWS CLI comandos.

## list-resource-definitions

En el siguiente ejemplo de código, se muestra cómo usar `list-resource-definitions`.

### AWS CLI

Para enumerar los recursos que están definidos

En el siguiente `list-resource-definitions` ejemplo, se enumeran los recursos que están definidos para que los utilice AWS IoT Greengrass.

```
aws greengrass list-resource-definitions
```

Salida:

```
{
  "Definitions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658",
      "CreationTimestamp": "2019-06-19T16:40:59.261Z",
      "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
      "LastUpdatedTimestamp": "2019-06-19T16:40:59.261Z",
      "LatestVersion": "26e8829a-491a-464d-9c87-664bf6f6f2be",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/
versions/26e8829a-491a-464d-9c87-664bf6f6f2be"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
      "CreationTimestamp": "2019-06-19T21:51:28.212Z",
      "Id": "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
      "LastUpdatedTimestamp": "2019-06-19T21:51:28.212Z",
      "LatestVersion": "a5f94d0b-f6bc-40f4-bb78-7a1c5fe13ba1",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38/versions/
a5f94d0b-f6bc-40f4-bb78-7a1c5fe13ba1",
      "Name": "MyGreengrassResources"
    }
  ]
}
```



- Para API obtener más información, consulte [ListResourceDefinitions](#) la Referencia de AWS CLI comandos.

## list-subscription-definition-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-subscription-definition-versions`.

### AWS CLI

Para enumerar las versiones de una definición de suscripción

En el siguiente `list-subscription-definition-versions` ejemplo, se enumeran todas las versiones de la suscripción especificada. Puede usar el `list-subscription-definitions` comando para buscar el identificador de suscripción.

```
aws greengrass list-subscription-definition-versions \
  --subscription-definition-id "70e49321-83d5-45d2-bc09-81f4917ae152"
```

Salida:

```
{
  "Versions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/versions/88ae8699-12ac-4663-ba3f-4d7f0519140b",
      "CreationTimestamp": "2019-06-18T17:03:52.499Z",
      "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
      "Version": "88ae8699-12ac-4663-ba3f-4d7f0519140b"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/versions/7e320ba3-c369-4069-a2f0-90acb7f219d6",
      "CreationTimestamp": "2019-06-18T17:03:52.392Z",
      "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
      "Version": "7e320ba3-c369-4069-a2f0-90acb7f219d6"
    }
  ]
}
```

- Para API obtener más información, consulte [ListSubscriptionDefinitionVersions](#) la Referencia de AWS CLI comandos.

## list-subscription-definitions

En el siguiente ejemplo de código, se muestra cómo usar `list-subscription-definitions`.

### AWS CLI

Para obtener una lista de definiciones de suscripciones

En el siguiente `list-subscription-definitions` ejemplo, se enumeran todas las suscripciones de AWS IoT Greengrass que están definidas en su AWS cuenta.

```
aws greengrass list-subscription-definitions
```

Salida:

```
{
  "Definitions": [
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152",
      "CreationTimestamp": "2019-06-18T17:03:52.392Z",
      "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
      "LastUpdatedTimestamp": "2019-06-18T17:03:52.392Z",
      "LatestVersion": "88ae8699-12ac-4663-ba3f-4d7f0519140b",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/
versions/88ae8699-12ac-4663-ba3f-4d7f0519140b"
    },
    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/subscriptions/cd6f1c37-d9a4-4e90-be94-01a7404f5967",
      "CreationTimestamp": "2018-10-18T15:45:34.024Z",
      "Id": "cd6f1c37-d9a4-4e90-be94-01a7404f5967",
      "LastUpdatedTimestamp": "2018-10-18T15:45:34.024Z",
      "LatestVersion": "d1cf8fac-284f-4f6a-98fe-a2d36d089373",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/cd6f1c37-d9a4-4e90-be94-01a7404f5967/versions/
d1cf8fac-284f-4f6a-98fe-a2d36d089373"
    },
  ],
}
```

```

    {
      "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/subscriptions/fa81bc84-3f59-4377-a84b-5d0134da359b",
      "CreationTimestamp": "2018-10-22T17:09:31.429Z",
      "Id": "fa81bc84-3f59-4377-a84b-5d0134da359b",
      "LastUpdatedTimestamp": "2018-10-22T17:09:31.429Z",
      "LatestVersion": "086d1b08-b25a-477c-a16f-6f9b3a9c295a",
      "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/fa81bc84-3f59-4377-a84b-5d0134da359b/
versions/086d1b08-b25a-477c-a16f-6f9b3a9c295a"
    }
  ]
}

```

- Para API obtener más información, consulte [ListSubscriptionDefinitions](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas adjuntas a un recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas y sus valores adjuntos al recurso especificado.

```

aws greengrass list-tags-for-resource \
  --resource-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658"

```

Salida:

```

{
  "tags": {
    "ResourceSubType": "USB",
    "ResourceType": "Device"
  }
}

```

Para obtener más información, consulte [Cómo etiquetar sus recursos de Greengrass en la Guía para desarrolladores de AWS IoT Greengrass](#).

- Para API obtener más información, consulte la Referencia de [ListTagsForResource](#) comandos AWS CLI .

## reset-deployments

En el siguiente ejemplo de código, se muestra cómo usar `reset-deployments`.

### AWS CLI

Para limpiar la información de despliegue de un grupo de Greengrass

El siguiente `reset-deployments` ejemplo limpia la información de despliegue del grupo de Greengrass especificado. Al agregar el `--force` option, la información de despliegue se restablece sin esperar a que el dispositivo principal responda.

```
aws greengrass reset-deployments \  
  --group-id "1402daf9-71cf-4cfe-8be0-d5e80526d0d8" \  
  --force
```

Salida:

```
{  
  "DeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/  
greengrass/groups/1402daf9-71cf-4cfe-8be0-d5e80526d0d8/  
deployments/7dd4e356-9882-46a3-9e28-6d21900c011a",  
  "DeploymentId": "7dd4e356-9882-46a3-9e28-6d21900c011a"  
}
```

Para obtener más información, consulte [Restablecer despliegues](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte la Referencia [ResetDeployments](#) de AWS CLI comandos.

## start-bulk-deployment

En el siguiente ejemplo de código, se muestra cómo usar `start-bulk-deployment`.

## AWS CLI

Para iniciar una operación de despliegue masivo

El siguiente `start-bulk-deployment` ejemplo inicia una operación de despliegue masivo mediante un archivo almacenado en un bucket de S3 para especificar los grupos que se van a implementar.

```
aws greengrass start-bulk-deployment \  
  --cli-input-json "{\"InputFileUri\":\"https://gg-group-deployment1.s3-us-  
west-2.amazonaws.com/MyBulkDeploymentInputFile.txt\", \"ExecutionRoleArn\":  
\"arn:aws:iam::123456789012:role/ggCreateDeploymentRole\", \"AmznClientToken\":  
\"yourAmazonClientToken\"}"
```

Salida:

```
{  
  "BulkDeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
bulk/deployments/870fb41b-6288-4e0c-bc76-a7ba4b4d3267",  
  "BulkDeploymentId": "870fb41b-6288-4e0c-bc76-a7ba4b4d3267"  
}
```

Para obtener más información, consulte [Crear despliegues masivos para grupos](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte la Referencia [StartBulkDeployment](#) de AWS CLI comandos.

## stop-bulk-deployment

En el siguiente ejemplo de código, se muestra cómo usar `stop-bulk-deployment`.

## AWS CLI

Para detener un despliegue masivo

El siguiente `stop-bulk-deployment` ejemplo detiene la implementación masiva especificada. Si intenta detener una implementación masiva que se ha completado, recibirá un error: `InvalidInputException: Cannot change state of finished execution.`

```
aws greengrass stop-bulk-deployment \  
  --cli-input-json "{\"BulkDeploymentId\":\"870fb41b-6288-4e0c-bc76-a7ba4b4d3267\"}"
```

```
--bulk-deployment-id "870fb41b-6288-4e0c-bc76-a7ba4b4d3267"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Crear despliegues masivos para grupos](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte la Referencia [StopBulkDeployment](#) de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para aplicar etiquetas a un recurso

El siguiente `tag-resource` ejemplo aplica dos etiquetas, `ResourceType` y `ResourceSubType`, al recurso de Greengrass especificado. Esta operación puede añadir etiquetas y valores nuevos o actualizar el valor de las etiquetas existentes. Utilice el `untag-resource` comando para eliminar una etiqueta.

```
aws greengrass tag-resource \  
  --resource-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658" \  
  --tags "ResourceType=Device,ResourceSubType=USB"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Cómo [etiquetar sus recursos de Greengrass en la Guía](#) para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte la Referencia de [TagResource](#) comandos AWS CLI .

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

## AWS CLI

Para eliminar una etiqueta y su valor de un recurso

En el siguiente `untag-resource` ejemplo, se quita la etiqueta cuya clave es `Category` del grupo Greengrass especificado. Si la clave `Category` no existe para el recurso especificado, no se devuelve ningún error.

```
aws greengrass untag-resource \  
  --resource-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/  
groups/1013db12-8b58-45ff-acc7-704248f66731" \  
  --tag-keys "Category"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Cómo [etiquetar sus recursos de Greengrass en la Guía para desarrolladores de AWS IoT Greengrass](#).

- Para API obtener más información, consulte la Referencia de [UntagResource](#) comandos AWS CLI .

## update-connectivity-info

En el siguiente ejemplo de código, se muestra cómo usar `update-connectivity-info`.

## AWS CLI

Para actualizar la información de conectividad de un núcleo de Greengrass

El siguiente `update-connectivity-info` ejemplo cambia los puntos finales que los dispositivos pueden usar para conectarse al núcleo de Greengrass especificado. La información de conectividad es una lista de direcciones IP o nombres de dominio, con los números de puerto correspondientes y los metadatos opcionales definidos por el cliente. Es posible que necesite actualizar la información de conectividad cuando cambie la red local.

```
aws greengrass update-connectivity-info \  
  --thing-name "MyGroup_Core" \  
  --connectivity-info "[{"Metadata":"","PortNumber":8883,"HostAddress":  
"127.0.0.1"},"Id":"localhost_127.0.0.1_0"}, {"Metadata":"","PortNumber  
":8883,"HostAddress":"192.168.1.3"},"Id":"localIP_192.168.1.3"}]"
```

Salida:

```
{
  "Version": "312de337-59af-4cf9-a278-2a23bd39c300"
}
```

- Para API obtener más información, consulte [UpdateConnectivityInfo](#) la Referencia de AWS CLI comandos.

## update-connector-definition

En el siguiente ejemplo de código, se muestra cómo usar `update-connector-definition`.

AWS CLI

Para actualizar el nombre de una definición de conector

En el siguiente `update-connector-definition` ejemplo, se actualiza el nombre de la definición de conector especificada. Si desea actualizar los detalles del conector, utilice el `create-connector-definition-version` comando para crear una nueva versión.

```
aws greengrass update-connector-definition \
  --connector-definition-id "55d0052b-0d7d-44d6-b56f-21867215e118" \
  --name "GreengrassConnectors2019"
```

Para obtener más información, consulte [Integración con servicios y protocolos mediante conectores](#) en la Guía para desarrolladores de AWS IoT Greengrass.

- Para API obtener más información, consulte [UpdateConnectorDefinition](#) la Referencia de AWS CLI comandos.

## update-core-definition

En el siguiente ejemplo de código, se muestra cómo usar `update-core-definition`.

AWS CLI

Para actualizar una definición básica

En el siguiente `update-core-definition` ejemplo, se cambia el nombre de la definición básica especificada. Solo puede actualizar la `name` propiedad de una definición básica.



```
aws greengrass update-core-definition \  
  --core-definition-id "582efe12-b05a-409e-9a24-a2ba1bcc4a12" \  
  --name "MyCoreDevices"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configurar el núcleo de AWS IoT Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [UpdateCoreDefinition](#) de AWS CLI comandos.

## update-device-definition

En el siguiente ejemplo de código, se muestra cómo usar `update-device-definition`.

### AWS CLI

Para actualizar una definición de dispositivo

En el siguiente `update-device-definition` ejemplo, se cambia el nombre de la definición de dispositivo especificada. Solo puede actualizar la `name` propiedad de una definición de dispositivo.

```
aws greengrass update-device-definition \  
  --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd" \  
  --name "TemperatureSensors"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UpdateDeviceDefinition](#) la Referencia de AWS CLI comandos.

## update-function-definition

En el siguiente ejemplo de código, se muestra cómo usar `update-function-definition`.

### AWS CLI

Para actualizar el nombre de una definición de función

En el siguiente `update-function-definition` ejemplo, se actualiza el nombre de la definición de función especificada. Si desea actualizar los detalles de la función, utilice el `create-function-definition-version` comando para crear una nueva versión.

```
aws greengrass update-function-definition \  
  --function-definition-id "e47952bd-dea9-4e2c-a7e1-37bbe8807f46" \  
  --name ObsoleteFunction
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Ejecutar funciones Lambda locales](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [UpdateFunctionDefinition](#) de AWS CLI comandos.

## update-group-certificate-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-group-certificate-configuration`.

### AWS CLI

Para actualizar la caducidad de los certificados de un grupo

En el siguiente `update-group-certificate-configuration` ejemplo, se establece un plazo de 10 días para los certificados generados para el grupo especificado.

```
aws greengrass update-group-certificate-configuration \  
  --group-id "8eaadd72-ce4b-4f15-892a-0cc4f3a343f1" \  
  --certificate-expiry-in-milliseconds 864000000
```

Salida:

```
{  
  "CertificateExpiryInMilliseconds": 864000000,  
  "CertificateAuthorityExpiryInMilliseconds": 2524607999000,  
  "GroupId": "8eaadd72-ce4b-4f15-892a-0cc4f3a343f1"  
}
```

Para obtener más información, consulte [Seguridad de AWS IoT Greengrass en la Guía para desarrolladores de IoT AWS Greengrass](#).

- Para API obtener más información, consulte la Referencia [UpdateGroupCertificateConfiguration](#) de AWS CLI comandos.

## update-group

En el siguiente ejemplo de código, se muestra cómo usar `update-group`.

### AWS CLI

Para actualizar el nombre del grupo

El siguiente `update-group` ejemplo actualiza el nombre del grupo Greengrass especificado. Si desea actualizar los detalles del grupo, utilice el `create-group-version` comando para crear una nueva versión.

```
aws greengrass update-group \  
  --group-id "1402daf9-71cf-4cfe-8be0-d5e80526d0d8" \  
  --name TestGroup4of6
```

Para obtener más información, consulte [Configurar AWS IoT Greengrass en IoT en la AWS Guía para desarrolladores de IoT AWS Greengrass](#).

- Para API obtener más información, consulte la Referencia [UpdateGroup](#) de AWS CLI comandos.

## update-logger-definition

En el siguiente ejemplo de código, se muestra cómo usar `update-logger-definition`.

### AWS CLI

Para actualizar la definición de un registrador

El siguiente `update-logger-definition` ejemplo cambia el nombre de la definición de registrador especificada. Solo puede actualizar la `name` propiedad de una definición de registrador.

```
aws greengrass update-logger-definition \  
  --name TestGroup4of6
```

```
--logger-definition-id "a454b62a-5d56-4ca9-bdc4-8254e1662cb0" \  
--name "LoggingConfigsForSensors"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Supervisión con registros de AWS IoT Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [UpdateLoggerDefinition](#) de AWS CLI comandos.

## update-resource-definition

En el siguiente ejemplo de código, se muestra cómo usar `update-resource-definition`.

### AWS CLI

Para actualizar el nombre de una definición de recurso

En el siguiente `update-resource-definition` ejemplo, se actualiza el nombre de la definición de recurso especificada. Si desea cambiar los detalles del recurso, utilice el `create-resource-definition-version` comando para crear una nueva versión.

```
aws greengrass update-resource-definition \  
--resource-definition-id "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38" \  
--name GreengrassConnectorResources
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Acceder a los recursos locales con funciones y conectores de Lambda](#) en la Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia [UpdateResourceDefinition](#) de AWS CLI comandos.

## update-subscription-definition

En el siguiente ejemplo de código, se muestra cómo usar `update-subscription-definition`.

### AWS CLI

Para actualizar el nombre de una definición de suscripción

En el siguiente `update-subscription-definition` ejemplo, se actualiza el nombre de la definición de suscripción especificada. Si desea cambiar los detalles de la suscripción, utilice el `create-subscription-definition-version` comando para crear una nueva versión.

```
aws greengrass update-subscription-definition \  
  --subscription-definition-id "fa81bc84-3f59-4377-a84b-5d0134da359b" \  
  --name "ObsoleteSubscription"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte el título de la guía.

- Para API obtener más información, consulte [UpdateSubscriptionDefinition](#) la Referencia de AWS CLI comandos.

## update-thing-runtime-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-thing-runtime-configuration`.

### AWS CLI

Para activar la telemetría en la configuración de tiempo de ejecución de un núcleo de Greengrass

El siguiente `update-thing-runtime-configuration` ejemplo actualiza la configuración de tiempo de ejecución de un núcleo de Greengrass para activar la telemetría.

```
aws greengrass update-thing-runtime-configuration \  
  --thing-name SampleGreengrassCore \  
  --telemetry-configuration {"Telemetry":\\"On\\"}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de los ajustes de telemetría en la](#) Guía para desarrolladores de IoT AWS Greengrass.

- Para API obtener más información, consulte la Referencia de [UpdateThingRuntimeConfiguration](#) comandos AWS CLI .

## AWS IoT Greengrass V2 ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS IoT Greengrass V2.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **associate-service-role-to-account**

En el siguiente ejemplo de código, se muestra cómo usar `associate-service-role-to-account`.

#### AWS CLI

Para asociar el rol de servicio de Greengrass a su cuenta AWS

El siguiente `associate-service-role-to-account` ejemplo asocia un rol de servicio a AWS IoT Greengrass para su AWS cuenta.

```
aws greengrassv2 associate-service-role-to-account \  
  --role-arn arn:aws:iam::123456789012:role/service-role/Greengrass_ServiceRole
```

Salida:

```
{  
  "associatedAt": "2022-01-19T19:21:53Z"  
}
```

Para obtener más información, consulte la [función de servicio de Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass V2.

- Para API obtener más información, consulte la Referencia [AssociateServiceRoleToAccount](#) de AWS CLI comandos.

## batch-associate-client-device-with-core-device

En el siguiente ejemplo de código, se muestra cómo usar `batch-associate-client-device-with-core-device`.

### AWS CLI

Para asociar dispositivos cliente a un dispositivo principal

El siguiente `batch-associate-client-device-with-core-device` ejemplo asocia dos dispositivos cliente a un dispositivo principal.

```
aws greengrassv2 batch-associate-client-device-with-core-device \
  --core-device-thing-name MyGreengrassCore \
  --entries thingName=MyClientDevice1 thingName=MyClientDevice2
```

Salida:

```
{
  "errorEntries": []
}
```

Para obtener más información, consulte [Interactuar con dispositivos de IoT locales](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [BatchAssociateClientDeviceWithCoreDevice](#) la Referencia de AWS CLI comandos.

## batch-disassociate-client-device-from-core-device

En el siguiente ejemplo de código, se muestra cómo usar `batch-disassociate-client-device-from-core-device`.

### AWS CLI

Para desasociar los dispositivos cliente de un dispositivo principal

El siguiente `batch-disassociate-client-device-from-core-device` ejemplo disocia dos dispositivos cliente de un dispositivo principal.

```
aws greengrassv2 batch-disassociate-client-device-from-core-device \  
  --core-device-thing-name MyGreengrassCore \  
  --entries thingName=MyClientDevice1 thingName=MyClientDevice2
```

Salida:

```
{  
  "errorEntries": []  
}
```

Para obtener más información, consulte [Interactuar con dispositivos de IoT locales](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [BatchDisassociateClientDeviceFromCoreDevice](#) la Referencia de AWS CLI comandos.

## cancel-deployment

En el siguiente ejemplo de código, se muestra cómo usar `cancel-deployment`.

### AWS CLI

Para cancelar un despliegue

El siguiente `cancel-deployment` ejemplo detiene un despliegue continuo en un grupo de cosas.

```
aws greengrassv2 cancel-deployment \  
  --deployment-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "message": "SUCCESS"  
}
```

Para obtener más información, consulte [Cancelar despliegues](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.



- Para API obtener más información, consulte la Referencia [CancelDeployment](#) de AWS CLI comandos.

## create-component-version

En el siguiente ejemplo de código, se muestra cómo usar `create-component-version`.

### AWS CLI

Ejemplo 1: Para crear una versión de un componente a partir de una receta

El siguiente `create-component-version` ejemplo crea una versión de un componente de Hello World a partir de un archivo de recetas.

```
aws greengrassv2 create-component-version \  
  --inline-recipe fileb://com.example.HelloWorld-1.0.0.json
```

Contenidos de `com.example.HelloWorld-1.0.0.json`:

```
{  
  "RecipeFormatVersion": "2020-01-25",  
  "ComponentName": "com.example.HelloWorld",  
  "ComponentVersion": "1.0.0",  
  "ComponentDescription": "My first AWS IoT Greengrass component.",  
  "ComponentPublisher": "Amazon",  
  "ComponentConfiguration": {  
    "DefaultConfiguration": {  
      "Message": "world"  
    }  
  },  
  "Manifests": [  
    {  
      "Platform": {  
        "os": "linux"  
      },  
      "Lifecycle": {  
        "Run": "echo 'Hello {configuration:/Message}'"  
      }  
    }  
  ]  
}
```

**Salida:**

```
{
  "arn": "arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0",
  "componentName": "com.example.HelloWorld",
  "componentVersion": "1.0.0",
  "creationTimestamp": "2021-01-07T16:24:33.650000-08:00",
  "status": {
    "componentState": "REQUESTED",
    "message": "NONE",
    "errors": {}
  }
}
```

Para obtener más información, consulte [Crear componentes personalizados](#) y [Cargar componentes para desplegarlos](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

Ejemplo 2: Para crear una versión de un componente a partir de una AWS función Lambda

El siguiente `create-component-version` ejemplo crea una versión de un componente Hello World a partir de una AWS función Lambda.

```
aws greengrassv2 create-component-version \
  --cli-input-json file://lambda-function-component.json
```

Contenidos de `lambda-function-component.json`:

```
{
  "lambdaFunction": {
    "lambdaArn": "arn:aws:lambda:us-
west-2:123456789012:function:HelloWorldPythonLambda:1",
    "componentName": "com.example.HelloWorld",
    "componentVersion": "1.0.0",
    "componentLambdaParameters": {
      "eventSources": [
        {
          "topic": "hello/world/+",
          "type": "IOT_CORE"
        }
      ]
    }
  }
}
```

```
}  
}
```

Salida:

```
{  
  "arn": "arn:aws:greengrass:us-  
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0",  
  "componentName": "com.example.HelloWorld",  
  "componentVersion": "1.0.0",  
  "creationTimestamp": "2021-01-07T17:05:27.347000-08:00",  
  "status": {  
    "componentState": "REQUESTED",  
    "message": "NONE",  
    "errors": {}  
  }  
}
```

Para obtener más información, consulte [Ejecutar funciones AWS Lambda](#) en la Guía para desarrolladores de IoT AWS Greengrass V2.

- Para API obtener más información, consulte la Referencia [CreateComponentVersion](#) de AWS CLI comandos.

## create-deployment

En el siguiente ejemplo de código, se muestra cómo usar create-deployment.

### AWS CLI

Ejemplo 1: Para crear un despliegue

El siguiente create-deployment ejemplo implementa la interfaz de línea de comandos de AWS IoT Greengrass en un dispositivo central.

```
aws greengrassv2 create-deployment \  
  --cli-input-json file://cli-deployment.json
```

Contenidos de cli-deployment.json:

```
{
```

```
"targetArn": "arn:aws:iot:us-west-2:123456789012:thing/MyGreengrassCore",
"deploymentName": "Deployment for MyGreengrassCore",
"components": {
  "aws.greengrass.Cli": {
    "componentVersion": "2.0.3"
  }
},
"deploymentPolicies": {
  "failureHandlingPolicy": "DO_NOTHING",
  "componentUpdatePolicy": {
    "timeoutInSeconds": 60,
    "action": "NOTIFY_COMPONENTS"
  },
  "configurationValidationPolicy": {
    "timeoutInSeconds": 60
  }
},
"iotJobConfiguration": {}
}
```

Salida:

```
{
  "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Para obtener más información, consulte [Crear despliegues](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

Ejemplo 2: Para crear una implementación que actualice las configuraciones de los componentes

En el siguiente `create-deployment` ejemplo, se implementa el componente núcleo de AWS IoT Greengrass en un grupo de dispositivos principales. Esta implementación aplica las siguientes actualizaciones de configuración para el componente núcleo:

Restablezca la configuración de proxy de los dispositivos de destino a su configuración predeterminada sin proxy. Restablezca la configuración de los dispositivos de destino a sus valores predeterminados. MQTT Establece las JVM opciones del núcleo. Establece el nivel de registro del núcleo. JVM

```
aws greengrassv2 create-deployment \
```

```
--cli-input-json file://nucleus-deployment.json
```

Contenidos de nucleus-deployment.json:

```
{
  "targetArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/
MyGreengrassCoreGroup",
  "deploymentName": "Deployment for MyGreengrassCoreGroup",
  "components": {
    "aws.greengrass.Nucleus": {
      "componentVersion": "2.0.3",
      "configurationUpdate": {
        "reset": [
          "/networkProxy",
          "/mqtt"
        ],
        "merge": "{\"jvmOptions\":\"-Xmx64m\",\"logging\":{\"level\":\"WARN
\"}}"}
      }
    }
  },
  "deploymentPolicies": {
    "failureHandlingPolicy": "ROLLBACK",
    "componentUpdatePolicy": {
      "timeoutInSeconds": 60,
      "action": "NOTIFY_COMPONENTS"
    },
    "configurationValidationPolicy": {
      "timeoutInSeconds": 60
    }
  },
  "iotJobConfiguration": {}
}
```

Salida:

```
{
  "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "iotJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "iotJobArn": "arn:aws:iot:us-west-2:123456789012:job/a1b2c3d4-5678-90ab-cdef-
EXAMPLE22222"
}
```

Para obtener más información, consulte [Crear despliegues](#) y [actualizar configuraciones de componentes](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte la Referencia [CreateDeployment](#) de AWS CLI comandos.

## delete-component

En el siguiente ejemplo de código, se muestra cómo usar delete-component.

### AWS CLI

Para eliminar una versión de un componente

En el siguiente delete-component ejemplo, se elimina un componente de Hello World.

```
aws greengrassv2 delete-component \  
  --arn arn:aws:greengrass:us-west-2:123456789012:components:com.example>HelloWorld:versions:1.0.0
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar componentes](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [DeleteComponent](#) la Referencia de AWS CLI comandos.

## delete-core-device

En el siguiente ejemplo de código, se muestra cómo usar delete-core-device.

### AWS CLI

Para eliminar un dispositivo principal

El siguiente delete-core-device ejemplo elimina un dispositivo principal de AWS IoT Greengrass.

```
aws greengrassv2 delete-core-device \  
  --core-device-thing-name MyGreengrassCore
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Desinstalar el software AWS IoT Greengrass Core](#) en la Guía para desarrolladores de IoT AWS Greengrass V2.

- Para API obtener más información, consulte la Referencia [DeleteCoreDevice](#) de AWS CLI comandos.

## describe-component

En el siguiente ejemplo de código, se muestra cómo usar `describe-component`.

### AWS CLI

Para describir la versión de un componente

El siguiente `describe-component` ejemplo describe un componente de Hello World.

```
aws greengrassv2 describe-component \  
  --arn arn:aws:greengrass:us-  
west-2:123456789012:components:com.example>HelloWorld:versions:1.0.0
```

Salida:

```
{  
  "arn": "arn:aws:greengrass:us-  
west-2:123456789012:components:com.example>HelloWorld:versions:1.0.0",  
  "componentName": "com.example>HelloWorld",  
  "componentVersion": "1.0.0",  
  "creationTimestamp": "2021-01-07T17:12:11.133000-08:00",  
  "publisher": "Amazon",  
  "description": "My first AWS IoT Greengrass component.",  
  "status": {  
    "componentState": "DEPLOYABLE",  
    "message": "NONE",  
    "errors": {}  
  },  
  "platforms": [  
    {  
      "attributes": {  
        "os": "linux"  
      }  
    }  
  ]  
}
```

```
]
}
```

Para obtener más información, consulte [Administrar componentes](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [DescribeComponent](#) la Referencia de AWS CLI comandos.

## **disassociate-service-role-from-account**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-service-role-from-account`.

### AWS CLI

Para desasociar el rol de servicio de Greengrass de su cuenta AWS

El siguiente `disassociate-service-role-from-account` ejemplo desvincula la función de servicio Greengrass de AWS IoT Greengrass para su cuenta. AWS

```
aws greengrassv2 disassociate-service-role-from-account
```

Salida:

```
{
  "disassociatedAt": "2022-01-19T19:26:09Z"
}
```

Para obtener más información, consulte la [función de servicio de Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass V2.

- Para API obtener más información, consulte la Referencia [DisassociateServiceRoleFromAccount](#) de AWS CLI comandos.

## **get-component-version-artifact**

En el siguiente ejemplo de código, se muestra cómo usar `get-component-version-artifact`.

### AWS CLI

Para URL descargar un artefacto componente



El siguiente `get-component-version-artifact` ejemplo permite descargar el URL archivo del componente de la consola de depuración local. JAR

```
aws greengrassv2 get-component-version-artifact \
  --arn arn:aws:greengrass:us-west-2:aws:components:aws.greengrass.LocalDebugConsole:versions:2.0.3 \
  --artifact-name "Uvt6ZEzQ9TKiAuLbfXBX_APdY0TWks3uc46tHFHTzBM=/aws.greengrass.LocalDebugConsole.jar"
```

Salida:

```
{
  "preSignedUrl": "https://evergreencomponentmanageme-
  artifactbucket7410c9ef-g18n1iya8kwr.s3.us-west-2.amazonaws.com/public/
  aws.greengrass.LocalDebugConsole/2.0.3/s3/ggv2-component-releases-prod-pdx/
  EvergreenHttpDebugView/2ffc496ba41b39568968b22c582b4714a937193ee7687a45527238e696672521/
  aws.greengrass.LocalDebugConsole/aws.greengrass.LocalDebugConsole.jar?X-Amz-
  Security-Token=KwFLKSdEXAMPLE..."
}
```

Para obtener más información, consulte [Administrar componentes](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [GetComponentVersionArtifact](#) la Referencia de AWS CLI comandos.

## get-component

En el siguiente ejemplo de código, se muestra cómo usar `get-component`.

### AWS CLI

Ejemplo 1: Para descargar la receta de un componente en YAML formato (Linux, macOS o Unix)

En el siguiente `get-component` ejemplo, se descarga la receta de un componente de Hello World a un archivo en YAML formato. Este comando hace lo siguiente:

Utiliza los `--query` parámetros `--output` y para controlar el resultado del comando. Estos parámetros extraen el blob de recetas de la salida del comando. Para obtener más información sobre el control de la salida, consulte [Control de la salida de comandos](#) en la Guía del usuario de la interfaz de línea de AWS comandos. Utiliza la base64 utilidad. Esta utilidad decodifica

el blob extraído para convertirlo en el texto original. El blob que devuelve un `get-component` comando correcto es texto codificado en base64. Debe decodificar este blob para obtener el texto original. Guarda el texto decodificado en un archivo. La última sección del comando (`> com.example.HelloWorld-1.0.0.json`) guarda el texto decodificado en un archivo.

```
aws greengrassv2 get-component \
  --arn arn:aws:greengrass:us-west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0 \
  --recipe-output-format YAML \
  --query recipe \
  --output text | base64 --decode > com.example.HelloWorld-1.0.0.json
```

Para obtener más información, consulte [Administrar componentes](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

Ejemplo 2: Para descargar la receta de un componente en YAML formato (WindowsCMD)

En el siguiente `get-component` ejemplo, se descarga la receta de un componente de Hello World a un archivo en YAML formato. Este comando usa la `certutil` utilidad.

```
aws greengrassv2 get-component ^
  --arn arn:aws:greengrass:us-west-2:675946970638:components:com.example.HelloWorld:versions:1.0.0 ^
  --recipe-output-format YAML ^
  --query recipe ^
  --output text > com.example.HelloWorld-1.0.0.yaml.b64

certutil -
decode com.example.HelloWorld-1.0.0.yaml.b64 com.example.HelloWorld-1.0.0.yaml
```

Para obtener más información, consulte [Administrar componentes](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

Ejemplo 3: Para descargar la receta de un componente en YAML formato (Windows PowerShell)

En el siguiente `get-component` ejemplo, se descarga la receta de un componente de Hello World a un archivo en YAML formato. Este comando usa la `certutil` utilidad.

```
aws greengrassv2 get-component `
  --arn arn:aws:greengrass:us-west-2:675946970638:components:com.example.HelloWorld:versions:1.0.0 `
  --recipe-output-format YAML `
```

```
--query recipe `  
--output text > com.example.HelloWorld-1.0.0.yaml.b64
```

```
certutil -  
decode com.example.HelloWorld-1.0.0.yaml.b64 com.example.HelloWorld-1.0.0.yaml
```

Para obtener más información, consulte [Administrar componentes](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [GetComponentent](#)la Referencia de AWS CLI comandos.

## get-connectivity-info

En el siguiente ejemplo de código, se muestra cómo usar `get-connectivity-info`.

### AWS CLI

Para obtener la información de conectividad de un dispositivo principal de Greengrass

El siguiente `get-connectivity-info` ejemplo obtiene la información de conectividad de un dispositivo principal de Greengrass. Los dispositivos cliente utilizan esta información para conectarse al MQTT intermediario que se ejecuta en este dispositivo principal.

```
aws greengrassv2 get-connectivity-info \  
--thing-name MyGreengrassCore
```

Salida:

```
{  
  "connectivityInfo": [  
    {  
      "id": "localIP_192.0.2.0",  
      "hostAddress": "192.0.2.0",  
      "portNumber": 8883  
    }  
  ]  
}
```

Para obtener más información, consulte [Administrar los puntos finales de los dispositivos principales](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte la Referencia [GetConnectivityInfo](#) de AWS CLI comandos.

## get-core-device

En el siguiente ejemplo de código, se muestra cómo usar `get-core-device`.

### AWS CLI

Para obtener un dispositivo básico

En el siguiente `get-core-device` ejemplo, se obtiene información sobre un AWS dispositivo principal de IoT Greengrass.

```
aws greengrassv2 get-core-device \  
  --core-device-thing-name MyGreengrassCore
```

Salida:

```
{  
  "coreDeviceThingName": "MyGreengrassCore",  
  "coreVersion": "2.0.3",  
  "platform": "linux",  
  "architecture": "amd64",  
  "status": "HEALTHY",  
  "lastStatusUpdateTimestamp": "2021-01-08T04:57:58.838000-08:00",  
  "tags": {}  
}
```

Para obtener más información, [consulte Comprobar el estado del dispositivo principal](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [GetCoreDevice](#) la Referencia de AWS CLI comandos.

## get-deployment

En el siguiente ejemplo de código, se muestra cómo usar `get-deployment`.

### AWS CLI

Para obtener un despliegue

En el siguiente `get-deployment` ejemplo, se obtiene información sobre la implementación del componente núcleo de AWS IoT Greengrass en un grupo de dispositivos principales.

```
aws greengrassv2 get-deployment \  
  --deployment-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "targetArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/  
MyGreengrassCoreGroup",  
  "revisionId": "14",  
  "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "deploymentName": "Deployment for MyGreengrassCoreGroup",  
  "deploymentStatus": "ACTIVE",  
  "iotJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
  "iotJobArn": "arn:aws:iot:us-west-2:123456789012:job/a1b2c3d4-5678-90ab-cdef-  
EXAMPLE22222",  
  "components": {  
    "aws.greengrass.Nucleus": {  
      "componentVersion": "2.0.3",  
      "configurationUpdate": {  
        "merge": "{\"jvmOptions\": \"-Xmx64m\", \"logging\": {\"level\": \"WARN  
\\\"}}\",  
        "reset": [  
          "/networkProxy",  
          "/mqtt"  
        ]  
      }  
    }  
  },  
  "deploymentPolicies": {  
    "failureHandlingPolicy": "ROLLBACK",  
    "componentUpdatePolicy": {  
      "timeoutInSeconds": 60,  
      "action": "NOTIFY_COMPONENTS"  
    },  
    "configurationValidationPolicy": {  
      "timeoutInSeconds": 60  
    }  
  },  
  "iotJobConfiguration": {},  
  "creationTimestamp": "2021-01-07T17:21:20.691000-08:00",
```

```
"isLatestForTarget": false,  
"tags": {}  
}
```

Para obtener más información, consulte [Implementar componentes en dispositivos](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [GetDeployment](#) la Referencia de AWS CLI comandos.

## **get-service-role-for-account**

En el siguiente ejemplo de código, se muestra cómo usar `get-service-role-for-account`.

### AWS CLI

Para obtener el rol de servicio de Greengrass para su cuenta AWS

En el siguiente `get-service-role-for-account` ejemplo, se obtiene el rol de servicio asociado a AWS IoT Greengrass para su AWS cuenta.

```
aws greengrassv2 get-service-role-for-account
```

Salida:

```
{  
  "associatedAt": "2022-01-19T19:21:53Z",  
  "roleArn": "arn:aws:iam::123456789012:role/service-role/Greengrass_ServiceRole"  
}
```

Para obtener más información, consulte la [función de servicio de Greengrass](#) en la Guía para desarrolladores de IoT AWS Greengrass V2.

- Para API obtener más información, consulte la Referencia [GetServiceRoleForAccount](#) de AWS CLI comandos.

## **list-client-devices-associated-with-core-device**

En el siguiente ejemplo de código, se muestra cómo usar `list-client-devices-associated-with-core-device`.

## AWS CLI

Para ver una lista de los dispositivos cliente asociados a un dispositivo principal

En el siguiente `list-client-devices-associated-with-core-device` ejemplo, se enumeran todos los dispositivos cliente asociados a un dispositivo principal.

```
aws greengrassv2 list-client-devices-associated-with-core-device \
  --core-device-thing-name MyTestGreengrassCore
```

Salida:

```
{
  "associatedClientDevices": [
    {
      "thingName": "MyClientDevice2",
      "associationTimestamp": "2021-07-12T16:33:55.843000-07:00"
    },
    {
      "thingName": "MyClientDevice1",
      "associationTimestamp": "2021-07-12T16:33:55.843000-07:00"
    }
  ]
}
```

Para obtener más información, consulte [Interactuar con dispositivos de IoT locales](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [ListClientDevicesAssociatedWithCoreDevice](#) la Referencia de AWS CLI comandos.

## **list-component-versions**

En el siguiente ejemplo de código, se muestra cómo usar `list-component-versions`.

## AWS CLI

Para enumerar las versiones de un componente

En el siguiente `list-component-versions` ejemplo, se enumeran todas las versiones de un componente de Hello World.

```
aws greengrassv2 list-component-versions \  
  --arn arn:aws:greengrass:us-  
west-2:123456789012:components:com.example.HelloWorld
```

Salida:

```
{  
  "componentVersions": [  
    {  
      "componentName": "com.example.HelloWorld",  
      "componentVersion": "1.0.1",  
      "arn": "arn:aws:greengrass:us-  
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.1"  
    },  
    {  
      "componentName": "com.example.HelloWorld",  
      "componentVersion": "1.0.0",  
      "arn": "arn:aws:greengrass:us-  
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0"  
    }  
  ]  
}
```

Para obtener más información, consulte [Administrar componentes](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [ListComponentVersions](#) la Referencia de AWS CLI comandos.

## list-components

En el siguiente ejemplo de código, se muestra cómo usar `list-components`.

AWS CLI

Para enumerar los componentes

El siguiente `list-components` ejemplo muestra cada componente y su última versión definida en su AWS cuenta en la región actual.

```
aws greengrassv2 list-components
```



**Salida:**

```
{
  "components": [
    {
      "arn": "arn:aws:greengrass:us-west-2:123456789012:components:com.example.HelloWorld",
      "componentName": "com.example.HelloWorld",
      "latestVersion": {
        "arn": "arn:aws:greengrass:us-west-2:123456789012:components:com.example.HelloWorld:versions:1.0.1",
        "componentVersion": "1.0.1",
        "creationTimestamp": "2021-01-08T16:51:07.352000-08:00",
        "description": "My first AWS IoT Greengrass component.",
        "publisher": "Amazon",
        "platforms": [
          {
            "attributes": {
              "os": "linux"
            }
          }
        ]
      }
    }
  ]
}
```

Para obtener más información, consulte [Administrar componentes](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [ListComponents](#) la Referencia de AWS CLI comandos.

**list-core-devices**

En el siguiente ejemplo de código, se muestra cómo usar `list-core-devices`.

**AWS CLI**

Para enumerar los dispositivos principales

El siguiente `list-core-devices` ejemplo muestra los dispositivos principales de AWS IoT Greengrass de tu AWS cuenta en la región actual.

```
aws greengrassv2 list-core-devices
```

Salida:

```
{
  "coreDevices": [
    {
      "coreDeviceThingName": "MyGreengrassCore",
      "status": "HEALTHY",
      "lastStatusUpdateTimestamp": "2021-01-08T04:57:58.838000-08:00"
    }
  ]
}
```

Para obtener más información, [consulte Comprobar el estado del dispositivo principal](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [ListCoreDevices](#) la Referencia de AWS CLI comandos.

## list-deployments

En el siguiente ejemplo de código, se muestra cómo usar `list-deployments`.

AWS CLI

Para enumerar las implementaciones

En el siguiente `list-deployments` ejemplo, se muestra la última revisión de cada despliegue definido en su AWS cuenta en la región actual.

```
aws greengrassv2 list-deployments
```

Salida:

```
{
  "deployments": [
    {
      "targetArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/MyGreengrassCoreGroup",
      "revisionId": "14",
    }
  ]
}
```

```

    "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "deploymentName": "Deployment for MyGreengrassCoreGroup",
    "creationTimestamp": "2021-01-07T17:21:20.691000-08:00",
    "deploymentStatus": "ACTIVE",
    "isLatestForTarget": false
  },
  {
    "targetArn": "arn:aws:iot:us-west-2:123456789012:thing/
MyGreengrassCore",
    "revisionId": "1",
    "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "deploymentName": "Deployment for MyGreengrassCore",
    "creationTimestamp": "2021-01-06T16:10:42.407000-08:00",
    "deploymentStatus": "COMPLETED",
    "isLatestForTarget": false
  }
]
}

```

Para obtener más información, consulte [Implementar componentes en dispositivos](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [ListDeployments](#) la Referencia de AWS CLI comandos.

## list-effective-deployments

En el siguiente ejemplo de código, se muestra cómo usar `list-effective-deployments`.

### AWS CLI

Para enumerar los trabajos de despliegue

En el siguiente `list-effective-deployments` ejemplo, se enumeran las implementaciones que se aplican a un dispositivo principal de AWS IoT Greengrass.

```

aws greengrassv2 list-effective-deployments \
  --core-device-thing-name MyGreengrassCore

```

Salida:

```

{
  "effectiveDeployments": [

```

```

    {
      "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "deploymentName": "Deployment for MyGreengrassCore",
      "iotJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
      "targetArn": "arn:aws:iot:us-west-2:123456789012:thing/
MyGreengrassCore",
      "coreDeviceExecutionStatus": "COMPLETED",
      "reason": "SUCCESSFUL",
      "creationTimestamp": "2021-01-06T16:10:42.442000-08:00",
      "modifiedTimestamp": "2021-01-08T17:21:27.830000-08:00"
    },
    {
      "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "deploymentName": "Deployment for MyGreengrassCoreGroup",
      "iotJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
      "iotJobArn": "arn:aws:iot:us-west-2:123456789012:job/a1b2c3d4-5678-90ab-
cdef-EXAMPLE44444",
      "targetArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/
MyGreengrassCoreGroup",
      "coreDeviceExecutionStatus": "SUCCEEDED",
      "reason": "SUCCESSFUL",
      "creationTimestamp": "2021-01-07T17:19:20.394000-08:00",
      "modifiedTimestamp": "2021-01-07T17:21:20.721000-08:00"
    }
  ]
}

```

Para obtener más información, [consulte Comprobar el estado del dispositivo principal](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [ListEffectiveDeployments](#) la Referencia de AWS CLI comandos.

## list-installed-components

En el siguiente ejemplo de código, se muestra cómo usar `list-installed-components`.

### AWS CLI

Para ver una lista de los componentes instalados en un dispositivo principal

En el siguiente `list-installed-components` ejemplo, se enumeran los componentes que están instalados en un dispositivo principal de AWS IoT Greengrass.

```
aws greengrassv2 list-installed-components \  
  --core-device-thing-name MyGreengrassCore
```

Salida:

```
{  
  "installedComponents": [  
    {  
      "componentName": "aws.greengrass.Cli",  
      "componentVersion": "2.0.3",  
      "lifecycleState": "RUNNING",  
      "isRoot": true  
    },  
    {  
      "componentName": "aws.greengrass.Nucleus",  
      "componentVersion": "2.0.3",  
      "lifecycleState": "FINISHED",  
      "isRoot": true  
    }  
  ]  
}
```

Para obtener más información, [consulte Comprobar el estado del dispositivo principal](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [ListInstalledComponents](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

AWS CLI

Para enumerar las etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas de un AWS dispositivo principal de IoT Greengrass.

```
aws greengrassv2 list-tags-for-resource \  
  \
```

```
--resource-arn arn:aws:greengrass:us-west-2:123456789012:coreDevices:MyGreengrassCore
```

Salida:

```
{
  "tags": {
    "Owner": "richard-roe"
  }
}
```

Para obtener más información, consulte [Etiquete sus recursos](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Cómo añadir una etiqueta a un recurso

El siguiente `tag-resource` ejemplo agrega una etiqueta de propietario a un dispositivo principal de AWS IoT Greengrass. Puede usar esta etiqueta para controlar el acceso al dispositivo principal en función de quién es su propietario.

```
aws greengrassv2 tag-resource \
  --resource-arn arn:aws:greengrass:us-west-2:123456789012:coreDevices:MyGreengrassCore \
  --tags Owner=richard-roe
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquete sus recursos](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar una etiqueta de un recurso

El siguiente `untag-resource` ejemplo elimina una etiqueta de propietario de un dispositivo principal de AWS IoT Greengrass.

```
aws iotsitewise untag-resource \  
  --resource-arn arn:aws:greengrass:us-west-2:123456789012:coreDevices:MyGreengrassCore \  
  --tag-keys Owner
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquete sus recursos](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-connectivity-info

En el siguiente ejemplo de código, se muestra cómo usar `update-connectivity-info`.

### AWS CLI

Para actualizar la información de conectividad de un dispositivo principal de Greengrass

El siguiente `update-connectivity-info` ejemplo obtiene la información de conectividad de un dispositivo principal de Greengrass. Los dispositivos cliente utilizan esta información para conectarse al MQTT intermediario que se ejecuta en este dispositivo principal.

```
aws greengrassv2 update-connectivity-info \  
  --thing-name MyGreengrassCore \  
  --cli-input-json file://core-device-connectivity-info.json
```

Contenidos de `core-device-connectivity-info.json`:

```
{
  "connectivityInfo": [
    {
      "hostAddress": "192.0.2.0",
      "portNumber": 8883,
      "id": "localIP_192.0.2.0"
    }
  ]
}
```

Salida:

```
{
  "version": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Para obtener más información, consulte [Administrar los puntos finales de los dispositivos principales](#) en la Guía para desarrolladores de AWS IoT Greengrass V2.

- Para API obtener más información, consulte la Referencia [UpdateConnectivityInfo](#) de AWS CLI comandos.

## AWS IoT Jobs SDK release ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS IoT Jobs SDK release.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)



## Acciones

### **describe-job-execution**

En el siguiente ejemplo de código, se muestra cómo usar `describe-job-execution`.

#### AWS CLI

Para obtener los detalles de la ejecución de un trabajo

En el siguiente `describe-job-execution` ejemplo, se recuperan los detalles de la última ejecución de la tarea y el objeto especificados.

```
aws iot-jobs-data describe-job-execution \  
  --job-id SampleJob \  
  --thing-name MotionSensor1 \  
  --endpoint-url https://1234567890abcd.jobs.iot.us-west-2.amazonaws.com
```

Salida:

```
{  
  "execution": {  
    "approximateSecondsBeforeTimedOut": 88,  
    "executionNumber": 2939653338,  
    "jobId": "SampleJob",  
    "lastUpdatedAt": 1567701875.743,  
    "queuedAt": 1567701902.444,  
    "status": "QUEUED",  
    "thingName": "MotionSensor1 ",  
    "versionNumber": 3  
  }  
}
```

Para obtener más información, consulte [Dispositivos y trabajos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [DescribeJobExecution](#) la Referencia de AWS CLI comandos.

### **get-pending-job-executions**

En el siguiente ejemplo de código, se muestra cómo usar `get-pending-job-executions`.

## AWS CLI

Para obtener una lista de todos los trabajos que no están en estado terminal para una cosa

En el siguiente `get-pending-job-executions` ejemplo, se muestra una lista de todos los trabajos que no están en estado terminal para el elemento especificado.

```
aws iot-jobs-data get-pending-job-executions \  
  --thing-name MotionSensor1 \  
  --endpoint-url https://1234567890abcd.jobs.iot.us-west-2.amazonaws.com
```

Salida:

```
{  
  "InProgressJobs": [  
  ],  
  "queuedJobs": [  
    {  
      "executionNumber": 2939653338,  
      "jobId": "SampleJob",  
      "lastUpdatedAt": 1567701875.743,  
      "queuedAt": 1567701902.444,  
      "versionNumber": 3  
    }  
  ]  
}
```

Para obtener más información, consulte [Dispositivos y trabajos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetPendingJobExecutions](#) la Referencia de AWS CLI comandos.

## **start-next-pending-job-execution**

En el siguiente ejemplo de código, se muestra cómo usar `start-next-pending-job-execution`.

## AWS CLI

Para iniciar la siguiente ejecución de trabajo pendiente de una cosa

El siguiente `start-next-pending-job-execution` ejemplo recupera e inicia la siguiente ejecución de trabajo cuyo estado sea `IN_PROGRESS` o `QUEUED` para el elemento especificado.

```
aws iot-jobs-data start-next-pending-job-execution \  
  --thing-name MotionSensor1 \  
  --endpoint-url https://1234567890abcd.jobs.iot.us-west-2.amazonaws.com
```

Salida:

```
{  
  "execution": {  
    "approximateSecondsBeforeTimedOut": 88,  
    "executionNumber": 2939653338,  
    "jobId": "SampleJob",  
    "lastUpdatedAt": 1567714853.743,  
    "queuedAt": 1567701902.444,  
    "startedAt": 1567714871.690,  
    "status": "IN_PROGRESS",  
    "thingName": "MotionSensor1 ",  
    "versionNumber": 3  
  }  
}
```

Para obtener más información, consulte [Dispositivos y trabajos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [StartNextPendingJobExecution](#) la Referencia de AWS CLI comandos.

## update-job-execution

En el siguiente ejemplo de código, se muestra cómo usar `update-job-execution`.

AWS CLI

Para actualizar el estado de la ejecución de un trabajo

En el siguiente `update-job-execution` ejemplo, se actualiza el estado de la tarea y el objeto especificados.

```
aws iot-jobs-data update-job-execution \  
  --job-id SampleJob \  
  --status IN_PROGRESS \  
  --thing-name MotionSensor1 \  
  --version-number 3
```

```
--job-id SampleJob \  
--thing-name MotionSensor1 \  
--status REMOVED \  
--endpoint-url https://1234567890abcd.jobs.iot.us-west-2.amazonaws.com
```

Salida:

```
{  
  "executionState": {  
    "status": "REMOVED",  
    "versionNumber": 3  
  },  
}
```

Para obtener más información, consulte [Dispositivos y trabajos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdateJobExecution](#) la Referencia de AWS CLI comandos.

## AWS IoT SiteWise ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS IoT SiteWise.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **associate-assets**

En el siguiente ejemplo de código, se muestra cómo usar `associate-assets`.

## AWS CLI

Para asociar un activo secundario a un activo principal

El siguiente `associate-assets` ejemplo asocia un activo de turbina eólica a un activo de parque eólico, donde el modelo de activos de turbina eólica existe como una jerarquía en el modelo de activos de parque eólico.

```
aws iotsitewise associate-assets \  
  --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE \  
  --hierarchy-id a1b2c3d4-5678-90ab-cdef-77777EXAMPLE \  
  --child-asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Asociar activos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [AssociateAssets](#) la Referencia de AWS CLI comandos.

## batch-associate-project-assets

En el siguiente ejemplo de código, se muestra cómo usar `batch-associate-project-assets`.

### AWS CLI

Para asociar un activo a un proyecto

El siguiente `batch-associate-project-assets` ejemplo asocia un activo de parque eólico a un proyecto.

```
aws iotsitewise batch-associate-project-assets \  
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE \  
  --asset-ids a1b2c3d4-5678-90ab-cdef-44444EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir activos a los proyectos](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [BatchAssociateProjectAssets](#) la Referencia de AWS CLI comandos.

## batch-disassociate-project-assets

En el siguiente ejemplo de código, se muestra cómo usar `batch-disassociate-project-assets`.

### AWS CLI

Para desasociar un activo de un proyecto

El siguiente `batch-disassociate-project-assets` ejemplo desvincula un activo de parque eólico de un proyecto.

```
aws iotsitewise batch-disassociate-project-assets \  
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE \  
  --asset-ids a1b2c3d4-5678-90ab-cdef-44444EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir activos a los proyectos](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [BatchDisassociateProjectAssets](#) la Referencia de AWS CLI comandos.

## batch-put-asset-property-value

En el siguiente ejemplo de código, se muestra cómo usar `batch-put-asset-property-value`.

### AWS CLI

Para enviar datos a las propiedades de los activos

En el siguiente `batch-put-asset-property-value` ejemplo, se envían datos de potencia y temperatura a las propiedades de los activos identificadas por los alias de las propiedades.

```
aws iotsitewise batch-put-asset-property-value \  
  --cli-input-json file://batch-put-asset-property-value.json
```

Contenidos de `batch-put-asset-property-value.json`:

```
{
  "entries": [
    {
      "entryId": "1575691200-company-windfarm-3-turbine-7-power",
      "propertyAlias": "company-windfarm-3-turbine-7-power",
      "propertyValues": [
        {
          "value": {
            "doubleValue": 4.92
          },
          "timestamp": {
            "timeInSeconds": 1575691200
          },
          "quality": "GOOD"
        }
      ]
    },
    {
      "entryId": "1575691200-company-windfarm-3-turbine-7-temperature",
      "propertyAlias": "company-windfarm-3-turbine-7-temperature",
      "propertyValues": [
        {
          "value": {
            "integerValue": 38
          },
          "timestamp": {
            "timeInSeconds": 1575691200
          }
        }
      ]
    }
  ]
}
```

Salida:

```
{
  "errorEntries": []
}
```

Para obtener más información, consulte [Ingesta de datos mediante el AWS IoT SiteWise API](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [BatchPutAssetPropertyValue](#) la Referencia de AWS CLI comandos.

## create-access-policy

En el siguiente ejemplo de código, se muestra cómo usar `create-access-policy`.

### AWS CLI

Ejemplo 1: conceder a un usuario acceso administrativo a un portal

El siguiente `create-access-policy` ejemplo crea una política de acceso que concede a un usuario acceso administrativo a un portal web de una empresa de parques eólicos.

```
aws iotsitewise create-access-policy \  
  --cli-input-json file://create-portal-administrator-access-policy.json
```

Contenidos de `create-portal-administrator-access-policy.json`:

```
{  
  "accessPolicyIdentity": {  
    "user": {  
      "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE"  
    }  
  },  
  "accessPolicyPermission": "ADMINISTRATOR",  
  "accessPolicyResource": {  
    "portal": {  
      "id": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE"  
    }  
  }  
}
```

Salida:

```
{  
  "accessPolicyId": "a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE",  
  "accessPolicyArn": "arn:aws:iotsitewise:us-west-2:123456789012:access-policy/  
a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE"  
}
```



Para obtener más información, consulte [Añadir o eliminar administradores de portales](#) en la Guía del SiteWise usuario de AWS IoT.

Ejemplo 2: Para conceder a un usuario acceso de solo lectura a un proyecto

El siguiente `create-access-policy` ejemplo crea una política de acceso que concede a un usuario acceso de solo lectura a un proyecto de parque eólico.

```
aws iotsitewise create-access-policy \  
  --cli-input-json file://create-project-viewer-access-policy.json
```

Contenidos de `create-project-viewer-access-policy.json`:

```
{  
  "accessPolicyIdentity": {  
    "user": {  
      "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE"  
    }  
  },  
  "accessPolicyPermission": "VIEWER",  
  "accessPolicyResource": {  
    "project": {  
      "id": "a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE"  
    }  
  }  
}
```

Salida:

```
{  
  "accessPolicyId": "a1b2c3d4-5678-90ab-cdef-dddddEXAMPLE",  
  "accessPolicyArn": "arn:aws:iotsitewise:us-west-2:123456789012:access-policy/  
a1b2c3d4-5678-90ab-cdef-dddddEXAMPLE"  
}
```

Para obtener más información, consulte [Asignación de espectadores de proyectos](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [CreateAccessPolicy](#) la Referencia de AWS CLI comandos.

## create-asset-model

En el siguiente ejemplo de código, se muestra cómo usar `create-asset-model`.

### AWS CLI

Para crear un modelo de activos

El siguiente `create-asset-model` ejemplo crea un modelo de activos que define un aerogenerador con las siguientes propiedades:

Número de serie - El número de serie de una turbineGenerated central eólica - El flujo de datos de potencia generada por un viento turbineTemperature C - El flujo de datos de temperatura de una turbina eólica en CelsiusTemperature F - Los puntos de datos de temperatura mapeados de grados Celsius a Fahrenheit

```
aws iotsitewise create-asset-model \  
  --cli-input-json file://create-wind-turbine-model.json
```

Contenidos de `create-wind-turbine-model.json`:

```
{  
  "assetModelName": "Wind Turbine Model",  
  "assetModelDescription": "Represents a wind turbine",  
  "assetModelProperties": [  
    {  
      "name": "Serial Number",  
      "dataType": "STRING",  
      "type": {  
        "attribute": {}  
      }  
    },  
    {  
      "name": "Generated Power",  
      "dataType": "DOUBLE",  
      "unit": "kW",  
      "type": {  
        "measurement": {}  
      }  
    },  
    {  
      "name": "Temperature C",  
      "dataType": "DOUBLE",
```

```
    "unit": "Celsius",
    "type": {
      "measurement": {}
    }
  },
  {
    "name": "Temperature F",
    "dataType": "DOUBLE",
    "unit": "Fahrenheit",
    "type": {
      "transform": {
        "expression": "temp_c * 9 / 5 + 32",
        "variables": [
          {
            "name": "temp_c",
            "value": {
              "propertyId": "Temperature C"
            }
          }
        ]
      }
    }
  },
  {
    "name": "Total Generated Power",
    "dataType": "DOUBLE",
    "unit": "kW",
    "type": {
      "metric": {
        "expression": "sum(power)",
        "variables": [
          {
            "name": "power",
            "value": {
              "propertyId": "Generated Power"
            }
          }
        ],
        "window": {
          "tumbling": {
            "interval": "1h"
          }
        }
      }
    }
  }
}
```

```
}  
  }  
]  
}
```

Salida:

```
{  
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
  "assetModelArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
  "assetModelStatus": {  
    "state": "CREATING"  
  }  
}
```

Para obtener más información, consulte [Definir modelos de activos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [CreateAssetModel](#) la Referencia de AWS CLI comandos.

## create-asset

En el siguiente ejemplo de código, se muestra cómo usar `create-asset`.

### AWS CLI

Para crear un activo

El siguiente `create-asset` ejemplo crea un activo de turbina eólica a partir de un modelo de activos de turbina eólica.

```
aws iotsitewise create-asset \  
  --asset-model-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \  
  --asset-name "Wind Turbine 1"
```

Salida:

```
{  
  "assetId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
```

```

    "assetArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/
a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
    "assetStatus": {
        "state": "CREATING"
    }
}

```

Para obtener más información, consulte [Creación de activos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [CreateAsset](#) la Referencia de AWS CLI comandos.

## create-dashboard

En el siguiente ejemplo de código, se muestra cómo usar create-dashboard.

### AWS CLI

Para crear un cuadro de mando

En el siguiente create-dashboard ejemplo, se crea un cuadro de mando con un gráfico de líneas que muestra la potencia total generada por un parque eólico.

```

aws iotsitewise create-dashboard \
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE \
  --dashboard-name "Wind Farm" \
  --dashboard-definition file://create-wind-farm-dashboard.json

```

Contenidos de create-wind-farm-dashboard.json:

```

{
  "widgets": [
    {
      "type": "monitor-line-chart",
      "title": "Generated Power",
      "x": 0,
      "y": 0,
      "height": 3,
      "width": 3,
      "metrics": [
        {
          "label": "Power",

```

```

        "type": "iotsitewise",
        "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
        "propertyId": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE"
      }
    ]
  }
}

```

Salida:

```

{
  "dashboardId": "a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE",
  "dashboardArn": "arn:aws:iotsitewise:us-west-2:123456789012:dashboard/
a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE"
}

```

Para obtener más información, consulte [Creación de paneles \(CLI\)](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [CreateDashboard](#) la Referencia de AWS CLI comandos.

## create-gateway

En el siguiente ejemplo de código, se muestra cómo usar create-gateway.

### AWS CLI

Para crear una puerta de enlace

El siguiente create-gateway ejemplo crea una puerta de enlace que se ejecuta en AWS IoT Greengrass.

```

aws iotsitewise create-gateway \
  --gateway-name ExampleCorpGateway \
  --gateway-platform greengrass={groupArn=arn:aws:greengrass:us-
west-2:123456789012:/greengrass/groups/a1b2c3d4-5678-90ab-cdef-1b1b1EXAMPLE}

```

Salida:

```

{

```

```

    "gatewayId": "a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
    "gatewayArn": "arn:aws:iotsitewise:us-west-2:123456789012:gateway/
a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE"
}

```

Para obtener más información, consulte [Configuración de una puerta de enlace](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [CreateGateway](#) la Referencia de AWS CLI comandos.

## create-portal

En el siguiente ejemplo de código, se muestra cómo usar `create-portal`.

### AWS CLI

Para crear un portal

El siguiente `create-portal` ejemplo crea un portal web para una empresa de parques eólicos. Solo puede crear portales en la misma región en la que haya activado el inicio de sesión AWS único.

```

aws iotsitewise create-portal \
  --portal-name WindFarmPortal \
  --portal-description "A portal that contains wind farm projects for Example Corp." \
  --portal-contact-email support@example.com \
  --role-arn arn:aws:iam::123456789012:role/service-role/MySiteWiseMonitorServiceRole

```

Salida:

```

{
  "portalId": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
  "portalArn": "arn:aws:iotsitewise:us-west-2:123456789012:portal/
a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
  "portalStartUrl": "https://a1b2c3d4-5678-90ab-cdef-
aaaaaEXAMPLE.app.iotsitewise.aws",
  "portalStatus": {
    "state": "CREATING"
  },
}

```

```
"ssoApplicationId": "ins-a1b2c3d4-EXAMPLE"  
}
```

Para obtener más información, consulte [Introducción a AWS IoT SiteWise Monitor](#) en la Guía del SiteWise usuario de AWS IoT y [Habilitación AWS SSO](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [CreatePortalla](#) Referencia de AWS CLI comandos.

## create-project

En el siguiente ejemplo de código, se muestra cómo usar `create-project`.

### AWS CLI

Para crear un proyecto

El siguiente `create-project` ejemplo crea un proyecto de parque eólico.

```
aws iotsitewise create-project \  
  --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE \  
  --project-name "Wind Farm 1" \  
  --project-description "Contains asset visualizations for Wind Farm #1 for  
Example Corp."
```

Salida:

```
{  
  "projectId": "a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE",  
  "projectArn": "arn:aws:iotsitewise:us-west-2:123456789012:project/  
a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE"  
}
```

Para obtener más información, consulte [Creación de proyectos](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [CreateProjectla](#) Referencia de AWS CLI comandos.

## delete-access-policy

En el siguiente ejemplo de código, se muestra cómo usar `delete-access-policy`.



## AWS CLI

Para revocar el acceso de un usuario a un proyecto o portal

En el siguiente `delete-access-policy` ejemplo, se elimina una política de acceso que concede a un usuario acceso administrativo a un portal.

```
aws iotsitewise delete-access-policy \  
  --access-policy-id a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir o eliminar administradores de portales](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DeleteAccessPolicy](#) la Referencia de AWS CLI comandos.

## `delete-asset-model`

En el siguiente ejemplo de código, se muestra cómo usar `delete-asset-model`.

## AWS CLI

Para eliminar un modelo de activos

En el siguiente `delete-asset-model` ejemplo, se elimina un modelo de activos de una turbina eólica.

```
aws iotsitewise delete-asset-model \  
  --asset-model-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Salida:

```
{  
  "assetModelStatus": {  
    "state": "DELETING"  
  }  
}
```

Para obtener más información, consulte [Eliminar modelos de activos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DeleteAsset](#) la Referencia de AWS CLI comandos.

## delete-asset

En el siguiente ejemplo de código, se muestra cómo usar delete-asset.

### AWS CLI

Para eliminar un activo

En el siguiente delete-asset ejemplo, se elimina un activo de aerogenerador.

```
aws iotsitewise delete-asset \  
  --asset-id a1b2c3d4-5678-90ab-cdef-3333EXAMPLE
```

Salida:

```
{  
  "assetStatus": {  
    "state": "DELETING"  
  }  
}
```

Para obtener más información, consulte [Eliminar activos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DeleteAsset](#) la Referencia de AWS CLI comandos.

## delete-dashboard

En el siguiente ejemplo de código, se muestra cómo usar delete-dashboard.

### AWS CLI

Para eliminar un cuadro de mando

En el siguiente delete-dashboard ejemplo, se elimina el panel de control de un aerogenerador.

```
aws iotsitewise delete-dashboard \  
  --asset-id a1b2c3d4-5678-90ab-cdef-3333EXAMPLE
```

```
--dashboard-id a1b2c3d4-5678-90ab-cdef-ffffEXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar paneles](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [DeleteDashboard](#) la Referencia de AWS CLI comandos.

## delete-gateway

En el siguiente ejemplo de código, se muestra cómo usar delete-gateway.

### AWS CLI

Para eliminar una puerta de enlace

En el siguiente delete-gateway ejemplo, se elimina una puerta de enlace.

```
aws iotsitewise delete-gateway \  
  --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Ingesta de datos mediante una puerta de enlace](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DeleteGateway](#) la Referencia de AWS CLI comandos.

## delete-portal

En el siguiente ejemplo de código, se muestra cómo usar delete-portal.

### AWS CLI

Para eliminar un portal

En el siguiente delete-portal ejemplo, se elimina un portal web de una empresa de parques eólicos.

```
aws iotsitewise delete-portal \  
  --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE
```

Salida:

```
{  
  "portalStatus": {  
    "state": "DELETING"  
  }  
}
```

Para obtener más información, consulte [Eliminar un portal](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DeletePortal](#) la Referencia de AWS CLI comandos.

## delete-project

En el siguiente ejemplo de código, se muestra cómo usar `delete-project`.

AWS CLI

Para eliminar un proyecto

El siguiente `delete-project` ejemplo elimina un proyecto de parque eólico.

```
aws iotsitewise delete-project \  
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar proyectos](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [DeleteProject](#) la Referencia de AWS CLI comandos.

## describe-access-policy

En el siguiente ejemplo de código, se muestra cómo usar `describe-access-policy`.

## AWS CLI

Para describir una política de acceso

El siguiente `describe-access-policy` ejemplo describe una política de acceso que concede a un usuario acceso administrativo a un portal web de una empresa de parques eólicos.

```
aws iotsitewise describe-access-policy \  
  --access-policy-id a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE
```

Salida:

```
{  
  "accessPolicyId": "a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE",  
  "accessPolicyArn": "arn:aws:iotsitewise:us-west-2:123456789012:access-policy/  
a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE",  
  "accessPolicyIdentity": {  
    "user": {  
      "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbbEXAMPLE"  
    }  
  },  
  "accessPolicyResource": {  
    "portal": {  
      "id": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE"  
    }  
  },  
  "accessPolicyPermission": "ADMINISTRATOR",  
  "accessPolicyCreationDate": "2020-02-20T22:35:15.552880124Z",  
  "accessPolicyLastUpdateDate": "2020-02-20T22:35:15.552880124Z"  
}
```

Para obtener más información, consulte [Añadir o eliminar administradores de portales](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DescribeAccessPolicy](#) la Referencia de AWS CLI comandos.

## **describe-asset-model**

En el siguiente ejemplo de código, se muestra cómo usar `describe-asset-model`.

## AWS CLI

Para describir un modelo de activos

El siguiente `describe-asset-model` ejemplo describe un modelo de activos de parque eólico.

```
aws iotsitewise describe-asset-model \  
  --asset-model-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Salida:

```
{  
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
  "assetModelArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/  
a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
  "assetModelName": "Wind Farm Model",  
  "assetModelDescription": "Represents a wind farm that comprises many wind  
turbines",  
  "assetModelProperties": [  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",  
      "name": "Total Generated Power",  
      "dataType": "DOUBLE",  
      "unit": "kW",  
      "type": {  
        "metric": {  
          "expression": "sum(power)",  
          "variables": [  
            {  
              "name": "power",  
              "value": {  
                "propertyId": "a1b2c3d4-5678-90ab-  
cdef-66666EXAMPLE",  
                "hierarchyId": "a1b2c3d4-5678-90ab-  
cdef-77777EXAMPLE"  
              }  
            }  
          ],  
          "window": {  
            "tumbling": {  
              "interval": "1h"  
            }  
          }  
        }  
      }  
    ]  
  }  
}
```

```

    }
  }
},
{
  "id": "a1b2c3d4-5678-90ab-cdef-88888EXAMPLE",
  "name": "Region",
  "dataType": "STRING",
  "type": {
    "attribute": {
      "defaultValue": " "
    }
  }
}
],
"assetModelHierarchies": [
  {
    "id": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE",
    "name": "Wind Turbines",
    "childAssetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
  }
],
"assetModelCreationDate": 1575671284.0,
"assetModelLastUpdateDate": 1575671988.0,
"assetModelStatus": {
  "state": "ACTIVE"
}
}

```

Para obtener más información, consulte [Descripción de un modelo de activo específico](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DescribeAssetModel](#) la Referencia de AWS CLI comandos.

## describe-asset-property

En el siguiente ejemplo de código, se muestra cómo usar `describe-asset-property`.

### AWS CLI

Para describir una propiedad de activo

El siguiente describe-asset-property ejemplo describe la propiedad de potencia total generada de un activo de parque eólico.

```
aws iotsitewise describe-asset-property \  
  --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE \  
  --property-id a1b2c3d4-5678-90ab-cdef-99999EXAMPLE
```

Salida:

```
{  
  "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",  
  "assetName": "Wind Farm 1",  
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
  "assetProperty": {  
    "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",  
    "name": "Total Generated Power",  
    "notification": {  
      "topic": "$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE/assets/a1b2c3d4-5678-90ab-cdef-44444EXAMPLE/properties/a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",  
      "state": "DISABLED"  
    },  
    "dataType": "DOUBLE",  
    "unit": "kW",  
    "type": {  
      "metric": {  
        "expression": "sum(power)",  
        "variables": [  
          {  
            "name": "power",  
            "value": {  
              "propertyId": "a1b2c3d4-5678-90ab-cdef-66666EXAMPLE",  
              "hierarchyId": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE"  
            }  
          }  
        ],  
      "window": {  
        "tumbling": {  
          "interval": "1h"  
        }  
      }  
    }  
  }  
}
```



```
}  
}
```

Para obtener más información, consulte [Descripción de una propiedad de activo específica](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DescribeAssetProperty](#) la Referencia de AWS CLI comandos.

## describe-asset

En el siguiente ejemplo de código, se muestra cómo usar `describe-asset`.

### AWS CLI

Para describir un activo

El siguiente `describe-asset` ejemplo describe un activo de parque eólico.

```
aws iotsitewise describe-asset \  
  --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE
```

Salida:

```
{  
  "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",  
  "assetArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/  
a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",  
  "assetName": "Wind Farm 1",  
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
  "assetProperties": [  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-88888EXAMPLE",  
      "name": "Region",  
      "dataType": "STRING"  
    },  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",  
      "name": "Total Generated Power",  
      "dataType": "DOUBLE",  
      "unit": "kW"  
    }  
  ]  
}
```

```

],
"assetHierarchies": [
  {
    "id": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE",
    "name": "Wind Turbines"
  }
],
"assetCreationDate": 1575672453.0,
"assetLastUpdateDate": 1575672453.0,
"assetStatus": {
  "state": "ACTIVE"
}
}

```

Para obtener más información, consulte [Descripción de un activo específico](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DescribeAsset](#) la Referencia de AWS CLI comandos.

## describe-dashboard

En el siguiente ejemplo de código, se muestra cómo usar describe-dashboard.

### AWS CLI

Para describir un panel

El siguiente describe-dashboard ejemplo describe el panel de control del parque eólico especificado.

```

aws iotsitewise describe-dashboard \
  --dashboard-id a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE

```

Salida:

```

{
  "dashboardId": "a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE",
  "dashboardArn": "arn:aws:iotsitewise:us-west-2:123456789012:dashboard/a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE",
  "dashboardName": "Wind Farm",
  "projectId": "a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE",
}

```

```

    "dashboardDefinition": "{\\"widgets\\":[{\\"type\\":\\"monitor-line-chart\\",\\"title\\":\\"Generated Power\\",\\"x\\":0,\\"y\\":0,\\"height\\":3,\\"width\\":3,\\"metrics\\":[{\\"label\\":\\"Power\\",\\"type\\":\\"iotsitewise\\",\\"assetId\\":\\"a1b2c3d4-5678-90ab-cdef-44444EXAMPLE\\",\\"propertyId\\":\\"a1b2c3d4-5678-90ab-cdef-99999EXAMPLE\\"}]]}],\\"dashboardCreationDate\\":\\"2020-05-01T20:32:12.228476348Z\\",\\"dashboardLastUpdateDate\\":\\"2020-05-01T20:32:12.228476348Z\\"}

```

Para obtener más información, consulte [Visualización de paneles](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [DescribeDashboard](#) la Referencia de AWS CLI comandos.

## describe-gateway-capability-configuration

En el siguiente ejemplo de código, se muestra cómo usar `describe-gateway-capability-configuration`.

### AWS CLI

Para describir la capacidad de una puerta de enlace

El siguiente `describe-gateway-capability-configuration` ejemplo describe una capacidad de fuente OPC-UA.

```

aws iotsitewise describe-gateway-capability-configuration \
  --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE \
  --capability-namespace "iotsitewise:opcuacollector:1"

```

Salida:

```

{
  "gatewayId": "a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
  "capabilityNamespace": "iotsitewise:opcuacollector:1",
  "capabilityConfiguration": "{\\"sources\\":[{\\"name\\":\\"Wind Farm #1\\",\\"endpoint\\":{\\"certificateTrust\\":{\\"type\\":\\"TrustAny\\"},\\"endpointUri\\":\\"opc.tcp://203.0.113.0:49320\\",\\"securityPolicy\\":\\"BASIC256\\",\\"messageSecurityMode\\":\\"SIGN_AND_ENCRYPT\\",\\"identityProvider\\":{\\"type\\":\\"Username\\",\\"usernameSecretArn\\":\\"arn:aws:secretsmanager:us-east-1:123456789012:secret:green-grass-factory1-auth-3QNDmM\\"},\\"nodeFilterRules\\":[],\\"measurementDataStreamPrefix\\":\\"\\"}]}",

```

```
"capabilitySyncStatus": "IN_SYNC"
}
```

Para obtener más información, consulte [Configuración de fuentes de datos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DescribeGatewayCapabilityConfiguration](#) la Referencia de AWS CLI comandos.

## describe-gateway

En el siguiente ejemplo de código, se muestra cómo usar `describe-gateway`.

### AWS CLI

Para describir una puerta de enlace

El siguiente `describe-gateway` ejemplo describe una puerta de enlace.

```
aws iotsitewise describe-gateway \
  --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE
```

Salida:

```
{
  "gatewayId": "a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
  "gatewayName": "ExampleCorpGateway",
  "gatewayArn": "arn:aws:iotsitewise:us-west-2:123456789012:gateway/a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
  "gatewayPlatform": {
    "greengrass": {
      "groupArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/a1b2c3d4-5678-90ab-cdef-1b1b1EXAMPLE"
    }
  },
  "gatewayCapabilitySummaries": [
    {
      "capabilityNamespace": "iotsitewise:opcuacollector:1",
      "capabilitySyncStatus": "IN_SYNC"
    }
  ],
  "creationDate": 1588369971.457,
  "lastUpdateDate": 1588369971.457
}
```

```
}
```

Para obtener más información, consulte [Ingesta de datos mediante una puerta de enlace](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DescribeGateway](#) la Referencia de AWS CLI comandos.

## **describe-logging-options**

En el siguiente ejemplo de código, se muestra cómo usar `describe-logging-options`.

### AWS CLI

Para recuperar las opciones de SiteWise registro de AWS IoT actuales

En el siguiente `describe-logging-options` ejemplo, se recuperan las opciones de SiteWise registro de AWS IoT actuales para su AWS cuenta en la región actual.

```
aws iotsitewise describe-logging-options
```

Salida:

```
{
  "loggingOptions": {
    "level": "INFO"
  }
}
```

Para obtener más información, consulte [Supervisión de AWS IoT SiteWise con Amazon CloudWatch Logs](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DescribeLoggingOptions](#) la Referencia de AWS CLI comandos.

## **describe-portal**

En el siguiente ejemplo de código, se muestra cómo usar `describe-portal`.

### AWS CLI

Para describir un portal

El siguiente `describe-portal` ejemplo describe un portal web para una empresa de parques eólicos.

```
aws iotsitewise describe-portal \  
  --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE
```

Salida:

```
{  
  "portalId": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",  
  "portalArn": "arn:aws:iotsitewise:us-west-2:123456789012:portal/  
a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",  
  "portalName": "WindFarmPortal",  
  "portalDescription": "A portal that contains wind farm projects for Example  
Corp.",  
  "portalClientId": "E-a1b2c3d4e5f6_a1b2c3d4e5f6EXAMPLE",  
  "portalStartUrl": "https://a1b2c3d4-5678-90ab-cdef-  
aaaaaEXAMPLE.app.iotsitewise.aws",  
  "portalContactEmail": "support@example.com",  
  "portalStatus": {  
    "state": "ACTIVE"  
  },  
  "portalCreationDate": "2020-02-04T23:01:52.90248068Z",  
  "portalLastUpdateDate": "2020-02-04T23:01:52.90248078Z",  
  "roleArn": "arn:aws:iam::123456789012:role/MySiteWiseMonitorServiceRole"  
}
```

Para obtener más información, consulte [Administrar sus portales](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DescribePortalla](#) Referencia de AWS CLI comandos.

## **describe-project**

En el siguiente ejemplo de código, se muestra cómo usar `describe-project`.

AWS CLI

Para describir un proyecto

El siguiente `describe-project` ejemplo describe un proyecto de parque eólico.

```
aws iotsitewise describe-project \  
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE
```

Salida:

```
{  
  "projectId": "a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE",  
  "projectArn": "arn:aws:iotsitewise:us-west-2:123456789012:project/  
a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE",  
  "projectName": "Wind Farm 1",  
  "portalId": "a1b2c3d4-5678-90ab-cdef-aaaaEXAMPLE",  
  "projectDescription": "Contains asset visualizations for Wind Farm #1 for  
Example Corp.",  
  "projectCreationDate": "2020-02-20T21:58:43.362246001Z",  
  "projectLastUpdateDate": "2020-02-20T21:58:43.362246095Z"  
}
```

Para obtener más información, consulte [Visualización de los detalles del proyecto](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [DescribeProject](#) la Referencia de AWS CLI comandos.

## disassociate-assets

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-assets`.

### AWS CLI

Para desasociar un activo secundario de un activo principal

El siguiente `disassociate-assets` ejemplo disocia un activo de turbina eólica de un activo de parque eólico.

```
aws iotsitewise disassociate-assets \  
  --asset-id a1b2c3d4-5678-90ab-cdef-4444EXAMPLE \  
  --hierarchy-id a1b2c3d4-5678-90ab-cdef-7777EXAMPLE \  
  --child-asset-id a1b2c3d4-5678-90ab-cdef-3333EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Asociar activos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [DisassociateAssets](#) la Referencia de AWS CLI comandos.

## get-asset-property-aggregates

En el siguiente ejemplo de código, se muestra cómo usar `get-asset-property-aggregates`.

### AWS CLI

Para recuperar el promedio agregado y los valores de recuento de una propiedad

El siguiente `get-asset-property-aggregates` ejemplo recupera la potencia total media de un activo de turbina eólica y el recuento de los puntos de datos de potencia total durante un período de 1 hora.

```
aws iotsitewise get-asset-property-aggregates \  
  --asset-id a1b2c3d4-5678-90ab-cdef-3333EXAMPLE \  
  --property-id a1b2c3d4-5678-90ab-cdef-6666EXAMPLE \  
  --start-date 1580849400 \  
  --end-date 1580853000 \  
  --aggregate-types AVERAGE COUNT \  
  --resolution 1h
```

Salida:

```
{  
  "aggregatedValues": [  
    {  
      "timestamp": 1580850000.0,  
      "quality": "GOOD",  
      "value": {  
        "average": 8723.46538886233,  
        "count": 12.0  
      }  
    }  
  ]  
}
```



Para obtener más información, consulte [Consulta los agregados de propiedades de los activos en la Guía SiteWise del usuario de AWS IoT](#).

- Para API obtener más información, consulte la Referencia [GetAssetPropertyAggregates](#) de AWS CLI comandos.

## get-asset-property-value-history

En el siguiente ejemplo de código, se muestra cómo usar `get-asset-property-value-history`.

### AWS CLI

Para recuperar los valores históricos de una propiedad de un activo

El siguiente `get-asset-property-value-history` ejemplo recupera los valores de potencia total de un activo de turbina eólica durante un período de 20 minutos.

```
aws iotsitewise get-asset-property-value-history \  
  --asset-id a1b2c3d4-5678-90ab-cdef-3333EXAMPLE \  
  --property-id a1b2c3d4-5678-90ab-cdef-6666EXAMPLE \  
  --start-date 1580851800 \  
  --end-date 1580853000
```

Salida:

```
{  
  "assetPropertyValueHistory": [  
    {  
      "value": {  
        "doubleValue": 7217.787046814844  
      },  
      "timestamp": {  
        "timeInSeconds": 1580852100,  
        "offsetInNanos": 0  
      },  
      "quality": "GOOD"  
    },  
    {  
      "value": {  
        "doubleValue": 6941.242811875451  
      },  
      "timestamp": {  
        "timeInSeconds": 1580852100,  
        "offsetInNanos": 0  
      },  
      "quality": "GOOD"  
    }  
  ]  
}
```

```
    "timestamp": {
      "timeInSeconds": 1580852400,
      "offsetInNanos": 0
    },
    "quality": "GOOD"
  },
  {
    "value": {
      "doubleValue": 6976.797662266717
    },
    "timestamp": {
      "timeInSeconds": 1580852700,
      "offsetInNanos": 0
    },
    "quality": "GOOD"
  },
  {
    "value": {
      "doubleValue": 6890.8677520453875
    },
    "timestamp": {
      "timeInSeconds": 1580853000,
      "offsetInNanos": 0
    },
    "quality": "GOOD"
  }
]
}
```

Para obtener más información, consulte [Consulta los valores históricos de las propiedades de los activos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [GetAssetPropertyValueHistory](#) la Referencia de AWS CLI comandos.

## get-asset-property-value

En el siguiente ejemplo de código, se muestra cómo usar `get-asset-property-value`.

### AWS CLI

Para recuperar el valor actual de una propiedad de un activo

El siguiente `get-asset-property-value` ejemplo recupera la potencia total actual de un activo de turbina eólica.

```
aws iotsitewise get-asset-property-value \  
  --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \  
  --property-id a1b2c3d4-5678-90ab-cdef-66666EXAMPLE
```

Salida:

```
{  
  "propertyValue": {  
    "value": {  
      "doubleValue": 6890.8677520453875  
    },  
    "timestamp": {  
      "timeInSeconds": 1580853000,  
      "offsetInNanos": 0  
    },  
    "quality": "GOOD"  
  }  
}
```

Para obtener más información, consulte [Consulta los valores actuales de las propiedades de los activos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [GetAssetPropertyValue](#) la Referencia de AWS CLI comandos.

## list-access-policies

En el siguiente ejemplo de código, se muestra cómo usar `list-access-policies`.

### AWS CLI

Para enumerar todas las políticas de acceso

En el siguiente `list-access-policies` ejemplo, se enumeran todas las políticas de acceso de un usuario que es administrador del portal.

```
aws iotsitewise list-access-policies \  
  --identity-type USER \  
  --
```

```
--identity-id a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE
```

Salida:

```
{
  "accessPolicySummaries": [
    {
      "id": "a1b2c3d4-5678-90ab-cdef-ccccEXAMPLE",
      "identity": {
        "user": {
          "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE"
        }
      },
      "resource": {
        "portal": {
          "id": "a1b2c3d4-5678-90ab-cdef-aaaaEXAMPLE"
        }
      },
      "permission": "ADMINISTRATOR"
    }
  ]
}
```

Para obtener más información, consulte [Administrar sus portales](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [ListAccessPolicies](#) la Referencia de AWS CLI comandos.

## **list-asset-models**

En el siguiente ejemplo de código, se muestra cómo usar `list-asset-models`.

AWS CLI

Para enumerar todos los modelos de activos

El siguiente `list-asset-models` ejemplo muestra todos los modelos de activos que están definidos en su AWS cuenta en la región actual.

```
aws iotsitewise list-asset-models
```

Salida:

```
{
  "assetModelSummaries": [
    {
      "id": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
      "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
      "name": "Wind Farm Model",
      "description": "Represents a wind farm that comprises many wind turbines",
      "creationDate": 1575671284.0,
      "lastUpdateDate": 1575671988.0,
      "status": {
        "state": "ACTIVE"
      }
    },
    {
      "id": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "name": "Wind Turbine Model",
      "description": "Represents a wind turbine manufactured by Example Corp",
      "creationDate": 1575671207.0,
      "lastUpdateDate": 1575686273.0,
      "status": {
        "state": "ACTIVE"
      }
    }
  ]
}
```

Para obtener más información, consulte [Listar todos los modelos de activos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [ListAssetModels](#) la Referencia de AWS CLI comandos.

## list-assets

En el siguiente ejemplo de código, se muestra cómo usar `list-assets`.

## AWS CLI

Ejemplo 1: Para enumerar todos los activos de nivel superior

En el siguiente `list-assets` ejemplo, se enumeran todos los activos que ocupan el primer nivel del árbol jerárquico de activos y están definidos en su AWS cuenta en la región actual.

```
aws iotsitewise list-assets \  
  --filter TOP_LEVEL
```

Salida:

```
{  
  "assetSummaries": [  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",  
      "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/  
a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",  
      "name": "Wind Farm 1",  
      "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",  
      "creationDate": 1575672453.0,  
      "lastUpdateDate": 1575672453.0,  
      "status": {  
        "state": "ACTIVE"  
      },  
      "hierarchies": [  
        {  
          "id": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE",  
          "name": "Wind Turbines"  
        }  
      ]  
    }  
  ]  
}
```

Para obtener más información, consulta Cómo [enumerar activos](#) en la Guía del SiteWise usuario de AWS IoT.

Ejemplo 2: Para enumerar todos los activos en función de un modelo de activos

En el siguiente `list-assets` ejemplo, se enumeran todos los activos según un modelo de activos y definidos en su AWS cuenta en la región actual.

```
aws iotsitewise list-assets \
  --asset-model-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Salida:

```
{
  "assetSummaries": [
    {
      "id": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
      "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
      "name": "Wind Turbine 1",
      "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "creationDate": 1575671550.0,
      "lastUpdateDate": 1575686308.0,
      "status": {
        "state": "ACTIVE"
      },
      "hierarchies": []
    }
  ]
}
```

Para obtener más información, consulta Cómo [enumerar activos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [ListAssets](#) la Referencia de AWS CLI comandos.

## list-associated-assets

En el siguiente ejemplo de código, se muestra cómo usar `list-associated-assets`.

AWS CLI

Para enumerar todos los activos asociados a un activo en una jerarquía específica

En el siguiente `list-associated-assets` ejemplo, se enumeran todos los activos de aerogeneradores asociados al activo de parque eólico especificado.

```
aws iotsitewise list-associated-assets \
  --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE \
```

```
--hierarchy-id a1b2c3d4-5678-90ab-cdef-7777EXAMPLE
```

Salida:

```
{
  "assetSummaries": [
    {
      "id": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
      "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
      "name": "Wind Turbine 1",
      "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "creationDate": 1575671550.0,
      "lastUpdateDate": 1575686308.0,
      "status": {
        "state": "ACTIVE"
      },
      "hierarchies": []
    }
  ]
}
```

Para obtener más información, consulte [Listar los activos asociados a un activo específico](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [ListAssociatedAssets](#) la Referencia de AWS CLI comandos.

## list-dashboards

En el siguiente ejemplo de código, se muestra cómo usar `list-dashboards`.

AWS CLI

Para enumerar todos los paneles de un proyecto

En el siguiente `list-dashboards` ejemplo, se enumeran todos los paneles que están definidos en un proyecto.

```
aws iotsitewise list-dashboards \
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeeeEXAMPLE
```



Salida:

```
{
  "dashboardSummaries": [
    {
      "id": "a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE",
      "name": "Wind Farm",
      "creationDate": "2020-05-01T20:32:12.228476348Z",
      "lastUpdateDate": "2020-05-01T20:32:12.228476348Z"
    }
  ]
}
```

Para obtener más información, consulte [Visualización de paneles](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [ListDashboards](#) la Referencia de AWS CLI comandos.

## list-gateways

En el siguiente ejemplo de código, se muestra cómo usar `list-gateways`.

### AWS CLI

Para ver una lista de todas las puertas de enlace

En el siguiente `list-gateways` ejemplo, se enumeran todas las puertas de enlace que están definidas en su AWS cuenta en la región actual.

```
aws iotsitewise list-gateways
```

Salida:

```
{
  "gatewaySummaries": [
    {
      "gatewayId": "a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
      "gatewayName": "ExampleCorpGateway",
      "gatewayCapabilitySummaries": [
        {
```

```

        "capabilityNamespace": "iotsitewise:opcuacollector:1",
        "capabilitySyncStatus": "IN_SYNC"
    }
],
"creationDate": 1588369971.457,
"lastUpdateDate": 1588369971.457
}
]
}

```

Para obtener más información, consulte [Ingesta de datos mediante una puerta de enlace](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [ListGateways](#) la Referencia de AWS CLI comandos.

## list-portals

En el siguiente ejemplo de código, se muestra cómo usar `list-portals`.

### AWS CLI

Para enumerar todos los portales

En el siguiente `list-portals` ejemplo, se enumeran todos los portales que están definidos en su AWS cuenta en la región actual.

```
aws iotsitewise list-portals
```

Salida:

```

{
  "portalSummaries": [
    {
      "id": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
      "name": "WindFarmPortal",
      "description": "A portal that contains wind farm projects for Example Corp.",
      "startUrl": "https://a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE.app.iotsitewise.aws",
      "creationDate": "2020-02-04T23:01:52.90248068Z",
      "lastUpdateDate": "2020-02-04T23:01:52.90248078Z",
    }
  ]
}

```

```

        "roleArn": "arn:aws:iam::123456789012:role/service-role/
MySiteWiseMonitorServiceRole"
    }
]
}

```

Para obtener más información, consulte [Administrar sus portales](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [ListPortals](#) la Referencia de AWS CLI comandos.

## list-project-assets

En el siguiente ejemplo de código, se muestra cómo usar `list-project-assets`.

### AWS CLI

Para enumerar todos los activos asociados a un proyecto

En el siguiente `list-project-assets` ejemplo se enumeran todos los activos que están asociados a un proyecto de parque eólico.

```

aws iotsitewise list-projects \
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE

```

Salida:

```

{
  "assetIds": [
    "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE"
  ]
}

```

Para obtener más información, consulte [Añadir activos a los proyectos](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [ListProjectAssets](#) la Referencia de AWS CLI comandos.

## list-projects

En el siguiente ejemplo de código, se muestra cómo usar `list-projects`.

## AWS CLI

Para enumerar todos los proyectos de un portal

En el siguiente `list-projects` ejemplo, se enumeran todos los proyectos que están definidos en un portal.

```
aws iotsitewise list-projects \  
  --portal-id a1b2c3d4-5678-90ab-cdef-aaaaEXAMPLE
```

Salida:

```
{  
  "projectSummaries": [  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE",  
      "name": "Wind Farm 1",  
      "description": "Contains asset visualizations for Wind Farm #1 for  
Example Corp.",  
      "creationDate": "2020-02-20T21:58:43.362246001Z",  
      "lastUpdateDate": "2020-02-20T21:58:43.362246095Z"  
    }  
  ]  
}
```

Para obtener más información, consulte [Visualización de los detalles del proyecto](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [ListProjects](#) la Referencia de AWS CLI comandos.

## `list-tags-for-resource`

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

## AWS CLI

Para enumerar todas las etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas de un activo de turbina eólica.

```
aws iotsitewise list-tags-for-resource \  
  --resource-id
```

```
--resource-arn arn:aws:iotsitewise:us-west-2:123456789012:asset/  
a1b2c3d4-5678-90ab-cdef-33333EXAMPLE
```

Salida:

```
{  
  "tags": {  
    "Owner": "richard-roe"  
  }  
}
```

Para obtener más información, consulte [Etiquetar sus recursos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## put-logging-options

En el siguiente ejemplo de código, se muestra cómo usar put-logging-options.

### AWS CLI

Para especificar el nivel de registro

El siguiente put-logging-options ejemplo habilita INFO el registro de niveles en AWS IoT SiteWise. Otros niveles incluyen DEBUG yOFF.

```
aws iotsitewise put-logging-options \  
  --logging-options Level=INFO
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Supervisión de AWS IoT SiteWise con Amazon CloudWatch Logs](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [PutLoggingOptions](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

## AWS CLI

Cómo añadir una etiqueta a un recurso

En el siguiente `tag-resource` ejemplo, se añade una etiqueta de propietario a un activo de aerogenerador. Esto le permite controlar el acceso al activo en función de quién es su propietario.

```
aws iotsitewise tag-resource \  
  --resource-arn arn:aws:iotsitewise:us-west-2:123456789012:asset/  
a1b2c3d4-5678-90ab-cdef-3333EXAMPLE \  
  --tags Owner=richard-roe
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar sus recursos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

## AWS CLI

Para eliminar una etiqueta de un recurso

En el siguiente `untag-resource` ejemplo, se elimina una etiqueta de propietario de un activo de aerogenerador.

```
aws iotsitewise untag-resource \  
  --resource-arn arn:aws:iotsitewise:us-west-2:123456789012:asset/  
a1b2c3d4-5678-90ab-cdef-3333EXAMPLE \  
  --tag-keys Owner
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar sus recursos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-access-policy

En el siguiente ejemplo de código, se muestra cómo usar `update-access-policy`.

### AWS CLI

Para conceder a un espectador de proyectos la propiedad de un proyecto

El siguiente `update-access-policy` ejemplo actualiza una política de acceso que otorga a un espectador de proyectos la propiedad de un proyecto.

```
aws iotsitewise update-access-policy \  
  --access-policy-id a1b2c3d4-5678-90ab-cdef-dddddEXAMPLE \  
  --cli-input-json file://update-project-viewer-access-policy.json
```

Contenidos de `update-project-viewer-access-policy.json`:

```
{  
  "accessPolicyIdentity": {  
    "user": {  
      "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE"  
    }  
  },  
  "accessPolicyPermission": "ADMINISTRATOR",  
  "accessPolicyResource": {  
    "project": {  
      "id": "a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE"  
    }  
  }  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Asignación de propietarios de proyectos](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [UpdateAccessPolicy](#) la Referencia de AWS CLI comandos.

## update-asset-model

En el siguiente ejemplo de código, se muestra cómo usar `update-asset-model`.

## AWS CLI

Para actualizar un modelo de activos

El siguiente `update-asset-model` ejemplo actualiza la descripción del modelo de activos de un parque eólico. Este ejemplo incluye el modelo existente IDs y las definiciones, ya que `update-asset-model` sobrescribe el modelo existente con el nuevo modelo.

```
aws iotsitewise update-asset-model \  
  --cli-input-json file://update-wind-farm-model.json
```

Contenidos de `update-wind-farm-model.json`:

```
{  
  "assetModelName": "Wind Farm Model",  
  "assetModelDescription": "Represents a wind farm that comprises many wind  
turbines",  
  "assetModelProperties": [  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-88888EXAMPLE",  
      "name": "Region",  
      "dataType": "STRING",  
      "type": {  
        "attribute": {}  
      }  
    },  
    {  
      "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",  
      "name": "Total Generated Power",  
      "dataType": "DOUBLE",  
      "unit": "kW",  
      "type": {  
        "metric": {  
          "expression": "sum(power)",  
          "variables": [  
            {  
              "name": "power",  
              "value": {  
                "hierarchyId": "a1b2c3d4-5678-90ab-  
cdef-77777EXAMPLE",  
                "propertyId": "a1b2c3d4-5678-90ab-cdef-66666EXAMPLE"  
              }  
            }  
          ]  
        }  
      }  
    }  
  ]  
}
```



```

    ],
    "window": {
      "tumbling": {
        "interval": "1h"
      }
    }
  }
],
"assetModelHierarchies": [
  {
    "id": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE",
    "name": "Wind Turbines",
    "childAssetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
  }
]
}

```

Salida:

```

{
  "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
  "assetModelArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
  "assetModelStatus": {
    "state": "CREATING"
  }
}

```

Para obtener más información, consulte [Actualización de modelos de activos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [UpdateAssetModel](#) la Referencia de AWS CLI comandos.

## update-asset-property

En el siguiente ejemplo de código, se muestra cómo usar `update-asset-property`.

AWS CLI

Ejemplo 1: Para actualizar el alias de una propiedad de un activo

En el siguiente `update-asset-property` ejemplo, se actualiza el alias de la propiedad de energía de un activo de turbina eólica.

```
aws iotsitewise update-asset-property \  
  --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \  
  --property-id a1b2c3d4-5678-90ab-cdef-55555EXAMPLE \  
  --property-alias "/examplecorp/windfarm/1/turbine/1/power" \  
  --property-notification-state DISABLED
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Asignación de flujos de datos industriales a propiedades de activos](#) en la Guía del SiteWise usuario de AWS IoT.

Ejemplo 2: Para habilitar las notificaciones de propiedades de activos

El siguiente `update-asset-property` ejemplo habilita las notificaciones de actualización de propiedades de activos para la propiedad de energía de un activo de aerogenerador. Las actualizaciones del valor de las propiedades se publican en el MQTT tema `aws/sitewise/asset-models/<assetModelId>/assets/<assetId>/properties/<propertyId>`, donde cada identificador se sustituye por el identificador de propiedad, activo y modelo de la propiedad del activo.

```
aws iotsitewise update-asset-property \  
  --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \  
  --property-id a1b2c3d4-5678-90ab-cdef-66666EXAMPLE \  
  --property-notification-state ENABLED \  
  --property-alias "/examplecorp/windfarm/1/turbine/1/power"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Interactuar con otros servicios](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [UpdateAssetProperty](#) la Referencia de AWS CLI comandos.

## update-asset

En el siguiente ejemplo de código, se muestra cómo usar `update-asset`.

## AWS CLI

Para actualizar el nombre de un activo

En el siguiente `update-asset` ejemplo, se actualiza el nombre de un activo de aerogenerador.

```
aws iotsitewise update-asset \  
  --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \  
  --asset-name "Wind Turbine 2"
```

Salida:

```
{  
  "assetStatus": {  
    "state": "UPDATING"  
  }  
}
```

Para obtener más información, consulte [Actualización de activos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [UpdateAsset](#) la Referencia de AWS CLI comandos.

## update-dashboard

En el siguiente ejemplo de código, se muestra cómo usar `update-dashboard`.

### AWS CLI

Para actualizar un panel

El siguiente `update-dashboard` ejemplo cambia el título del gráfico de líneas de un cuadro de mando que muestra la potencia total generada por un parque eólico.

```
aws iotsitewise update-dashboard \  
  --project-id a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE \  
  --dashboard-name "Wind Farm" \  
  --dashboard-definition file://update-wind-farm-dashboard.json
```

Contenidos de `update-wind-farm-dashboard.json`:

```
{
```

```
"widgets": [
  {
    "type": "monitor-line-chart",
    "title": "Total Generated Power",
    "x": 0,
    "y": 0,
    "height": 3,
    "width": 3,
    "metrics": [
      {
        "label": "Power",
        "type": "iotsitewise",
        "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
        "propertyId": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE"
      }
    ]
  }
]
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de paneles \(CLI\)](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [UpdateDashboard](#) la Referencia de AWS CLI comandos.

## update-gateway-capability-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-gateway-capability-configuration`.

### AWS CLI

Para actualizar la capacidad de una puerta de enlace

El siguiente `update-gateway-capability-configuration` ejemplo configura una fuente OPC-UA con las siguientes propiedades:

Confía en cualquier certificado. Utiliza el algoritmo Basic256 para proteger los mensajes. Utiliza el SignAndEncrypt modo para proteger las conexiones. Utiliza las credenciales de autenticación almacenadas en un secreto de Secrets Manager. AWS

```
aws iotsitewise update-gateway-capability-configuration \
  --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE \
  --capability-namespace "iotsitewise:opcuacollector:1" \
  --capability-configuration file://opc-ua-capability-configuration.json
```

Contenidos de `opc-ua-capability-configuration.json`:

```
{
  "sources": [
    {
      "name": "Wind Farm #1",
      "endpoint": {
        "certificateTrust": {
          "type": "TrustAny"
        },
        "endpointUri": "opc.tcp://203.0.113.0:49320",
        "securityPolicy": "BASIC256",
        "messageSecurityMode": "SIGN_AND_ENCRYPT",
        "identityProvider": {
          "type": "Username",
          "usernameSecretArn": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:greengrass-windfarm1-auth-1ABCDE"
        },
        "nodeFilterRules": []
      },
      "measurementDataStreamPrefix": ""
    }
  ]
}
```

Salida:

```
{
  "capabilityNamespace": "iotsitewise:opcuacollector:1",
  "capabilitySyncStatus": "OUT_OF_SYNC"
}
```

Para obtener más información, consulte [Configuración de fuentes de datos](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [UpdateGatewayCapabilityConfiguration](#) la Referencia de AWS CLI comandos.

## update-gateway

En el siguiente ejemplo de código, se muestra cómo usar `update-gateway`.

### AWS CLI

Para actualizar el nombre de una puerta de enlace

En el siguiente `update-gateway` ejemplo, se actualiza el nombre de una puerta de enlace.

```
aws iotsitewise update-gateway \  
  --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE \  
  --gateway-name ExampleCorpGateway1
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Ingesta de datos mediante una puerta de enlace](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [UpdateGateway](#) la Referencia de AWS CLI comandos.

## update-portal

En el siguiente ejemplo de código, se muestra cómo usar `update-portal`.

### AWS CLI

Para actualizar los detalles de un portal

En el siguiente `update-portal` ejemplo, se actualiza el portal web de una empresa de parques eólicos.

```
aws iotsitewise update-portal \  
  --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE \  
  --portal-name WindFarmPortal \  
  --portal-description "A portal that contains wind farm projects for Example Corp." \  
  --portal-contact-email support@example.com \  
  --role-arn arn:aws:iam::123456789012:role/MySiteWiseMonitorServiceRole
```

Salida:

```
{
  "portalStatus": {
    "state": "UPDATING"
  }
}
```

Para obtener más información, consulte [Administrar sus portales](#) en la Guía del SiteWise usuario de AWS IoT.

- Para API obtener más información, consulte [UpdatePortal](#) la Referencia de AWS CLI comandos.

## update-project

En el siguiente ejemplo de código, se muestra cómo usar `update-project`.

### AWS CLI

Para actualizar los detalles de un proyecto

El siguiente `update-project` ejemplo actualiza un proyecto de parque eólico.

```
aws iotsitewise update-project \
  --project-id a1b2c3d4-5678-90ab-cdef-eeeeEXAMPLE \
  --project-name "Wind Farm 1" \
  --project-description "Contains asset visualizations for Wind Farm #1 for Example Corp."
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cambiar los detalles del proyecto](#) en la Guía de aplicaciones de AWS IoT SiteWise Monitor.

- Para API obtener más información, consulte [UpdateProject](#) la Referencia de AWS CLI comandos.

## AWS IoT Things Graph ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS IoT Things Graph.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **associate-entity-to-thing**

En el siguiente ejemplo de código, se muestra cómo usar `associate-entity-to-thing`.

AWS CLI

Para asociar algo a un dispositivo

En el siguiente `associate-entity-to-thing` ejemplo, se asocia un objeto a un dispositivo. En el ejemplo se utiliza un dispositivo sensor de movimiento que se encuentra en el espacio de nombres público.

```
aws iotthingsgraph associate-entity-to-thing \  
  --thing-name "MotionSensorName" \  
  --entity-id "urn:tdm:aws/examples:Device:HCSR501MotionSensor"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación y carga de modelos](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [AssociateEntityToThing](#) la Referencia de AWS CLI comandos.

### **create-flow-template**

En el siguiente ejemplo de código, se muestra cómo usar `create-flow-template`.



## AWS CLI

Para crear un flujo

El siguiente `create-flow-template` ejemplo crea un flujo (flujo de trabajo). El valor de `MyFlowDefinition` es el GraphQL que modela el flujo.

```
aws iotthingsgraph create-flow-template \  
  --definition language=GRAPHQL,text="MyFlowDefinition"
```

Salida:

```
{  
  "summary": {  
    "createdAt": 1559248067.545,  
    "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",  
    "revisionNumber": 1  
  }  
}
```

Para obtener más información, consulte Cómo [trabajar con flujos](#) en la guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [CreateFlowTemplate](#) la Referencia de AWS CLI comandos.

## `create-system-instance`

En el siguiente ejemplo de código, se muestra cómo usar `create-system-instance`.

### AWS CLI

Para crear una instancia del sistema

En el siguiente `create-system-instance` ejemplo, se crea una instancia del sistema. El valor de `MySystemInstanceDefinition` es el GraphQL que modela la instancia del sistema.

```
aws iotthingsgraph create-system-instance -\  
  -definition language=GRAPHQL,text="MySystemInstanceDefinition" \  
  --target CLOUD \  
  --
```

```
--flow-actions-role-arn myRoleARN
```

Salida:

```
{
  "summary": {
    "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room218",
    "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/default/Room218",
    "status": "NOT_DEPLOYED",
    "target": "CLOUD",
    "createdAt": 1559249315.208,
    "updatedAt": 1559249315.208
  }
}
```

Para obtener más información, consulte [Trabajar con configuraciones de sistemas y flujos](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [CreateSystemInstance](#) la Referencia de AWS CLI comandos.

## create-system-template

En el siguiente ejemplo de código, se muestra cómo usar `create-system-template`.

AWS CLI

Para crear un sistema

En el `create-system-template` ejemplo siguiente se crea un sistema. El valor de `MySystemDefinition` es el GraphQL que modela el sistema.

```
aws iotthingsgraph create-system-template \
  --definition language=GRAPHQL,text="MySystemDefinition"
```

Salida:

```
{
  "summary": {
    "createdAt": 1559249776.254,
```

```
    "id": "urn:tdm:us-west-2/123456789012/default:System:MySystem",
    "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/
MySystem",
    "revisionNumber": 1
  }
}
```

Para obtener más información, consulte [Creación de sistemas](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [CreateSystemTemplate](#) la Referencia de AWS CLI comandos.

## delete-flow-template

En el siguiente ejemplo de código, se muestra cómo usar `delete-flow-template`.

### AWS CLI

Para eliminar un flujo

En el siguiente `delete-flow-template` ejemplo, se elimina un flujo (flujo de trabajo).

```
aws iotthingsgraph delete-flow-template \
  --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [DeleteFlowTemplate](#) la Referencia de AWS CLI comandos.

## delete-namespace

En el siguiente ejemplo de código, se muestra cómo usar `delete-namespace`.

### AWS CLI

Para eliminar un espacio de nombres

En el siguiente delete-namespace ejemplo, se elimina un espacio de nombres.

```
aws iotthingsgraph delete-namespace
```

Salida:

```
{
  "namespaceArn": "arn:aws:iotthingsgraph:us-west-2:123456789012",
  "namespaceName": "us-west-2/123456789012/default"
}
```

Para obtener más información, consulte [Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [DeleteNamespace](#) la Referencia de AWS CLI comandos.

## delete-system-instance

En el siguiente ejemplo de código, se muestra cómo usar delete-system-instance.

AWS CLI

Para eliminar una instancia del sistema

En el siguiente delete-system-instance ejemplo, se elimina una instancia del sistema.

```
aws iotthingsgraph delete-system-instance \
  --id "urn:tdm:us-west-2/123456789012/default:Deployment:Room218"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [DeleteSystemInstance](#) la Referencia de AWS CLI comandos.

## delete-system-template

En el siguiente ejemplo de código, se muestra cómo usar delete-system-template.

## AWS CLI

Para eliminar un sistema

En el siguiente `delete-system-template` ejemplo, se elimina un sistema.

```
aws iotthingsgraph delete-system-template \  
  --id "urn:tdm:us-west-2/123456789012/default:System:MySystem"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [DeleteSystemTemplate](#) la Referencia de AWS CLI comandos.

## `deploy-system-instance`

En el siguiente ejemplo de código, se muestra cómo usar `deploy-system-instance`.

## AWS CLI

Para implementar una instancia del sistema

En el siguiente `delete-system-template` ejemplo, se implementa una instancia del sistema.

```
aws iotthingsgraph deploy-system-instance \  
  --id "urn:tdm:us-west-2/123456789012/default:Deployment:Room218"
```

Salida:

```
{  
  "summary": {  
    "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment:Room218",  
    "createdAt": 1559249776.254,  
    "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room218",  
    "status": "DEPLOYED_IN_TARGET",  
    "target": "CLOUD",  
    "updatedAt": 1559249776.254  
  }  
}
```

```
}
```

Para obtener más información, consulte [Trabajar con configuraciones de sistemas y flujos](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [DeploySystemInstance](#) la Referencia de AWS CLI comandos.

## deprecate-flow-template

En el siguiente ejemplo de código, se muestra cómo usar `deprecate-flow-template`.

### AWS CLI

Para desaprobar un flujo

En el siguiente `deprecate-flow-template` ejemplo, se desaprueba un flujo (flujo de trabajo).

```
aws iotthingsgraph deprecate-flow-template \  
  --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [DeprecateFlowTemplate](#) la Referencia de AWS CLI comandos.

## deprecate-system-template

En el siguiente ejemplo de código, se muestra cómo usar `deprecate-system-template`.

### AWS CLI

Para desaprobar un sistema

En el siguiente `deprecate-system-template` ejemplo, se desaprueba un sistema.

```
aws iotthingsgraph deprecate-system-template \  
  --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow"
```

```
--id "urn:tdm:us-west-2/123456789012/default:System:MySystem"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [DeprecateSystemTemplate](#) la Referencia de AWS CLI comandos.

## describe-namespace

En el siguiente ejemplo de código, se muestra cómo usar `describe-namespace`.

### AWS CLI

Para obtener una descripción de su espacio de nombres

En el siguiente `describe-namespace` ejemplo, se obtiene una descripción del espacio de nombres.

```
aws iotthingsgraph describe-namespace
```

Salida:

```
{
  "namespaceName": "us-west-2/123456789012/default",
  "trackingNamespaceName": "aws",
  "trackingNamespaceVersion": 1,
  "namespaceVersion": 5
}
```

Para obtener más información, consulte los [espacios de nombres](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte la Referencia [DescribeNamespace](#) de AWS CLI comandos.

## dissociate-entity-from-thing

En el siguiente ejemplo de código, se muestra cómo usar `dissociate-entity-from-thing`.

## AWS CLI

Para disociar una cosa de un dispositivo

En el siguiente `dissociate-entity-from-thing` ejemplo, se disocia un objeto de un dispositivo.

```
aws iotthingsgraph dissociate-entity-from-thing \  
  --thing-name "MotionSensorName" \  
  --entity-type "DEVICE"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación y carga de modelos](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [DissociateEntityFromThing](#) la Referencia de AWS CLI comandos.

## get-entities

En el siguiente ejemplo de código, se muestra cómo usar `get-entities`.

## AWS CLI

Para obtener definiciones de entidades

En el siguiente `get-entities` ejemplo, se obtiene una definición de un modelo de dispositivo.

```
aws iotthingsgraph get-entities \  
  --ids "urn:tdm:aws/examples:DeviceModel:MotionSensor"
```

Salida:

```
{  
  "descriptions": [  
    {  
      "id": "urn:tdm:aws/examples:DeviceModel:MotionSensor",  
      "type": "DEVICE_MODEL",  
      "createdAt": 1559256190.599,  
    }  
  ]  
}
```



```
    "definition": {
      "language": "GraphQL",
      "text": "##\n# Specification of motion sensor devices interface.\n##
\n#type MotionSensor @deviceModel(id: \"urn:tdm:aws/examples:deviceModel:MotionSensor
\", \n
  capability: \"urn:tdm:aws/examples:capability:MotionSensorCapability\")
  {ignore:void}"
    }
  }
]
```

Para obtener más información, consulte [Creación y carga de modelos](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [GetEntities](#) la Referencia de AWS CLI comandos.

## get-flow-template-revisions

En el siguiente ejemplo de código, se muestra cómo usar `get-flow-template-revisions`.

### AWS CLI

Para obtener información de revisión sobre un flujo

El siguiente `get-flow-template-revisions` ejemplo obtiene información de revisión sobre un flujo (flujo de trabajo).

```
aws iotthingsgraph get-flow-template-revisions \
  --id urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow
```

Salida:

```
{
  "summaries": [
    {
      "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
      "revisionNumber": 1,
      "createdAt": 1559247540.292
    }
  ]
}
```

Para obtener más información, consulte Cómo [trabajar con flujos](#) en la guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [GetFlowTemplateRevisions](#) la Referencia de AWS CLI comandos.

## get-flow-template

En el siguiente ejemplo de código, se muestra cómo usar `get-flow-template`.

### AWS CLI

Para obtener una definición de flujo

El siguiente `get-flow-template` ejemplo obtiene una definición de un flujo (flujo de trabajo).

```
aws iotthingsgraph get-flow-template \
  --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow"
```

Salida:

```
{
  "description": {
    "summary": {
      "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
      "revisionNumber": 1,
      "createdAt": 1559247540.292
    },
    "definition": {
      "language": "GRAPHQL",
      "text": "{\nquery MyFlow($camera: string!, $screen: string!)
@workflowType(id: \"urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow\")
@annotation(type: \"tgc:FlowEvent\", id: \"sledged790c1b2bcd949e09da0c9bfc077f79d
\", x: 1586, y: 653) @triggers(definition: \"{MotionSensor(description:
\\\"\\\"\\\" @position(x: 1045, y: 635.6666564941406) {\n  condition(expr:
\\\"devices[name == \\\"\\\"\\\"motionSensor\\\"\\\"\\\"].events[name == \\\"
\\\"\\\"StateChanged\\\"\\\"\\\"].lastEvent\\\"\\\"))\n  action(expr: \\\"\\\"\\\"\\\"\\\"
\n}}\") {\n  variables {\n    cameraResult @property(id: \"urn:tdm:aws/
examples:property:CameraStateProperty\")\n  }\n  steps {\n    step(name: \"Camera
\", outEvent: [\"sledged790c1b2bcd949e09da0c9bfc077f79d\"] @position(x: 1377,
y: 638.6666564941406) {\n      DeviceActivity(deviceModel: \"urn:tdm:aws/
examples:deviceModel:Camera\", out: \"cameraResult\", deviceId: \"${camera}\")
```

```
{
  "capture": {
    "step": {
      "name": "Screen",
      "inEvent": [
        "sledged790c1b2bcd949e09da0c9bfc077f79d"
      ],
      "position": {
        "x": 1675.6666870117188,
        "y": 637.9999847412109
      },
      "DeviceActivity": {
        "deviceModel": "urn:tdm:aws/examples:deviceModel:Screen",
        "deviceId": "${screen}"
      },
      "display": {
        "imageUrl": "${cameraResult.lastClickedImage}"
      }
    }
  },
  "validatedNamespaceVersion": 5
}
```

Para obtener más información, consulte Cómo [trabajar con flujos](#) en la guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [GetFlowTemplate](#) la Referencia de AWS CLI comandos.

## get-namespace-deletion-status

En el siguiente ejemplo de código, se muestra cómo usar `get-namespace-deletion-status`.

### AWS CLI

Para obtener el estado de la tarea de eliminación del espacio de nombres

En el siguiente `get-namespace-deletion-status` ejemplo, se obtiene el estado de la tarea de eliminación del espacio de nombres.

```
aws iotthingsgraph get-namespace-deletion-status
```

Salida:

```
{
  "namespaceArn": "arn:aws:iotthingsgraph:us-west-2:123456789012",
  "namespaceName": "us-west-2/123456789012/default",
  "status": "SUCCEEDED "
}
```

Para obtener más información, consulte los [espacios de nombres](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte la Referencia [GetNamespaceDeletionStatus](#) de AWS CLI comandos.

## get-system-instance

En el siguiente ejemplo de código, se muestra cómo usar `get-system-instance`.

### AWS CLI

Para obtener una instancia del sistema

En el siguiente `get-system-instance` ejemplo, se obtiene una definición de una instancia del sistema.

```
aws iotthingsgraph get-system-instance \
  --id "urn:tdm:us-west-2/123456789012/default:Deployment:Room218"
```

Salida:

```
{
  "description": {
    "summary": {
      "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room218",
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/default/Room218",
      "status": "NOT_DEPLOYED",
      "target": "CLOUD",
      "createdAt": 1559249315.208,
      "updatedAt": 1559249315.208
    },
    "definition": {
      "language": "GRAPHQL",
      "text": "{\r\nquery Room218 @deployment(id: \"urn:tdm:us-west-2/123456789012/default:Deployment:Room218\", systemId: \"urn:tdm:us-west-2/123456789012/default:System:SecurityFlow\") {\r\n  motionSensor(deviceId: \"MotionSensorName\")\r\n  screen(deviceId: \"ScreenName\")\r\n  camera(deviceId: \"CameraName\") \r\n  triggers {MotionEventTrigger(description: \"a trigger\") { \r\n    condition(expr: \"devices[name == 'motionSensor'].events[name == 'StateChanged'].lastEvent\") \r\n    action(expr: \"ThingsGraph.startFlow('SecurityFlow', bindings[name == 'camera'].deviceId, bindings[name == 'screen'].deviceId))\r\n  }\r\n  }\r\n  }\r\n  }\r\n  }"}"
    },
    "metricsConfiguration": {
      "cloudMetricEnabled": false
    },
    "validatedNamespaceVersion": 5,
  }
}
```

```
    "flowActionsRoleArn": "arn:aws:iam::123456789012:role/ThingsGraphRole"
  }
}
```

Para obtener más información, consulte [Trabajar con configuraciones de sistemas y flujos](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [GetSystemInstance](#) la Referencia de AWS CLI comandos.

## get-system-template-revisions

En el siguiente ejemplo de código, se muestra cómo usar `get-system-template-revisions`.

### AWS CLI

Para obtener información sobre las revisiones de un sistema

En el siguiente `get-system-template-revisions` ejemplo, se obtiene información de revisión sobre un sistema.

```
aws iotthingsgraph get-system-template-revisions \
  --id "urn:tdm:us-west-2/123456789012/default:System:MySystem"
```

Salida:

```
{
  "summaries": [
    {
      "id": "urn:tdm:us-west-2/123456789012/default:System:MySystem",
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/MySystem",
      "revisionNumber": 1,
      "createdAt": 1559247540.656
    }
  ]
}
```

Para obtener más información, consulte [Trabajar con configuraciones de sistemas y flujos](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [GetSystemTemplateRevisions](#) la Referencia de AWS CLI comandos.

## get-system-template

En el siguiente ejemplo de código, se muestra cómo usar `get-system-template`.

### AWS CLI

Para obtener un sistema

En el siguiente `get-system-template` ejemplo se obtiene una definición de un sistema.

```
aws iotthingsgraph get-system-template \
  --id "urn:tdm:us-west-2/123456789012/default:System:MySystem"
```

Salida:

```
{
  "description": {
    "summary": {
      "id": "urn:tdm:us-west-2/123456789012/default:System:MySystem",
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/MyFlow",
      "revisionNumber": 1,
      "createdAt": 1559247540.656
    },
    "definition": {
      "language": "GRAPHQL",
      "text": "{\n  type MySystem @systemType(id: \"urn:tdm:us-west-2/123456789012/default:System:MySystem\", description: \"\") {\n    camera: Camera @thing(id: \"urn:tdm:aws/examples:deviceModel:Camera\")\n    screen: Screen @thing(id: \"urn:tdm:aws/examples:deviceModel:Screen\")\n    motionSensor: MotionSensor @thing(id: \"urn:tdm:aws/examples:deviceModel:MotionSensor\")\n    MyFlow: MyFlow @workflow(id: \"urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow\")\n  }\n}"
    },
    "validatedNamespaceVersion": 5
  }
}
```

Para obtener más información, consulte [Trabajar con configuraciones de sistemas y flujos](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [GetSystemTemplate](#) la Referencia de AWS CLI comandos.

## get-upload-status

En el siguiente ejemplo de código, se muestra cómo usar `get-upload-status`.

### AWS CLI

Para obtener el estado de su entidad, suba

En el siguiente `get-upload-status` ejemplo, se obtiene el estado de la operación de carga de la entidad. El valor de `MyUploadId` es el valor de ID devuelto por la `upload-entity-definitions` operación.

```
aws iotthingsgraph get-upload-status \  
  --upload-id "MyUploadId"
```

Salida:

```
{  
  "namespaceName": "us-west-2/123456789012/default",  
  "namespaceVersion": 5,  
  "uploadId": "f6294f1e-b109-4bbe-9073-f451a2dda2da",  
  "uploadStatus": "SUCCEEDED"  
}
```

Para obtener más información, consulte [Modelado de entidades](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [GetUploadStatus](#) la Referencia de AWS CLI comandos.

## list-flow-execution-messages

En el siguiente ejemplo de código, se muestra cómo usar `list-flow-execution-messages`.

## AWS CLI

Para obtener información sobre los eventos de la ejecución de un flujo

El siguiente `list-flow-execution-messages` ejemplo obtiene información sobre los eventos de una ejecución de flujo.

```
aws iotthingsgraph list-flow-execution-messages \
  --flow-execution-id "urn:tdm:us-west-2/123456789012/
  default:Workflow:SecurityFlow_2019-05-11T19:39:55.317Z_MotionSensor_69b151ad-
  a611-42f5-ac21-fe537f9868ad"
```

Salida:

```
{
  "messages": [
    {
      "eventType": "EXECUTION_STARTED",
      "messageId": "f6294f1e-b109-4bbe-9073-f451a2dda2da",
      "payload": "Flow execution started",
      "timestamp": 1559247540.656
    }
  ]
}
```

Para obtener más información, consulte Cómo [trabajar con flujos](#) en la guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [ListFlowExecutionMessages](#) la Referencia de AWS CLI comandos.

## **list-tags-for-resource**

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

## AWS CLI

Para enumerar todas las etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas de un recurso de AWS IoT Things Graph.



```
aws iotthingsgraph list-tags-for-resource \  
  --resource-arn "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/  
default/Room218"
```

Salida:

```
{  
  "tags": [  
    {  
      "key": "Type",  
      "value": "Residential"  
    }  
  ]  
}
```

Para obtener más información, consulte Cómo [etiquetar los recursos de gráficos de AWS IoT Things](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## search-entities

En el siguiente ejemplo de código, se muestra cómo usar search-entities.

AWS CLI

Para buscar entidades

En el siguiente search-entities ejemplo, se buscan todas las entidades del tipoEVENT.

```
aws iotthingsgraph search-entities \  
  --entity-types "EVENT"
```

Salida:

```
{  
  "descriptions": [  
    {  
      "id": "urn:tdm:aws/examples:Event:MotionSensorEvent",  
      "type": "EVENT",  
      "definition": {
```

```

        "language": "GRAPHQL",
        "text": "##\n# Description of events emitted by motion
sensor.\n##\n\ntype MotionSensorEvent @eventType(id: \"urn:tdm:aws/
examples:event:MotionSensorEvent\", \n          payload: \"urn:tdm:aws/
examples:property:MotionSensorStateProperty\") {ignore:void}"
      }
    },
    {
      "id": "urn:tdm:us-west-2/123456789012/
default:Event:CameraClickedEventV2",
      "type": "EVENT",
      "definition": {
        "language": "GRAPHQL",
        "text": "type CameraClickedEventV2 @eventType(id: \"urn:tdm:us-
west-2/123456789012/default:event:CameraClickedEventV2\", \r\npayload:
\"urn:tdm:aws:Property:Boolean\") {ignore:void}"
      }
    },
    {
      "id": "urn:tdm:us-west-2/123456789012/
default:Event:MotionSensorEventV2",
      "type": "EVENT",
      "definition": {
        "language": "GRAPHQL",
        "text": "# Event emitted by the motion sensor.\r\n\ntype
MotionSensorEventV2 @eventType(id: \"urn:tdm:us-west-2/123456789012/
default:event:MotionSensorEventV2\", \r\npayload: \"urn:tdm:us-west-2/123456789012/
default:property:MotionSensorStateProperty2\") {ignore:void}"
      }
    }
  ],
  "nextToken": "urn:tdm:us-west-2/123456789012/default:Event:MotionSensorEventV2"
}

```

Para obtener más información, consulte la [referencia del modelo de datos de AWS IoT Things Graph](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [SearchEntities](#) la Referencia de AWS CLI comandos.

## search-flow-executions

En el siguiente ejemplo de código, se muestra cómo usar `search-flow-executions`.

## AWS CLI

Para buscar ejecuciones de flujos

En el siguiente `search-flow-executions` ejemplo, se buscan todas las ejecuciones de un flujo en una instancia del sistema especificada.

```
aws iotthingsgraph search-flow-executions \  
  --system-instance-id "urn:tdm:us-west-2/123456789012/default:Deployment:Room218"
```

Salida:

```
{  
  "summaries": [  
    {  
      "createdAt": 1559247540.656,  
      "flowExecutionId": "f6294f1e-b109-4bbe-9073-f451a2dda2da",  
      "flowTemplateId": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",  
      "status": "RUNNING ",  
      "systemInstanceId": "urn:tdm:us-west-2/123456789012/  
default:System:MySystem",  
      "updatedAt": 1559247540.656  
    }  
  ]  
}
```

Para obtener más información, consulte [Trabajar con configuraciones de sistemas y flujos](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [SearchFlowExecutions](#) la Referencia de AWS CLI comandos.

## search-flow-templates

En el siguiente ejemplo de código, se muestra cómo usar `search-flow-templates`.

### AWS CLI

Para buscar flujos (o flujos de trabajo)

El siguiente `search-flow-templates` ejemplo busca todos los flujos (flujos de trabajo) que contienen el modelo de dispositivo de cámara.

```
aws iotthingsgraph search-flow-templates \  
  --filters name="DEVICE_MODEL_ID",value="urn:tdm:aws/examples:DeviceModel:Camera"
```

Salida:

```
{  
  "summaries": [  
    {  
      "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",  
      "revisionNumber": 1,  
      "createdAt": 1559247540.292  
    },  
    {  
      "id": "urn:tdm:us-west-2/123456789012/default:Workflow:SecurityFlow",  
      "revisionNumber": 3,  
      "createdAt": 1548283099.27  
    }  
  ]  
}
```

Para obtener más información, consulte Cómo [trabajar con flujos](#) en la guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [SearchFlowTemplates](#) la Referencia de AWS CLI comandos.

## search-system-instances

En el siguiente ejemplo de código, se muestra cómo usar `search-system-instances`.

AWS CLI

Para buscar instancias del sistema

En el siguiente `search-system-instances` ejemplo, se buscan todas las instancias del sistema que contienen el sistema especificado.

```
aws iotthingsgraph search-system-instances \  
  --filters name="SYSTEM_TEMPLATE_ID",value="urn:tdm:us-west-2/123456789012/  
  default:System:SecurityFlow"
```

**Salida:**

```
{
  "summaries": [
    {
      "id": "urn:tdm:us-west-2/123456789012/
default:Deployment:DeploymentForSample",
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/DeploymentForSample",
      "status": "NOT_DEPLOYED",
      "target": "GREENGRASS",
      "greengrassGroupName": "ThingsGraphGrnGr",
      "createdAt": 1555716314.707,
      "updatedAt": 1555716314.707
    },
    {
      "id": "urn:tdm:us-west-2/123456789012/
default:Deployment:MockDeployment",
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/MockDeployment",
      "status": "DELETED_IN_TARGET",
      "target": "GREENGRASS",
      "greengrassGroupName": "ThingsGraphGrnGr",
      "createdAt": 1549416462.049,
      "updatedAt": 1549416722.361,
      "greengrassGroupId": "01d04b07-2a51-467f-9d03-0c90b3cdcaaf",
      "greengrassGroupVersionId": "7365aed7-2d3e-4d13-aad8-75443d45eb05"
    },
    {
      "id": "urn:tdm:us-west-2/123456789012/
default:Deployment:MockDeployment2",
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/MockDeployment2",
      "status": "DEPLOYED_IN_TARGET",
      "target": "GREENGRASS",
      "greengrassGroupName": "ThingsGraphGrnGr",
      "createdAt": 1549572385.774,
      "updatedAt": 1549572418.408,
      "greengrassGroupId": "01d04b07-2a51-467f-9d03-0c90b3cdcaaf",
      "greengrassGroupVersionId": "bfa70ab3-2bf7-409c-a4d4-bc8328ae5b86"
    },
    {
      "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room215",
```

```

        "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/Room215",
        "status": "NOT_DEPLOYED",
        "target": "GREENGRASS",
        "greengrassGroupName": "ThingsGraphGG",
        "createdAt": 1547056918.413,
        "updatedAt": 1547056918.413
    },
    {
        "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room218",
        "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/Room218",
        "status": "NOT_DEPLOYED",
        "target": "CLOUD",
        "createdAt": 1559249315.208,
        "updatedAt": 1559249315.208
    }
]
}

```

Para obtener más información, consulte [Trabajar con configuraciones de sistemas y flujos](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [SearchSystemInstances](#) la Referencia de AWS CLI comandos.

## search-system-templates

En el siguiente ejemplo de código, se muestra cómo usar `search-system-templates`.

### AWS CLI

Para buscar un sistema

En el siguiente `search-system-templates` ejemplo, se buscan todos los sistemas que contienen el flujo especificado.

```

aws iotthingsgraph search-system-templates \
  --filters name="FLOW_TEMPLATE_ID",value="urn:tdm:us-west-2/123456789012/
default:Workflow:SecurityFlow"

```

Salida:

```
{
  "summaries": [
    {
      "id": "urn:tdm:us-west-2/123456789012/default:System:SecurityFlow",
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/SecurityFlow",
      "revisionNumber": 1,
      "createdAt": 1548283099.433
    }
  ]
}
```

Para obtener más información, consulte Cómo [trabajar con flujos](#) en la guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [SearchSystemTemplates](#) la Referencia de AWS CLI comandos.

## search-things

En el siguiente ejemplo de código, se muestra cómo usar search-things.

### AWS CLI

Para buscar elementos asociados a dispositivos y modelos de dispositivos

En el siguiente search-things ejemplo, se buscan todos los elementos asociados al MotionSensor dispositivo HCSR5 01.

```
aws iotthingsgraph search-things \
  --entity-id "urn:tdm:aws/examples:Device:HCSR501MotionSensor"
```

Salida:

```
{
  "things": [
    {
      "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MotionSensor1",
      "thingName": "MotionSensor1"
    },
    {
```

```
        "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/TG_MS",
        "thingName": "TG_MS"
    }
]
}
```

Para obtener más información, consulte [Creación y carga de modelos](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [SearchThings](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para crear una etiqueta para un recurso

En el siguiente `tag-resource` ejemplo, se crea una etiqueta para el recurso especificado.

```
aws iotthingsgraph tag-resource \
  --resource-arn "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
  default/Room218" \
  --tags key="Type",value="Residential"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Cómo [etiquetar los recursos de gráficos de AWS IoT Things](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## undeploy-system-instance

En el siguiente ejemplo de código, se muestra cómo usar `undeploy-system-instance`.

### AWS CLI

Para anular el despliegue de una instancia del sistema de su destino



En el siguiente `undeploy-system-instance` ejemplo, se elimina una instancia del sistema de su destino.

```
aws iotthingsgraph undeploy-system-instance \  
  --id "urn:tdm:us-west-2/123456789012/default:Deployment:Room215"
```

Salida:

```
{  
  "summary": {  
    "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room215",  
    "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/default/  
Room215",  
    "status": "PENDING_DELETE",  
    "target": "GREENGRASS",  
    "greengrassGroupName": "ThingsGraphGrnGr",  
    "createdAt": 1553189694.255,  
    "updatedAt": 1559344549.601,  
    "greengrassGroupId": "01d04b07-2a51-467f-9d03-0c90b3cdcaaf",  
    "greengrassGroupVersionId": "731b371d-d644-4b67-ac64-3934e99b75d7"  
  }  
}
```

Para obtener más información, consulte [Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [UndeploySystemInstance](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

AWS CLI

Para eliminar una etiqueta de un recurso

En el siguiente `untag-resource` ejemplo, se elimina una etiqueta del recurso especificado.

```
aws iotthingsgraph untag-resource \  
  --resource-arn "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/  
default/Room218" \  
  --tags "tag:Key=Value"
```

```
--tag-keys "Type"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Cómo [etiquetar los recursos de gráficos de AWS IoT Things](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-flow-template

En el siguiente ejemplo de código, se muestra cómo usar `update-flow-template`.

AWS CLI

Para actualizar un flujo

En el siguiente `update-flow-template` ejemplo, se actualiza un flujo (flujo de trabajo). El valor de `MyFlowDefinition` es el GraphQL que modela el flujo.

```
aws iotthingsgraph update-flow-template \  
  --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow" \  
  --definition language=GRAPHQL,text="MyFlowDefinition"
```

Salida:

```
{  
  "summary": {  
    "createdAt": 1559248067.545,  
    "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",  
    "revisionNumber": 2  
  }  
}
```

Para obtener más información, consulte Cómo [trabajar con flujos](#) en la guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [UpdateFlowTemplate](#) la Referencia de AWS CLI comandos.

## update-system-template

En el siguiente ejemplo de código, se muestra cómo usar `update-system-template`.

### AWS CLI

Para actualizar un sistema

En el siguiente `update-system-template` ejemplo, se actualiza un sistema. El valor de `MySystemDefinition` es el GraphQL que modela el sistema.

```
aws iotthingsgraph update-system-template \  
  --id "urn:tdm:us-west-2/123456789012/default:System:MySystem" \  
  --definition language=GRAPHQL,text="MySystemDefinition"
```

Salida:

```
{  
  "summary": {  
    "createdAt": 1559249776.254,  
    "id": "urn:tdm:us-west-2/123456789012/default:System:MySystem",  
    "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/  
MySystem",  
    "revisionNumber": 2  
  }  
}
```

Para obtener más información, consulte [Creación de sistemas](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [UpdateSystemTemplate](#) la Referencia de AWS CLI comandos.

## upload-entity-definitions

En el siguiente ejemplo de código, se muestra cómo usar `upload-entity-definitions`.

### AWS CLI

Para cargar definiciones de entidades

En el siguiente `upload-entity-definitions` ejemplo, se cargan las definiciones de entidades en tu espacio de nombres. El valor de `MyEntityDefinitions` es el GraphQL que modela las entidades.

```
aws iotthingsgraph upload-entity-definitions \  
  --document language=GRAPHQL,text="MyEntityDefinitions"
```

Salida:

```
{  
  "uploadId": "f6294f1e-b109-4bbe-9073-f451a2dda2da"  
}
```

Para obtener más información, consulte [Modelado de entidades](#) en la Guía del usuario de AWS IoT Things Graph.

- Para API obtener más información, consulte [UploadEntityDefinitions](#) la Referencia de AWS CLI comandos.

## AWS IoT Wireless ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS IoT Wireless.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **associate-aws-account-with-partner-account**

En el siguiente ejemplo de código, se muestra cómo usar `associate-aws-account-with-partner-account`.

## AWS CLI

Para asociar una cuenta de socio a tu AWS cuenta

El siguiente `associate-aws-account-with-partner-account` ejemplo asocia las siguientes credenciales de la cuenta de Sidewalk a su AWS cuenta.

```
aws iotwireless associate-aws-account-with-partner-account \  
  --sidewalk  
  AmazonId="12345678901234",AppServerPrivateKey="a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78
```

Salida:

```
{  
  "Sidewalk": {  
    "AmazonId": "12345678901234",  
    "AppServerPrivateKey":  
    "a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"  
  }  
}
```

Para obtener más información, consulte [Amazon Sidewalk Integration for AWS IoT Core](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [AssociateAwsAccountWithPartnerAccount](#) la Referencia de AWS CLI comandos.

## `associate-wireless-device-with-thing`

En el siguiente ejemplo de código, se muestra cómo usar `associate-wireless-device-with-thing`.

## AWS CLI

Para asociar algo a un dispositivo inalámbrico

El siguiente `associate-wireless-device-with-thing` ejemplo asocia un elemento al dispositivo inalámbrico que tiene el identificador especificado.

```
aws iotwireless associate-wireless-device-with-thing \  
  \
```

```
--id "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \  
--thing-arn "arn:aws:iot:us-east-1:123456789012:thing/MyIoTWirelessThing"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Agregar sus puertas de enlace y dispositivos inalámbricos a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [AssociateWirelessDeviceWithThing](#) la Referencia de AWS CLI comandos.

## associate-wireless-gateway-with-certificate

En el siguiente ejemplo de código, se muestra cómo usar `associate-wireless-gateway-with-certificate`.

### AWS CLI

Para asociar el certificado a la puerta de enlace inalámbrica

A continuación, se `associate-wireless-gateway-with-certificate` asocia una puerta de enlace inalámbrica a un certificado.

```
aws iotwireless associate-wireless-gateway-with-certificate \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \  
  --iot-certificate-  
id "a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"
```

Salida:

```
{  
  "IotCertificateId":  
  "a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"  
}
```

Para obtener más información, consulte [Agregar sus puertas de enlace y dispositivos inalámbricos a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [AssociateWirelessGatewayWithCertificate](#) la Referencia de AWS CLI comandos.

## associate-wireless-gateway-with-thing

En el siguiente ejemplo de código, se muestra cómo usar `associate-wireless-gateway-with-thing`.

### AWS CLI

Para asociar algo a una puerta de enlace inalámbrica

El siguiente `associate-wireless-gateway-with-thing` ejemplo asocia un elemento a una puerta de enlace inalámbrica.

```
aws iotwireless associate-wireless-gateway-with-thing \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \  
  --thing-arn "arn:aws:iot:us-east-1:123456789012:thing/MyIoTWirelessThing"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Agregar sus puertas de enlace y dispositivos inalámbricos a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [AssociateWirelessGatewayWithThing](#) la Referencia de AWS CLI comandos.

## create-destination

En el siguiente ejemplo de código, se muestra cómo usar `create-destination`.

### AWS CLI

Para crear un destino inalámbrico de IoT

El siguiente `create-destination` ejemplo crea un destino para asignar un mensaje de dispositivo a una regla de AWS IoT. Antes de ejecutar este comando, debe haber creado un IAM rol que otorgue a AWS IoT Core LoRa WAN los permisos necesarios para enviar datos a la regla de AWS IoT.

```
aws iotwireless create-destination \  
  --name IoTWirelessDestination \  
  --expression-type RuleName \  
  --expression IoTWirelessRule \  
  --
```

```
--role-arn arn:aws:iam::123456789012:role/IoTWirelessDestinationRole
```

Salida:

```
{
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:Destination/
IoTWirelessDestination",
  "Name": "IoTWirelessDestination"
}
```

Para obtener más información, consulte [Añadir destinos a AWS IoT Core LoRa WAN en la Guía para](#) desarrolladores de AWS IoT.

- Para API obtener más información, consulte [CreateDestination](#) la Referencia de AWS CLI comandos.

## create-device-profile

En el siguiente ejemplo de código, se muestra cómo usar create-device-profile.

AWS CLI

Para crear un nuevo perfil de dispositivo

El siguiente create-device-profile ejemplo crea un nuevo perfil de dispositivo inalámbrico de IoT.

```
aws iotwireless create-device-profile
```

Salida:

```
{
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:DeviceProfile/12345678-
a1b2-3c45-67d8-e90fa1b2c34d",
  "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
}
```

Para obtener más información, consulte [Añadir perfiles a AWS IoT Core LoRa WAN en la Guía para](#) desarrolladores de AWS IoT.



- Para API obtener más información, consulte [CreateDeviceProfile](#) la Referencia de AWS CLI comandos.

## **create-service-profile**

En el siguiente ejemplo de código, se muestra cómo usar `create-service-profile`.

AWS CLI

Para crear un nuevo perfil de servicio

El siguiente `create-service-profile` ejemplo crea un nuevo perfil de servicio inalámbrico de IoT.

```
aws iotwireless create-service-profile
```

Salida:

```
{
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:ServiceProfile/12345678-
a1b2-3c45-67d8-e90fa1b2c34d",
  "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
}
```

Para obtener más información, consulte [Añadir perfiles a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [CreateServiceProfile](#) la Referencia de AWS CLI comandos.

## **create-wireless-device**

En el siguiente ejemplo de código, se muestra cómo usar `create-wireless-device`.

AWS CLI

Para crear un dispositivo inalámbrico de IoT

En el siguiente `create-wireless-device` ejemplo, se crea un recurso de dispositivo inalámbrico de este tipo LoRaWAN.

```
aws iotwireless create-wireless-device \  
--cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "Description": "My LoRaWAN wireless device"  
  "DestinationName": "IoTWirelessDestination"  
  "LoRaWAN": {  
    "DeviceProfileId": "ab0c23d3-b001-45ef-6a01-2bc3de4f5333",  
    "ServiceProfileId": "fe98dc76-cd12-001e-2d34-5550432da100",  
    "OtaaV1_1": {  
      "AppKey": "3f4ca100e2fc675ea123f4eb12c4a012",  
      "JoinEui": "b4c231a359bc2e3d",  
      "NwkKey": "01c3f004a2d6efffe32c4eda14bcd2b4"  
    },  
    "DevEui": "ac12efc654d23fc2"  
  },  
  "Name": "SampleIoTWirelessThing"  
  "Type": LoRaWAN  
}
```

Salida:

```
{  
  "Arn": "arn:aws:iotwireless:us-  
east-1:123456789012:WirelessDevice/1ffd32c8-8130-4194-96df-622f072a315f",  
  "Id": "1ffd32c8-8130-4194-96df-622f072a315f"  
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [CreateWirelessDevice](#) la Referencia de AWS CLI comandos.

## **create-wireless-gateway-task-definition**

En el siguiente ejemplo de código, se muestra cómo usar `create-wireless-gateway-task-definition`.

## AWS CLI

Para crear una puerta de enlace inalámbrica, definición de tareas

Lo siguiente crea `create-wireless-gateway-task-definition` automáticamente tareas utilizando esta definición de tarea para todas las puertas de enlace con la versión actual especificada.

```
aws iotwireless create-wireless-gateway-task-definition \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "AutoCreateTasks": true,  
  "Name": "TestAutoUpdate",  
  "Update": {  
    "UpdateDataSource" : "s3://cupsalphagafirmwarebin/station",  
    "UpdateDataRole" : "arn:aws:iam::001234567890:role/SDK_Test_Role",  
    "LoRaWAN" : {  
      "CurrentVersion" : {  
        "PackageVersion" : "1.0.0",  
        "Station" : "2.0.5",  
        "Model" : "linux"  
      },  
      "UpdateVersion" : {  
        "PackageVersion" : "1.0.1",  
        "Station" : "2.0.5",  
        "Model" : "minihub"  
      }  
    }  
  }  
}
```

Salida:

```
{  
  "Id": "b7d3baad-25c7-35e7-a4e1-1683a0d61da9"  
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [CreateWirelessGatewayTaskDefinition](#) la Referencia de AWS CLI comandos.

## create-wireless-gateway-task

En el siguiente ejemplo de código, se muestra cómo usar `create-wireless-gateway-task`.

### AWS CLI

Para crear la tarea de una puerta de enlace inalámbrica

El siguiente `create-wireless-gateway-task` ejemplo crea una tarea para una puerta de enlace inalámbrica.

```
aws iotwireless create-wireless-gateway-task \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \  
  --wireless-gateway-task-definition-id "aa000102-0304-b0cd-ef56-a1b23cde456a"
```

Salida:

```
{  
  "WirelessGatewayTaskDefinitionId": "aa204003-0604-30fb-ac82-a4f95aaf450a",  
  "Status": "Success"  
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [CreateWirelessGatewayTask](#) la Referencia de AWS CLI comandos.

## create-wireless-gateway

En el siguiente ejemplo de código, se muestra cómo usar `create-wireless-gateway`.

### AWS CLI

Para crear una puerta de enlace inalámbrica

El siguiente `create-wireless-gateway` ejemplo crea una puerta de enlace LoRa WAN para dispositivos inalámbricos.

```
aws iotwireless create-wireless-gateway \  
  --lorawan GatewayEui="a1b2c3d4567890ab",RfRegion="US915" \  
  --name "myFirstLoRaWANGateway" \  
  --description "Using my first LoRaWAN gateway"
```

Salida:

```
{  
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:WirelessGateway/12345678-  
a1b2-3c45-67d8-e90fa1b2c34d",  
  "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d"  
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [CreateWirelessGateway](#) la Referencia de AWS CLI comandos.

## delete-destination

En el siguiente ejemplo de código, se muestra cómo usar delete-destination.

### AWS CLI

Para eliminar un destino inalámbrico de IoT

En el siguiente delete-destination ejemplo, se elimina el recurso de destino inalámbrico con el nombre IoTWirelessDestination que usted creó.

```
aws iotwireless delete-destination \  
  --name "IoTWirelessDestination"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir destinos a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [DeleteDestination](#) la Referencia de AWS CLI comandos.

## delete-device-profile

En el siguiente ejemplo de código, se muestra cómo usar delete-device-profile.

### AWS CLI

Para eliminar un perfil de dispositivo

En el siguiente delete-device-profile ejemplo, se elimina un perfil de dispositivo con el ID especificado que ha creado.

```
aws iotwireless delete-device-profile \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir perfiles a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [DeleteDeviceProfile](#) la Referencia de AWS CLI comandos.

## delete-service-profile

En el siguiente ejemplo de código, se muestra cómo usar delete-service-profile.

### AWS CLI

Para eliminar un perfil de servicio

En el siguiente delete-service-profile ejemplo, se elimina un perfil de servicio con el ID especificado que ha creado.

```
aws iotwireless delete-service-profile \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir perfiles a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [DeleteServiceProfile](#) la Referencia de AWS CLI comandos.

## **delete-wireless-device**

En el siguiente ejemplo de código, se muestra cómo usar `delete-wireless-device`.

### AWS CLI

Para eliminar un dispositivo inalámbrico

En el siguiente `delete-wireless-device` ejemplo, se elimina un dispositivo inalámbrico que tiene el identificador especificado.

```
aws iotwireless delete-wireless-device \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [DeleteWirelessDevice](#) la Referencia de AWS CLI comandos.

## **delete-wireless-gateway-task-definition**

En el siguiente ejemplo de código, se muestra cómo usar `delete-wireless-gateway-task-definition`.

### AWS CLI

Para eliminar una definición de tarea de puerta de enlace inalámbrica

En el siguiente `delete-wireless-gateway-task-definition` ejemplo, se elimina la definición de tarea de puerta de enlace inalámbrica que creó con el siguiente identificador.

```
aws iotwireless delete-wireless-gateway-task-definition \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [DeleteWirelessGatewayTaskDefinition](#) la Referencia de AWS CLI comandos.

## **delete-wireless-gateway-task**

En el siguiente ejemplo de código, se muestra cómo usar `delete-wireless-gateway-task`.

### AWS CLI

Para eliminar una tarea de puerta de enlace inalámbrica

En el siguiente `delete-wireless-gateway-task` ejemplo, se elimina la tarea de puerta de enlace inalámbrica que tiene el ID especificado.

```
aws iotwireless delete-wireless-gateway-task \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [DeleteWirelessGatewayTask](#) la Referencia de AWS CLI comandos.

## **delete-wireless-gateway**

En el siguiente ejemplo de código, se muestra cómo usar `delete-wireless-gateway`.

### AWS CLI

Para eliminar una puerta de enlace inalámbrica

En el siguiente `delete-wireless-gateway` ejemplo, se elimina una puerta de enlace inalámbrica que tiene el ID especificado.

```
aws iotwireless delete-wireless-gateway \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```



```
--id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [DeleteWirelessGateway](#) la Referencia de AWS CLI comandos.

## **disassociate-aws-account-from-partner-account**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-aws-account-from-partner-account`.

### AWS CLI

Para desasociar la cuenta asociada de la AWS cuenta

En el siguiente `disassociate-aws-account-from-partner-account` ejemplo, se desvincula una cuenta asociada de su cuenta actualmente asociada AWS .

```
aws iotwireless disassociate-aws-account-from-partner-account \  
  --partner-account-id "12345678901234" \  
  --partner-type "Sidewalk"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Agregar sus puertas de enlace y dispositivos inalámbricos a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [DisassociateAwsAccountFromPartnerAccount](#) la Referencia de AWS CLI comandos.

## **disassociate-wireless-device-from-thing**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-wireless-device-from-thing`.

### AWS CLI

Para desasociar la cosa del dispositivo inalámbrico

En el siguiente `disassociate-wireless-device-from-thing` ejemplo, se desvincula un dispositivo inalámbrico del dispositivo que está asociado actualmente.

```
aws iotwireless disassociate-wireless-device-from-thing \  
--id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Agregar sus puertas de enlace y dispositivos inalámbricos a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [DisassociateWirelessDeviceFromThing](#) la Referencia de AWS CLI comandos.

### **disassociate-wireless-gateway-from-certificate**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-wireless-gateway-from-certificate`.

#### AWS CLI

Para desasociar el certificado de la puerta de enlace inalámbrica

Lo siguiente `disassociate-wireless-gateway-from-certificate` desvincula una puerta de enlace inalámbrica de su certificado actualmente asociado.

```
aws iotwireless disassociate-wireless-gateway-from-certificate \  
--id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Agregar sus puertas de enlace y dispositivos inalámbricos a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [DisassociateWirelessGatewayFromCertificate](#) la Referencia de AWS CLI comandos.

### **disassociate-wireless-gateway-from-thing**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-wireless-gateway-from-thing`.

## AWS CLI

Para desasociarlo de la puerta de enlace inalámbrica

En el siguiente `disassociate-wireless-gateway-from-thing` ejemplo, se desvincula una puerta de enlace inalámbrica de la que está asociada actualmente.

```
aws iotwireless disassociate-wireless-gateway-from-thing \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Agregar sus puertas de enlace y dispositivos inalámbricos a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [DisassociateWirelessGatewayFromThing](#) la Referencia de AWS CLI comandos.

## get-destination

En el siguiente ejemplo de código, se muestra cómo usar `get-destination`.

## AWS CLI

Para obtener información sobre un destino inalámbrico de IoT

En el siguiente `get-destination` ejemplo, se obtiene información sobre el recurso de destino con el nombre `IoTWirelessDestination` que ha creado.

```
aws iotwireless get-destination \  
  --name "IoTWirelessDestination"
```

Salida:

```
{  
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:Destination/  
IoTWirelessDestination",  
  "Name": "IoTWirelessDestination",  
  "Expression": "IoTWirelessRule",  
  "ExpressionType": "RuleName",  
  "RoleArn": "arn:aws:iam::123456789012:role/IoTWirelessDestinationRole"  
}
```

Para obtener más información, consulte [Añadir destinos a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetDestination](#) la Referencia de AWS CLI comandos.

## get-device-profile

En el siguiente ejemplo de código, se muestra cómo usar `get-device-profile`.

### AWS CLI

Para obtener información sobre el perfil de un dispositivo

En el siguiente `get-device-profile` ejemplo, se obtiene información sobre el perfil del dispositivo con el ID especificado que ha creado.

```
aws iotwireless get-device-profile \  
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Salida:

```
{  
  "Arn": "arn:aws:iotwireless:us-east-1:123456789012:DeviceProfile/12345678-  
a1b2-3c45-67d8-e90fa1b2c34d",  
  "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",  
  "LoRaWAN": {  
    "MacVersion": "1.0.3",  
    "MaxDutyCycle": 10,  
    "Supports32BitFCnt": false,  
    "RegParamsRevision": "RP002-1.0.1",  
    "SupportsJoin": true,  
    "RfRegion": "US915",  
    "MaxEirp": 13,  
    "SupportsClassB": false,  
    "SupportsClassC": false  
  }  
}
```

Para obtener más información, consulte [Añadir perfiles a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetDeviceProfile](#) la Referencia de AWS CLI comandos.

## get-partner-account

En el siguiente ejemplo de código, se muestra cómo usar `get-partner-account`.

### AWS CLI

Para obtener la información de la cuenta del socio

En el siguiente `get-partner-account` ejemplo, se obtiene información sobre tu cuenta de Sidewalk que tiene el siguiente identificador.

```
aws iotwireless get-partner-account \  
  --partner-account-id "12345678901234" \  
  --partner-type "Sidewalk"
```

Salida:

```
{  
  "Sidewalk": {  
    "AmazonId": "12345678901234",  
    "Fingerprint":  
    "a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"  
  },  
  "AccountLinked": false  
}
```

Para obtener más información, consulte [Amazon Sidewalk Integration for AWS IoT Core](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [GetPartnerAccount](#) la Referencia de AWS CLI comandos.

## get-service-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `get-service-endpoint`.

### AWS CLI

Para obtener el punto final del servicio

En el siguiente `get-service-endpoint` ejemplo, se obtiene el punto final específico de la cuenta para el CUPS protocolo.

```
aws iotwireless get-service-endpoint
```

Salida:

```
{
  "ServiceType": "CUPS",
  "ServiceEndpoint": "https://A1RMKZ37ACAGOT.cups.lorawan.us-east-1.amazonaws.com:443",
  "ServerTrust": "-----BEGIN CERTIFICATE-----\n
MIIESTCCAzGgAwIBAgITBn+UV4WH6Kx33rJTMlu8mYtWDTANBgkqhkiG9w0BAQsF\n
ADA5MQswCQYDVQQGEwJVUzEPMA0GA1UEChMGQW1hcm9uMRkwFwYDVQQDExBBbWF6\n
b24gUm9vdCBDQSAxMB4XDTE1MTAyMjAwMDAwMFoXDTE1MTAxOTAwMDAwMFowRjEL\n
MAKGA1UEBhMCVVMxZDZANBgNVBAoTBkFtYXpvcjEVMGMGA1UECjMMU2VydMvYIENB\n
IDFCMQ8wDQYDVQQDEwZBbWF6b24wgG9EiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEK\n
AoIBAQCThZn3c68asg3Wuw6MLAd5tES6BIOsMzoKcG5b1PVo+sD0RrMd4f2AbnZ\n
cMzPa43j4wNxp1ty6aUKk4T1qe9B0wKFjwK6zmxXLVYo7bHViXsPlJ6q0MpFge5\n
b1DP+18x+B26A0piiQ0uPkfyDyeR4xQghfj66Yo19V+emU3nazfvpFA+R0z6WoVm\n
B5x+F2pV8xeKNR7u6azDdU5YVX1TawprmxRC1+WsAYmz6qP+z8ArDITC2FMVy2fw\n
0IjK0tEXc/VfmtTFch5+AfgYMGmqqvJ6LcXiAhqG5TI+Dr0RtM88k+8XUBCeQ8IG\n
KuANaL7TiItKZYxK1MMuTJtV9Ib1AgMBAAGjggE7MIIBNzASBgNVHRMBAf8ECDAG\n
AQH/AgEAMA4GA1UdDwEB/wQEAwIBhjAdBgNVHQ4EFgQUWaRmBlKge5WSPK0UByeW\n
dFv5PdAwHwYDVR0jBBgwFoAUhBjMhTTsvAyUlC4IWZzHshB0CggewYIKwYBBQUH\n
AQEEbzBtMC8GCCsGAQUFBzABhiNodHRwOi8vb2Nzc5yb290Y2ExLmFtYXpvcnRy\n
dXN0LmNvbTA6BgggrBgEFBQcwAoYuaHR0cDovL2NydC5yb290Y2ExLmFtYXpvcnRy\n
dXN0LmNvbS9yb290Y2ExLmN1c2A/BgNVHR8EODA2MDSGMAwhi5odHRwOi8vY3Js\n
LnJvb3RjYTEuYW1hem9udHJ1c3QuY29tL3Jvb3RjYTEuY3JsMBMGA1UdIAQMMAow\n
CAYGZ4EMAQIBMA0GCSqGSIb3DQEBCwUAA4IBAQCfkr41u3nPo4FCH0TjY3NT0VI1\n
59Gt/a6ZiqyJEi+752+a1U5y6iAwYfmXss2lJwJFqMp2PphKg5625kXg8kP2CN5t\n
6G7bMQcT8C8xDZntYtd7WPD8UZiRKAJPBXa30/AbwuZe0GaFEQ8ugcYQgSn+IGBI\n
8/LwhBNTZTUVEWuCUUBVV18YtbAiPq3yXqMB480z+ctBWuZSkbvKnodPLamkB2g1\n
upRyzQ7qDn1X8nn8N8V7YJ6y68AtkHcNSRAnpTitxBKjtKPISLMVCx7i4hncxHZS\n
yLyKQXhw2W2Xs0qLeC1etA+jTGDK4UfLeC0SF7FSi8o5LL21L8IzApar2pR/\n
-----END CERTIFICATE-----\n"
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetServiceEndpoint](#) la Referencia de AWS CLI comandos.

## get-service-profile

En el siguiente ejemplo de código, se muestra cómo usar `get-service-profile`.

### AWS CLI

Para obtener información sobre un perfil de servicio

En el siguiente `get-service-profile` ejemplo, se obtiene información sobre el perfil de servicio con el identificador especificado que ha creado.

```
aws iotwireless get-service-profile \
  --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Salida:

```
{
  "Arn": "arn:aws:iotwireless:us-east-1:651419225604:ServiceProfile/538185bb-
d7e7-4b95-96a0-c51aa4a5b9a0",
  "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
  "LoRaWAN": {
    "HrAllowed": false,
    "NwkGeoLoc": false,
    "DrMax": 15,
    "UlBucketSize": 4096,
    "PrAllowed": false,
    "ReportDevStatusBattery": false,
    "DrMin": 0,
    "DlRate": 60,
    "AddGwMetadata": false,
    "ReportDevStatusMargin": false,
    "MinGwDiversity": 1,
    "RaAllowed": false,
    "DlBucketSize": 4096,
    "DevStatusReqFreq": 24,
    "TargetPer": 5,
    "UlRate": 60
  }
}
```

Para obtener más información, consulte [Añadir perfiles a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetServiceProfile](#) la Referencia de AWS CLI comandos.

## get-wireless-device-statistics

En el siguiente ejemplo de código, se muestra cómo usar `get-wireless-device-statistics`.

### AWS CLI

Para obtener información sobre el funcionamiento de un dispositivo inalámbrico

En el siguiente `get-wireless-device-statistics` ejemplo, se obtiene información de funcionamiento sobre un dispositivo inalámbrico.

```
aws iotwireless get-wireless-device-statistics \  
  --wireless-device-id "1ffd32c8-8130-4194-96df-622f072a315f"
```

Salida:

```
{  
  "WirelessDeviceId": "1ffd32c8-8130-4194-96df-622f072a315f"  
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetWirelessDeviceStatistics](#) la Referencia de AWS CLI comandos.

## get-wireless-device

En el siguiente ejemplo de código, se muestra cómo usar `get-wireless-device`.

### AWS CLI

Para obtener información sobre el dispositivo inalámbrico

En el siguiente `get-wireless-device` ejemplo, se enumeran los widgets disponibles en su AWS cuenta.

```
aws iotwireless get-wireless-device \  
  --wireless-device-id "1ffd32c8-8130-4194-96df-622f072a315f"
```



```
--identifier "1ffd32c8-8130-4194-96df-622f072a315f" \
--identifier-type WirelessDeviceID
```

Salida:

```
{
  "Name": "myLoRaWANDevice",
  "ThingArn": "arn:aws:iot:us-east-1:123456789012:thing/44b87eb4-9bce-423d-
b5fc-973f5ecc358b",
  "DestinationName": "IoTWirelessDestination",
  "Id": "1ffd32c8-8130-4194-96df-622f072a315f",
  "ThingName": "44b87eb4-9bce-423d-b5fc-973f5ecc358b",
  "Type": "LoRaWAN",
  "LoRaWAN": {
    "DeviceProfileId": "ab0c23d3-b001-45ef-6a01-2bc3de4f5333",
    "ServiceProfileId": "fe98dc76-cd12-001e-2d34-5550432da100",
    "OtaaV1_1": {
      "AppKey": "3f4ca100e2fc675ea123f4eb12c4a012",
      "JoinEui": "b4c231a359bc2e3d",
      "NwkKey": "01c3f004a2d6efffe32c4eda14bcd2b4"
    },
    "DevEui": "ac12efc654d23fc2"
  },
  "Arn": "arn:aws:iotwireless:us-
east-1:123456789012:WirelessDevice/1ffd32c8-8130-4194-96df-622f072a315f",
  "Description": "My LoRaWAN wireless device"
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetWirelessDevice](#) la Referencia de AWS CLI comandos.

## get-wireless-gateway-certificate

En el siguiente ejemplo de código, se muestra cómo usar `get-wireless-gateway-certificate`.

### AWS CLI

Para obtener el ID de un certificado asociado a una puerta de enlace inalámbrica

En el siguiente `get-wireless-gateway-certificate` ejemplo, se obtiene el identificador del certificado asociado a una puerta de enlace inalámbrica que tiene el identificador especificado.

```
aws iotwireless get-wireless-gateway-certificate \  
  --id "6c44ab31-8b4d-407a-bed3-19b6c7cda551"
```

Salida:

```
{  
  "IotCertificateId":  
    "8ea4aeae3db34c78cce75d9abd830356869ead6972997e0603e5fd032c804b6f"  
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetWirelessGatewayCertificate](#) la Referencia de AWS CLI comandos.

## `get-wireless-gateway-firmware-information`

En el siguiente ejemplo de código, se muestra cómo usar `get-wireless-gateway-firmware-information`.

AWS CLI

Para obtener información sobre el firmware de una puerta de enlace inalámbrica

En el siguiente `get-wireless-gateway-firmware-information` ejemplo, se obtiene la versión del firmware y otra información acerca de una puerta de enlace inalámbrica.

```
aws iotwireless get-wireless-gateway-firmware-information \  
  --id "3039b406-5cc9-4307-925b-9948c63da25b"
```

Salida:

```
{  
  "LoRaWAN" :{  
    "CurrentVersion" :{  
      "PackageVersion" : "1.0.0",  
      "Station" : "2.0.5",
```

```
        "Model" : "linux"
      }
    }
  }
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetWirelessGatewayFirmwareInformation](#) la Referencia de AWS CLI comandos.

## get-wireless-gateway-statistics

En el siguiente ejemplo de código, se muestra cómo usar `get-wireless-gateway-statistics`.

### AWS CLI

Para obtener información sobre el funcionamiento de una puerta de enlace inalámbrica

En el siguiente `get-wireless-gateway-statistics` ejemplo, se obtiene información operativa sobre una puerta de enlace inalámbrica.

```
aws iotwireless get-wireless-gateway-statistics \
  --wireless-gateway-id "3039b406-5cc9-4307-925b-9948c63da25b"
```

Salida:

```
{
  "WirelessGatewayId": "3039b406-5cc9-4307-925b-9948c63da25b"
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetWirelessGatewayStatistics](#) la Referencia de AWS CLI comandos.

## get-wireless-gateway-task-definition

En el siguiente ejemplo de código, se muestra cómo usar `get-wireless-gateway-task-definition`.

## AWS CLI

Para obtener información sobre la definición de tareas de una puerta de enlace inalámbrica

En el siguiente `get-wireless-gateway-task-definition` ejemplo, se obtiene información sobre la definición de la tarea inalámbrica con el identificador especificado.

```
aws iotwireless get-wireless-gateway-task-definition \  
--id "b7d3baad-25c7-35e7-a4e1-1683a0d61da9"
```

Salida:

```
{  
  "AutoCreateTasks": true,  
  "Name": "TestAutoUpdate",  
  "Update": {  
    "UpdateDataSource" : "s3://cupsalphagafirmwarebin/station",  
    "UpdateDataRole" : "arn:aws:iam::001234567890:role/SDK_Test_Role",  
    "LoRaWAN" : {  
      "CurrentVersion" : {  
        "PackageVersion" : "1.0.0",  
        "Station" : "2.0.5",  
        "Model" : "linux"  
      },  
      "UpdateVersion" : {  
        "PackageVersion" : "1.0.1",  
        "Station" : "2.0.5",  
        "Model" : "minihub"  
      }  
    }  
  }  
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetWirelessGatewayTaskDefinition](#) la Referencia de AWS CLI comandos.

### **get-wireless-gateway-task**

En el siguiente ejemplo de código, se muestra cómo usar `get-wireless-gateway-task`.

## AWS CLI

Para obtener información sobre la tarea de puerta de enlace inalámbrica

En el siguiente `get-wireless-gateway-task` ejemplo, se obtiene información sobre la tarea de puerta de enlace inalámbrica con el identificador especificado.

```
aws iotwireless get-wireless-gateway-task \  
  --id "11693a46-6866-47c3-a031-c9a616e7644b"
```

Salida:

```
{  
  "WirelessGatewayId": "6c44ab31-8b4d-407a-bed3-19b6c7cda551",  
  "WirelessGatewayTaskDefinitionId": "b7d3baad-25c7-35e7-a4e1-1683a0d61da9",  
  "Status": "Success"  
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetWirelessGatewayTask](#) la Referencia de AWS CLI comandos.

## `get-wireless-gateway`

En el siguiente ejemplo de código, se muestra cómo usar `get-wireless-gateway`.

## AWS CLI

Para obtener información sobre una puerta de enlace inalámbrica

En el siguiente `get-wireless-gateway` ejemplo, se obtiene información sobre la puerta de enlace inalámbrica `myFirstLoRaWANGateway`.

```
aws iotwireless get-wireless-gateway \  
  --identifier "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \  
  --identifier-type WirelessGatewayId
```

Salida:

```
{
```

```
"Description": "My first LoRaWAN gateway",
"ThingArn": "arn:aws:iot:us-east-1:123456789012:thing/a1b2c3d4-5678-90ab-
cdef-12ab345c67de",
"LoRaWAN": {
  "RfRegion": "US915",
  "GatewayEui": "a1b2c3d4567890ab"
},
"ThingName": "a1b2c3d4-5678-90ab-cdef-12ab345c67de",
"Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
"Arn": "arn:aws:iotwireless:us-
east-1:123456789012:WirelessGateway/6c44ab31-8b4d-407a-bed3-19b6c7cda551",
"Name": "myFirstLoRaWANGateway"
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [GetWirelessGateway](#) la Referencia de AWS CLI comandos.

## list-destinations

En el siguiente ejemplo de código, se muestra cómo usar `list-destinations`.

### AWS CLI

Para ver una lista de los destinos inalámbricos

El siguiente `list-destinations` ejemplo muestra los destinos disponibles registrados en su AWS cuenta.

```
aws iotwireless list-destinations
```

Salida:

```
{
  "DestinationList": [
    {
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:Destination/
IoTWirelessDestination",
      "Name": "IoTWirelessDestination",
      "Expression": "IoTWirelessRule",
```

```

        "Description": "Destination for messages processed using
IoTWirelessRule",
        "RoleArn": "arn:aws:iam::123456789012:role/IoTWirelessDestinationRole"
    },
    {
        "Arn": "arn:aws:iotwireless:us-east-1:123456789012:Destination/
IoTWirelessDestination2",
        "Name": "IoTWirelessDestination2",
        "Expression": "IoTWirelessRule2",
        "RoleArn": "arn:aws:iam::123456789012:role/IoTWirelessDestinationRole"
    }
]
}

```

Para obtener más información, consulte [Añadir destinos a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [ListDestinations](#) la Referencia de AWS CLI comandos.

## list-device-profiles

En el siguiente ejemplo de código, se muestra cómo usar `list-device-profiles`.

### AWS CLI

Para ver una lista de los perfiles de los dispositivos

En el siguiente `list-device-profiles` ejemplo, se enumeran los perfiles de dispositivo disponibles registrados en su AWS cuenta.

```
aws iotwireless list-device-profiles
```

Salida:

```

{
  "DeviceProfileList": [
    {
      "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
      "Arn": "arn:aws:iotwireless:us-
east-1:123456789012:DeviceProfile/12345678-a1b2-3c45-67d8-e90fa1b2c34d"
    },
    {

```

```
        "Id": "a1b2c3d4-5678-90ab-cdef-12ab345c67de",
        "Arn": "arn:aws:iotwireless:us-east-1:123456789012:DeviceProfile/
a1b2c3d4-5678-90ab-cdef-12ab345c67de"
    }
]
}
```

Para obtener más información, consulte [Añadir perfiles a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [ListDeviceProfiles](#) la Referencia de AWS CLI comandos.

## list-partner-accounts

En el siguiente ejemplo de código, se muestra cómo usar `list-partner-accounts`.

### AWS CLI

Para enumerar las cuentas de los socios

En el siguiente `list-partner-accounts` ejemplo, se enumeran las cuentas de socios disponibles asociadas a su AWS cuenta.

```
aws iotwireless list-partner-accounts
```

Salida:

```
{
  "Sidewalk": [
    {
      "AmazonId": "78965678771228",
      "Fingerprint":
"bd96d8ef66dbfd2160eb60e156849e82ad7018b8b73c1ba0b4fc65c32498ee35"
    },
    {
      "AmazonId": "89656787651228",
      "Fingerprint":
"bc5e99e151c07be14be7e6603e4489c53f858b271213a36ebe3370777ba06e9b"
    }
  ]
}
```



Para obtener más información, consulte [Amazon Sidewalk Integration for AWS IoT Core](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListPartnerAccounts](#) la Referencia de AWS CLI comandos.

## **list-service-profiles**

En el siguiente ejemplo de código, se muestra cómo usar `list-service-profiles`.

### AWS CLI

Para ver una lista de los perfiles de servicio

El siguiente `list-service-profiles` ejemplo muestra los perfiles de servicio disponibles registrados en su AWS cuenta.

```
aws iotwireless list-service-profiles
```

Salida:

```
{
  "ServiceProfileList": [
    {
      "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:ServiceProfile/538185bb-d7e7-4b95-96a0-c51aa4a5b9a0"
    },
    {
      "Id": "a1b2c3d4-5678-90ab-cdef-12ab345c67de",
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:ServiceProfile/ea8bc823-5d13-472e-8d26-9550737d8100"
    }
  ]
}
```

Para obtener más información, consulte [Añadir perfiles a AWS IoT Core LoRa WAN en la Guía para](#) desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListServiceProfiles](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas asignadas al recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas asignadas a un recurso de destino inalámbrico.

```
aws iotwireless list-tags-for-resource \
  --resource-arn "arn:aws:iotwireless:us-east-1:123456789012:Destination/
  IoWirelessDestination"
```

Salida:

```
{
  "Tags": [
    {
      "Value": "MyValue",
      "Key": "MyTag"
    }
  ]
}
```

Para obtener más información, consulte [Describa su núcleo de AWS IoT para obtener LoRa WAN recursos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## list-wireless-devices

En el siguiente ejemplo de código, se muestra cómo usar `list-wireless-devices`.

### AWS CLI

Para ver una lista de los dispositivos inalámbricos disponibles

El siguiente `list-wireless-devices` ejemplo muestra los dispositivos inalámbricos disponibles registrados en su AWS cuenta.

```
aws iotwireless list-wireless-devices
```

Salida:

```
{
  "WirelessDeviceList": [
    {
      "Name": "myLoRaWANDevice",
      "DestinationName": "IoTWirelessDestination",
      "Id": "1fffd32c8-8130-4194-96df-622f072a315f",
      "Type": "LoRaWAN",
      "LoRaWAN": {
        "DevEui": "ac12efc654d23fc2"
      },
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:WirelessDevice/1fffd32c8-8130-4194-96df-622f072a315f"
    }
  ]
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [ListWirelessDevices](#) la Referencia de AWS CLI comandos.

## list-wireless-gateway-task-definitions

En el siguiente ejemplo de código, se muestra cómo usar `list-wireless-gateway-task-definitions`.

AWS CLI

Para enumerar las definiciones de tareas de la puerta de enlace inalámbrica

En el siguiente `list-wireless-gateway-task-definitions` ejemplo, se enumeran las definiciones de tareas de puerta de enlace inalámbrica disponibles registradas AWS en su cuenta.

```
aws iotwireless list-wireless-gateway-task-definitions
```

Salida:

```
{
  "TaskDefinitions": [
    {
      "Id": "b7d3baad-25c7-35e7-a4e1-1683a0d61da9",
      "LoRaWAN" :
        {
          "CurrentVersion" :{
            "PackageVersion" : "1.0.0",
            "Station" : "2.0.5",
            "Model" : "linux"
          },
          "UpdateVersion" :{
            "PackageVersion" : "1.0.1",
            "Station" : "2.0.5",
            "Model" : "minihub"
          }
        }
    }
  ]
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [ListWirelessGatewayTaskDefinitions](#) la Referencia de AWS CLI comandos.

## **list-wireless-gateways**

En el siguiente ejemplo de código, se muestra cómo usar `list-wireless-gateways`.

AWS CLI

Para ver una lista de las puertas de enlace inalámbricas

En el siguiente `list-wireless-gateways` ejemplo, se enumeran las puertas de enlace inalámbricas disponibles en su AWS cuenta.

```
aws iotwireless list-wireless-gateways
```

Salida:

```
{
  "WirelessGatewayList": [
    {
      "Description": "My first LoRaWAN gateway",
      "LoRaWAN": {
        "RfRegion": "US915",
        "GatewayEui": "dac632ebc01d23e4"
      },
      "Id": "3039b406-5cc9-4307-925b-9948c63da25b",
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:WirelessGateway/3039b406-5cc9-4307-925b-9948c63da25b",
      "Name": "myFirstLoRaWANGateway"
    },
    {
      "Description": "My second LoRaWAN gateway",
      "LoRaWAN": {
        "RfRegion": "US915",
        "GatewayEui": "cda123ffffe92ecd2"
      },
      "Id": "3285bdc7-5a12-4991-84ed-dadca65e342e",
      "Arn": "arn:aws:iotwireless:us-east-1:123456789012:WirelessGateway/3285bdc7-5a12-4991-84ed-dadca65e342e",
      "Name": "mySecondLoRaWANGateway"
    }
  ]
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [ListWirelessGateways](#) la Referencia de AWS CLI comandos.

## send-data-to-wireless-device

En el siguiente ejemplo de código, se muestra cómo usar send-data-to-wireless-device.

### AWS CLI

Para enviar datos al dispositivo inalámbrico

El siguiente `send-data-to-wireless-device` ejemplo envía un marco de datos de aplicación descifrado al dispositivo inalámbrico.

```
aws iotwireless send-data-to-wireless-device \  
  --id "11aa5eae-2f56-4b8e-a023-b28d98494e49" \  
  --transmit-mode "1" \  
  --payload-data "SGVsbG8gVG8gRGV2c2lt" \  
  --wireless-metadata LoRaWAN={FPort=1}
```

Salida:

```
{  
  MessageId: "6011dd36-0043d6eb-0072-0008"  
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [SendDataToWirelessDevice](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para especificar una etiqueta, clave y valor para un recurso

El siguiente `tag-resource` ejemplo etiqueta el destino inalámbrico `IoTWirelessDestination` con la clave `MyTag` y el valor `MyValue`.

```
aws iotwireless tag-resource \  
  --resource-arn "arn:aws:iotwireless:us-east-1:651419225604:Destination/  
IoTWirelessDestination" \  
  --tags Key="MyTag",Value="MyValue"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Describe su núcleo de AWS IoT para obtener LoRa WAN recursos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## test-wireless-device

En el siguiente ejemplo de código, se muestra cómo usar `test-wireless-device`.

### AWS CLI

Para probar el dispositivo inalámbrico

El siguiente `test-wireless-device` ejemplo envía datos de enlace ascendente Hello a un dispositivo con un ID especificado.

```
aws iotwireless test-wireless-device \  
  --id "11aa5eae-2f56-4b8e-a023-b28d98494e49"
```

Salida:

```
{  
  Result: "Test succeeded. one message is sent with payload: hello"  
}
```

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [TestWirelessDevice](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar una o más etiquetas de un recurso

En el siguiente `untag-resource` ejemplo, se quita la etiqueta MyTag y su valor del destino inalámbricoIoTWirelessDestination.

```
aws iotwireless untag-resource \  
  --resource-id "arn:aws:iotwireless:us-east-1:123456789012:wireless-destination:MyDestination"
```

```
--resource-arn "arn:aws:iotwireless:us-east-1:123456789012:Destination/IoWirelessDestination" \  
--tag-keys "MyTag"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Describe su núcleo de AWS IoT para obtener LoRa WAN recursos](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-destination

En el siguiente ejemplo de código, se muestra cómo usar update-destination.

### AWS CLI

Para actualizar las propiedades de un destino

En el siguiente update-destination ejemplo, se actualiza la propiedad de descripción de un destino inalámbrico.

```
aws iotwireless update-destination \  
--name "IoWirelessDestination" \  
--description "Destination for messages processed using IoWirelessRule"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir destinos a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [UpdateDestination](#) la Referencia de AWS CLI comandos.

## update-partner-account

En el siguiente ejemplo de código, se muestra cómo usar update-partner-account.

### AWS CLI

Para actualizar las propiedades de una cuenta de socio



A continuación, se `update-partner-account` actualiza `AppServerPrivateKey` la de la cuenta que tiene el ID especificado.

```
aws iotwireless update-partner-account \  
  --partner-account-id "78965678771228" \  
  --partner-type "Sidewalk" \  
  --sidewalk  
  AppServerPrivateKey="f798ab4899346a88599180fee9e14fa1ada7b6df989425b7c6d2146dd6c815bb"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Amazon Sidewalk Integration for AWS IoT Core](#) en la Guía para desarrolladores de AWS IoT.

- Para API obtener más información, consulte [UpdatePartnerAccount](#) la Referencia de AWS CLI comandos.

## update-wireless-device

En el siguiente ejemplo de código, se muestra cómo usar `update-wireless-device`.

### AWS CLI

Para actualizar las propiedades de un dispositivo inalámbrico

En el siguiente `update-wireless-device` ejemplo, se actualizan las propiedades de un dispositivo inalámbrico registrado en su AWS cuenta.

```
aws iotwireless update-wireless-device \  
  --id "1ffd32c8-8130-4194-96df-622f072a315f" \  
  --destination-name IoWirelessDestination2 \  
  --description "Using my first LoRaWAN device"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [UpdateWirelessDevice](#) la Referencia de AWS CLI comandos.

## update-wireless-gateway

En el siguiente ejemplo de código, se muestra cómo usar `update-wireless-gateway`.

### AWS CLI

Para actualizar la puerta de enlace inalámbrica

El siguiente `update-wireless-gateway` ejemplo actualiza la descripción de la puerta de enlace inalámbrica.

```
aws iotwireless update-wireless-gateway \  
  --id "3285bdc7-5a12-4991-84ed-dadca65e342e" \  
  --description "Using my LoRaWAN gateway"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Conexión de dispositivos y puertas de enlace a AWS IoT Core LoRa WAN en la Guía para desarrolladores de AWS IoT](#).

- Para API obtener más información, consulte [UpdateWirelessGateway](#) la Referencia de AWS CLI comandos.

## IVSEjemplos de Amazon que utilizan AWS CLI

En los siguientes ejemplos de código, se muestra cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con AmazonIVS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### batch-get-channel

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-channel`.

#### AWS CLI

Para obtener información sobre la configuración de varios canales

En el siguiente `batch-get-channel` ejemplo, se muestra información sobre los canales especificados.

```
aws ivs batch-get-channel \
  --arns arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \
  arn:aws:ivs:us-west-2:123456789012:channel/efghEFGHijkl
```

Salida:

```
{
  "channels": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "authorized": false,
      "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
      "insecureIngest": false,
      "latencyMode": "LOW",
      "name": "channel-1",
      "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel-1.abcdEFGH.m3u8",
      "preset": "",
      "playbackRestrictionPolicyArn": "",
      "recordingConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCD12cdEFgh",
      "srt": {
        "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
        "passphrase": "AB1C2defGHijklMNop3PqRstUvwxyzABCDefghh4ijklMN5opqrStuVWXYZAbCDEfghIJ"
      },
      "tags": {},
      "type": "STANDARD"
    },
  ],
}
```

```

    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/efghEFGHijkl",
      "authorized": false,
      "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
      "insecureIngest": true,
      "latencyMode": "LOW",
      "name": "channel-2",
      "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel-2.abcdEFGH.m3u8",
      "preset": "",
      "playbackRestrictionPolicyArn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/ABCDEF34ghIJ",
      "recordingConfigurationArn": "",
      "srt": {
        "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
        "passphrase":
"BA1C2defGHijklMNop3PqQRstUvwxyzABCDEFghh4ijklMN5opqrStuVWxyzABCDEFghIJ"
      },
      "tags": {},
      "type": "STANDARD"
    }
  ]
}

```

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [BatchGetChannel](#) la Referencia de AWS CLI comandos.

## batch-get-stream-key

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-stream-key`.

### AWS CLI

Para obtener información sobre varias claves de transmisión

En el siguiente `batch-get-stream-key` ejemplo, se obtiene información sobre las claves de flujo especificadas.

```

aws ivs batch-get-stream-key \
  --arns arn:aws:ivs:us-west-2:123456789012:stream-key/skSKABCDefgh \

```

```
arn:aws:ivs:us-west-2:123456789012:stream-key/skSKIJKLmnop
```

Salida:

```
{
  "streamKeys": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/skSKABCDefgh",
      "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
      "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "tags": {}
    },
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/skSKIJKLmnop",
      "value": "sk_us-west-2_abcdABCDefgh_567890ghijkl",
      "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "tags": {}
    }
  ]
}
```

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [BatchGetStreamKey](#) la Referencia de AWS CLI comandos.

## batch-start-viewer-session-revocation

En el siguiente ejemplo de código, se muestra cómo usar `batch-start-viewer-session-revocation`.

AWS CLI

Para revocar las sesiones de espectadores para varios pares de canales ARN e ID de espectador

En el siguiente `batch-start-viewer-session-revocation` ejemplo, se revoca la sesión en varios pares de ID de canal e ID de espectador de forma ARN simultánea. La solicitud puede completarse con normalidad, pero devuelve valores en el campo de errores si la persona que llama no tiene permiso para revocar una sesión específica.

```
aws ivs batch-start-viewer-session-revocation \
```

```
--viewer-sessions '[{"channelArn":"arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh1","viewerId":"abcdefg1","viewerSessionVersionsLessThanOrEqualTo":1234567890},
\
{"channelArn":"arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh2","viewerId":"abcdefg2","viewerSessionVersionsLessThanOrEqualTo":1234567890}]'
```

Salida:

```
{
  "errors": [
    {
      "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh1",
      "viewerId": "abcdefg1",
      "code": "403",
      "message": "not authorized",
    },
    {
      "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh2",
      "viewerId": "abcdefg2",
      "code": "403",
      "message": "not authorized",
    }
  ]
}
```

Para obtener más información, consulte [Configuración de canales privados](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte la Referencia [BatchStartViewerSessionRevocation](#) de AWS CLI comandos.

## create-channel

En el siguiente ejemplo de código, se muestra cómo usar `create-channel`.

### AWS CLI

Ejemplo 1: Para crear un canal sin grabación

El siguiente `create-channel` ejemplo crea un canal nuevo y una clave de transmisión asociada para iniciar la transmisión.

```
aws ivs create-channel \  
  --name "test-channel" \  
  --no-insecure-ingest
```

Salida:

```
{  
  "channel": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
    "authorized": false,  
    "name": "test-channel",  
    "latencyMode": "LOW",  
    "playbackRestrictionPolicyArn": "",  
    "recordingConfigurationArn": "",  
    "srt": {  
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",  
      "passphrase":  
"AB1C2defGHijklMNop3PqQRstUvwxyzABCDefghh4ijklMN5opqrStuVWXYZAbCDEfghIJ"  
    },  
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",  
    "insecureIngest": false,  
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/  
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",  
    "preset": "",  
    "tags": {},  
    "type": "STANDARD"  
  },  
  "streamKey": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/g1H2I3j4k5L6",  
    "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",  
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
    "tags": {}  
  }  
}
```

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

Ejemplo 2: Para crear un canal con la grabación habilitada, utilice el RecordingConfiguration recurso especificado por su ARN

El siguiente `create-channel` ejemplo crea un canal nuevo y una clave de transmisión asociada para iniciar la transmisión y configura la grabación para el canal.

```
aws ivs create-channel \
  --name test-channel-with-recording \
  --insecure-ingest \
  --recording-configuration-arn "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCD12cdEFgh"
```

Salida:

```
{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "name": "test-channel-with-recording",
    "latencyMode": "LOW",
    "type": "STANDARD",
    "playbackRestrictionPolicyArn": "",
    "recordingConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCD12cdEFgh",
    "srt": {
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
      "passphrase":
"BA1C2defGHijklMNop3PqRstUvwxyzABCDEFGHIJKlmn5opqrStuVWxyzABCDEFGHIJ"
    },
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": true,
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "preset": "",
    "authorized": false,
    "tags": {},
    "type": "STANDARD"
  },
  "streamKey": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/abcdABCDefgh",
    "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "tags": {}
  }
}
```



Para obtener más información, consulte [Grabar en Amazon S3](#) en la Guía del usuario IVS de baja latencia.

Ejemplo 3: Para crear un canal con una política de restricción de reproducción especificada por su ARN

El siguiente `create-channel` ejemplo crea un canal nuevo y una clave de transmisión asociada para iniciar la transmisión, y configura una política de restricción de reproducción para el canal.

```
aws ivs create-channel \
  --name test-channel-with-playback-restriction-policy \
  --insecure-ingest \
  --playback-restriction-policy-arn "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ"
```

Salida:

```
{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "name": "test-channel-with-playback-restriction-policy",
    "latencyMode": "LOW",
    "type": "STANDARD",
    "playbackRestrictionPolicyArn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ",
    "recordingConfigurationArn": "",
    "srt": {
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
      "passphrase":
"AB1C2edfGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijk1MN5opqrStuVWxyzAbCDEfghIJ"
    },
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": true,
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "preset": "",
    "authorized": false,
    "tags": {},
    "type": "STANDARD"
  },
  "streamKey": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/abcdABCDefgh",
    "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
  }
}
```

```

    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "tags": {}
  }
}

```

Para obtener más información, consulte [Contenido no deseado y espectadores](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte la Referencia [CreateChannel](#) de AWS CLI comandos.

## create-playback-restriction-policy

En el siguiente ejemplo de código, se muestra cómo usar `create-playback-restriction-policy`.

### AWS CLI

Para crear una política de restricción de reproducción

El siguiente `create-playback-restriction-policy` ejemplo crea una nueva política de restricción de reproducción.

```

aws ivs create-playback-restriction-policy \
  --name "test-playback-restriction-policy" \
  --enable-strict-origin-enforcement \
  --tags "key1=value1, key2=value2" \
  --allowed-countries US MX \
  --allowed-origins https://www.website1.com https://www.website2.com

```

Salida:

```

{
  "playbackRestrictionPolicy": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/ABCdef34ghIJ",
    "allowedCountries": [
      "US",
      "MX"
    ],
    "allowedOrigins": [

```

```

        "https://www.website1.com",
        "https://www.website2.com"
    ],
    "enableStrictOriginEnforcement": true,
    "name": "test-playback-restriction-policy",
    "tags": {
        "key1": "value1",
        "key2": "value2"
    }
}
}

```

Para obtener más información, consulte [Contenido no deseado y espectadores](#) en la Guía del usuario de IVSbaja latencia.

- Para API obtener más información, consulte la Referencia [CreatePlaybackRestrictionPolicy](#) de AWS CLI comandos.

## create-recording-configuration

En el siguiente ejemplo de código, se muestra cómo usar `create-recording-configuration`.

### AWS CLI

Para crear un `RecordingConfiguration` recurso

El siguiente `create-recording-configuration` ejemplo crea un `RecordingConfiguration` recurso para permitir la grabación en Amazon S3.

```

aws ivs create-recording-configuration \
  --name "test-recording-config" \
  --recording-reconnect-window-seconds 60 \
  --tags "key1=value1, key2=value2" \
  --rendition-configuration renditionSelection="CUSTOM",renditions="HD" \
  --thumbnail-configuration
recordingMode="INTERVAL",targetIntervalSeconds=1,storage="LATEST",resolution="LOWEST_RESOLUTION" \
  --destination-configuration s3={bucketName=demo-recording-bucket}

```

Salida:

```
{
```

```
"recordingConfiguration": {
  "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
ABCdef34ghIJ",
  "name": "test-recording-config",
  "destinationConfiguration": {
    "s3": {
      "bucketName": "demo-recording-bucket"
    }
  },
  "state": "CREATING",
  "tags": {
    "key1": "value1",
    "key2": "value2"
  },
  "thumbnailConfiguration": {
    "recordingMode": "INTERVAL",
    "targetIntervalSeconds": 1,
    "resolution": "LOWEST_RESOLUTION",
    "storage": [
      "LATEST"
    ]
  },
  "recordingReconnectWindowSeconds": 60,
  "renditionConfiguration": {
    "renditionSelection": "CUSTOM",
    "renditions": [
      "HD"
    ]
  }
}
```

Para obtener más información, consulte [Grabar en Amazon S3](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [CreateRecordingConfiguration](#) la Referencia de AWS CLI comandos.

## **create-stream-key**

En el siguiente ejemplo de código, se muestra cómo usar `create-stream-key`.

## AWS CLI

Para crear una clave de flujo

En el siguiente `create-stream-key` ejemplo, se crea una clave de flujo para un determinado ARN (Amazon Resource Name).

```
aws ivs create-stream-key \  
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Salida:

```
{  
  "streamKey": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/abcdABCDefgh",  
    "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",  
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
    "tags": {}  
  }  
}
```

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [CreateStreamKey](#) la Referencia de AWS CLI comandos.

## `delete-channel`

En el siguiente ejemplo de código, se muestra cómo usar `delete-channel`.

## AWS CLI

Para eliminar un canal y sus claves de transmisión asociadas

En el siguiente `delete-channel` ejemplo, se elimina el canal con el ARN (Amazon Resource Name) especificado.

```
aws ivs delete-channel \  
  --arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [DeleteChannel](#) en la Referencia de AWS CLI comandos.

## **delete-playback-key-pair**

En el siguiente ejemplo de código, se muestra cómo usar `delete-playback-key-pair`.

### AWS CLI

Para eliminar un par de claves de reproducción especificado

En el siguiente `delete-playback-key-pair` ejemplo, se devuelve la huella digital del par de claves especificado.

```
aws ivs delete-playback-key-pair \  
  --arn arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de canales privados](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [DeletePlaybackKeyPair](#) en la Referencia de AWS CLI comandos.

## **delete-playback-restriction-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-playback-restriction-policy`.

### AWS CLI

Para eliminar una política de restricción de reproducción

En el siguiente `delete-playback-restriction-policy` ejemplo, se elimina la política de restricción de reproducción con la política especificada (ARN Amazon Resource Name).

```
aws ivs delete-playback-restriction-policy \  
  --arn arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/abcd1234efgh
```

```
--arn "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Contenido no deseado y espectadores](#) en la Guía del IVS usuario de baja latencia.

- Para API obtener más información, consulte la Referencia [DeletePlaybackRestrictionPolicy](#) de AWS CLI comandos.

## delete-recording-configuration

En el siguiente ejemplo de código, se muestra cómo usar delete-recording-configuration.

AWS CLI

Para eliminar el RecordingConfiguration recurso especificado por su ARN

En el siguiente delete-recording-configuration ejemplo, se elimina el RecordingConfiguration recurso con lo especificadoARN.

```
aws ivs delete-recording-configuration \  
--arn "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABcdef34ghIJ"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Grabar en Amazon S3](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [DeleteRecordingConfiguration](#) la Referencia de AWS CLI comandos.

## delete-stream-key

En el siguiente ejemplo de código, se muestra cómo usar delete-stream-key.

AWS CLI

Para eliminar una clave de flujo

En el siguiente delete-stream-key ejemplo, se elimina la clave de transmisión de un elemento especificado ARN (Amazon Resource Name), por lo que ya no se puede usar para transmitir.

```
aws ivs delete-stream-key \  
  --arn arn:aws:ivs:us-west-2:123456789012:stream-key/g1H2I3j4k5L6
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [DeleteStreamKey](#) la Referencia de AWS CLI comandos.

## get-channel

En el siguiente ejemplo de código, se muestra cómo usar `get-channel`.

### AWS CLI

Para obtener la información de configuración de un canal

En el siguiente `get-channel` ejemplo, se obtiene la configuración de canal de un canal específico ARN (Amazon Resource Name).

```
aws ivs get-channel \  
  --arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Salida:

```
{  
  "channel": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
    "name": "channel-1",  
    "latencyMode": "LOW",  
    "type": "STANDARD",  
    "playbackRestrictionPolicyArn": "",  
    "preset": "",  
    "recordingConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:recording-  
configuration/ABCD12cdEFgh",  
    "srt": {  
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",  
      "passphrase":  
"AB1C2defGHijklMN03PqQRstUvwxyzABCDefghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"    }  
  }  
}
```



```
    },
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": false,
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "tags": {}
  }
}
```

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [GetChannel](#) la Referencia de AWS CLI comandos.

## get-playback-key-pair

En el siguiente ejemplo de código, se muestra cómo usar `get-playback-key-pair`.

### AWS CLI

Para obtener un par de claves de reproducción específico

En el siguiente `get-playback-key-pair` ejemplo, se devuelve la huella digital del par de claves especificado.

```
aws ivs get-playback-key-pair \
  --arn arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh
```

Salida:

```
{
  "keyPair": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh",
    "name": "my-playback-key",
    "fingerprint": "0a:1b:2c:ab:cd:ef:34:56:70:b1:b2:71:01:2a:a3:72",
    "tags": {}
  }
}
```

Para obtener más información, consulte [Configuración de canales privados](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [GetPlaybackKeyPair](#) la Referencia de AWS CLI comandos.

## get-playback-restriction-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-playback-restriction-policy`.

### AWS CLI

Para obtener la información de configuración de una política de restricción de reproducción

El siguiente `get-playback-restriction-policy` ejemplo obtiene la configuración de la política de restricción de reproducción con la política especificada (ARN Amazon Resource Name).

```
aws ivs get-playback-restriction-policy \  
  --arn "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/  
  ABcdef34ghIJ"
```

Salida:

```
{  
  "playbackRestrictionPolicy": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/  
  ABcdef34ghIJ",  
    "allowedCountries": [  
      "US",  
      "MX"  
    ],  
    "allowedOrigins": [  
      "https://www.website1.com",  
      "https://www.website2.com"  
    ],  
    "enableStrictOriginEnforcement": true,  
    "name": "test-playback-restriction-policy",  
    "tags": {  
      "key1": "value1",  
      "key2": "value2"  
    }  
  }  
}
```

Para obtener más información, consulte [Contenido no deseado y espectadores](#) en la Guía del IVS usuario de baja latencia.

- Para API obtener más información, consulte la Referencia [GetPlaybackRestrictionPolicy](#) de AWS CLI comandos.

## get-recording-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-recording-configuration`.

### AWS CLI

Para obtener información sobre un `RecordingConfiguration` recurso

En el siguiente `get-recording-configuration` ejemplo, se obtiene información sobre el `RecordingConfiguration` recurso especificadoARN.

```
aws ivs get-recording-configuration \  
  --arn "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCdef34ghIJ"
```

Salida:

```
{  
  "recordingConfiguration": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/  
ABCdef34ghIJ",  
    "destinationConfiguration": {  
      "s3": {  
        "bucketName": "demo-recording-bucket"  
      }  
    },  
    "name": "test-recording-config",  
    "recordingReconnectWindowSeconds": 60,  
    "state": "ACTIVE",  
    "tags": {  
      "key1" : "value1",  
      "key2" : "value2"  
    },  
    "thumbnailConfiguration": {  
      "recordingMode": "INTERVAL",  
      "targetIntervalSeconds": 1,  
      "resolution": "LOWEST_RESOLUTION",  
      "storage": [  

```

```

        "LATEST"
      ]
    },
    "renditionConfiguration": {
      "renditionSelection": "CUSTOM",
      "renditions": [
        "HD"
      ]
    }
  }
}

```

Para obtener más información, consulte [Grabar en Amazon S3](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [GetRecordingConfiguration](#) la Referencia de AWS CLI comandos.

## get-stream-key

En el siguiente ejemplo de código, se muestra cómo usar `get-stream-key`.

### AWS CLI

Para obtener información sobre una transmisión

En el siguiente `get-stream-key` ejemplo, se obtiene información sobre la clave de flujo especificada.

```

aws ivs get-stream-key \
  --arn arn:aws:ivs:us-west-2:123456789012:stream-key/skSKABCDefgh --region=us-
west-2

```

Salida:

```

{
  "streamKey": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/skSKABCDefgh",
    "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "tags": {}
  }
}

```

```
}
```

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [GetStreamKey](#) la Referencia de AWS CLI comandos.

## get-stream-session

En el siguiente ejemplo de código, se muestra cómo usar `get-stream-session`.

### AWS CLI

Para obtener los metadatos de una transmisión específica

El siguiente `get-stream-session` ejemplo obtiene la configuración de metadatos para el canal especificado ARN (Amazon Resource Name) y la transmisión especificada; si no `streamId` se proporciona, se selecciona la transmisión más reciente del canal.

```
aws ivs get-stream-session \  
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \  
  --stream-id "mystream"
```

Salida:

```
{  
  "streamSession": {  
    "streamId": "mystream1",  
    "startTime": "2023-06-26T19:09:28+00:00",  
    "channel": {  
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
      "name": "mychannel",  
      "latencyMode": "LOW",  
      "type": "STANDARD",  
      "recordingConfigurationArn": "arn:aws:ivs:us-  
west-2:123456789012:recording-configuration/ABCdef34ghIJ",  
      "srt": {  
        "endpoint": "a1b2c3d4e5f6.srt.live-video.net",  
        "passphrase":  
"AB1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijk1MN5opqrStuVWxyzAbCDEfghIJ"  
      },  
    },  
  },  
}
```

```
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "playbackUrl": "url-string",
    "authorized": false,
    "insecureIngest": false,
    "preset": ""
  },
  "ingestConfiguration": {
    "video": {
      "avcProfile": "Baseline",
      "avcLevel": "4.2",
      "codec": "avc1.42C02A",
      "encoder": "Lavf58.45.100",
      "targetBitrate": 8789062,
      "targetFramerate": 60,
      "videoHeight": 1080,
      "videoWidth": 1920
    },
    "audio": {
      "codec": "mp4a.40.2",
      "targetBitrate": 46875,
      "sampleRate": 8000,
      "channels": 2
    }
  },
  "recordingConfiguration": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
ABCdef34ghIJ",
    "name": "test-recording-config",
    "destinationConfiguration": {
      "s3": {
        "bucketName": "demo-recording-bucket"
      }
    },
    "state": "ACTIVE",
    "tags": {
      "key1": "value1",
      "key2": "value2"
    },
    "thumbnailConfiguration": {
      "recordingMode": "INTERVAL",
      "targetIntervalSeconds": 1,
      "resolution": "LOWEST_RESOLUTION",
      "storage": [
        "LATEST"
```

```
    ]
  },
  "recordingReconnectWindowSeconds": 60,
  "renditionConfiguration": {
    "renditionSelection": "CUSTOM",
    "renditions": [
      "HD"
    ]
  }
},
"truncatedEvents": [
  {
    "name": "Recording Start",
    "type": "IVS Recording State Change",
    "eventTime": "2023-06-26T19:09:35+00:00"
  },
  {
    "name": "Stream Start",
    "type": "IVS Stream State Change",
    "eventTime": "2023-06-26T19:09:34+00:00"
  },
  {
    "name": "Session Created",
    "type": "IVS Stream State Change",
    "eventTime": "2023-06-26T19:09:28+00:00"
  }
]
}
}
```

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [GetStreamSession](#) la Referencia de AWS CLI comandos.

## get-stream

En el siguiente ejemplo de código, se muestra cómo usar `get-stream`.

### AWS CLI

Para obtener información sobre una transmisión

En el siguiente `get-stream` ejemplo, se obtiene información sobre la transmisión del canal especificado.

```
aws ivs get-stream \  
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Salida:

```
{  
  "stream": {  
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/  
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",  
    "startTime": "2020-05-05T21:55:38Z",  
    "state": "LIVE",  
    "health": "HEALTHY",  
    "streamId": "st-ABCDefghij01234KLMN5678",  
    "viewerCount": 1  
  }  
}
```

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [GetStream](#) la Referencia de AWS CLI comandos.

## **import-playback-key-pair**

En el siguiente ejemplo de código, se muestra cómo usar `import-playback-key-pair`.

### AWS CLI

Para importar la parte pública de un nuevo key pair

El siguiente `import-playback-key-pair` ejemplo importa la clave pública especificada (especificada como una cadena en PEM formato) y devuelve el arn y la huella digital del nuevo par de claves.

```
aws ivs import-playback-key-pair \  
  --name "my-playback-key" \  
  --public-key "-----BEGIN PUBLIC KEY-----  
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA...
```



```
--public-key-material "G11bnQx0TA3BgNVBAMMFdoeSBhcmUgeW91IGR1..."
```

Salida:

```
{
  "keyPair": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh",
    "name": "my-playback-key",
    "fingerprint": "0a:1b:2c:ab:cd:ef:34:56:70:b1:b2:71:01:2a:a3:72",
    "tags": {}
  }
}
```

Para obtener más información, consulte [Configuración de canales privados](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ImportPlaybackKeyPair](#) la Referencia de AWS CLI comandos.

## list-channels

En el siguiente ejemplo de código, se muestra cómo usar `list-channels`.

AWS CLI

Ejemplo 1: Para obtener información resumida sobre todos los canales

En el siguiente `list-channels` ejemplo, se enumeran todos los canales de tu AWS cuenta.

```
aws ivs list-channels
```

Salida:

```
{
  "channels": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "name": "channel-1",
      "latencyMode": "LOW",
      "authorized": false,
      "insecureIngest": false,
      "preset": "",
    }
  ]
}
```

```

        "playbackRestrictionPolicyArn": "",
        "recordingConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/ABCD12cdEFgh",
        "tags": {},
        "type": "STANDARD"
    },
    {
        "arn": "arn:aws:ivs:us-west-2:123456789012:channel/efghEFGHijkl",
        "name": "channel-2",
        "latencyMode": "LOW",
        "authorized": false,
        "preset": "",
        "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ",
        "recordingConfigurationArn": "",
        "tags": {},
        "type": "STANDARD"
    }
]
}

```

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

Ejemplo 2: Para obtener información resumida sobre todos los canales, filtrada por los especificados RecordingConfiguration ARN

En el siguiente `list-channels` ejemplo, se enumeran todos los canales de su AWS cuenta que están asociados al canal especificado RecordingConfiguration ARN.

```

aws ivs list-channels \
  --filter-by-recording-configuration-arn "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/ABCD12cdEFgh"

```

Salida:

```

{
  "channels": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "name": "channel-1",
      "latencyMode": "LOW",

```

```

        "authorized": false,
        "insecureIngest": false,
        "preset": "",
        "playbackRestrictionPolicyArn": "",
        "recordingConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/ABCD12cdEFgh",
        "tags": {},
        "type": "STANDARD"
    }
]
}

```

Para obtener más información, consulte [Grabar en Amazon S3](#) en la Guía del usuario IVS de baja latencia.

Ejemplo 3: Para obtener información resumida sobre todos los canales, filtrada según lo especificado PlaybackRestrictionPolicy ARN

En el siguiente `list-channels` ejemplo, se enumeran todos los canales de su AWS cuenta que están asociados al canal especificado PlaybackRestrictionPolicy ARN.

```

aws ivs list-channels \
  --filter-by-playback-restriction-policy-arn "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ"

```

Salida:

```

{
  "channels": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/efghEFGHijkl",
      "name": "channel-2",
      "latencyMode": "LOW",
      "authorized": false,
      "preset": "",
      "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ",
      "recordingConfigurationArn": "",
      "tags": {},
      "type": "STANDARD"
    }
  ]
}

```

```
}
```

Para obtener más información, consulte [Contenido no deseado y espectadores](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte la Referencia [ListChannels](#) de AWS CLI comandos.

## list-playback-key-pairs

En el siguiente ejemplo de código, se muestra cómo usar `list-playback-key-pairs`.

### AWS CLI

Para obtener información resumida sobre todos los pares de claves de reproducción

El siguiente `list-playback-key-pairs` ejemplo devuelve información sobre todos los pares de claves.

```
aws ivs list-playback-key-pairs
```

Salida:

```
{
  "keyPairs": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh",
      "name": "test-key-0",
      "tags": {}
    },
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:playback-key/ijkl15678mnop",
      "name": "test-key-1",
      "tags": {}
    }
  ]
}
```

Para obtener más información, consulte [Configuración de canales privados](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListPlaybackKeyPairs](#) la Referencia de AWS CLI comandos.

## list-playback-restriction-policies

En el siguiente ejemplo de código, se muestra cómo usar `list-playback-restriction-policies`.

### AWS CLI

Para obtener información resumida sobre todas las políticas de restricción de reproducción

En el siguiente `list-playback-restriction-policies` ejemplo, se enumeran todas las políticas de restricción de reproducción de su AWS cuenta.

```
aws ivs list-playback-restriction-policies
```

Salida:

```
{
  "playbackRestrictionPolicies": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/
ABcdef34ghIJ",
      "allowedCountries": [
        "US",
        "MX"
      ],
      "allowedOrigins": [
        "https://www.website1.com",
        "https://www.website2.com"
      ],
      "enableStrictOriginEnforcement": true,
      "name": "test-playback-restriction-policy",
      "tags": {
        "key1": "value1",
        "key2": "value2"
      }
    }
  ]
}
```

Para obtener más información, consulte [Contenido no deseado y espectadores](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte la Referencia [ListPlaybackRestrictionPolicies](#) de AWS CLI comandos.

## list-recording-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-recording-configurations`.

AWS CLI

Para enumerar todos los `RecordingConfiguration` recursos creados en esta cuenta

En el siguiente `list-recording-configurations` ejemplo, se obtiene información sobre todos `RecordingConfiguration` los recursos de la cuenta.

```
aws ivs list-recording-configurations
```

Salida:

```
{
  "recordingConfigurations": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
ABcdef34ghIJ",
      "name": "test-recording-config-1",
      "destinationConfiguration": {
        "s3": {
          "bucketName": "demo-recording-bucket-1"
        }
      },
      "state": "ACTIVE",
      "tags": {}
    },
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
CD12abcdGHIJ",
      "name": "test-recording-config-2",
      "destinationConfiguration": {
        "s3": {
          "bucketName": "demo-recording-bucket-2"
        }
      },
      "state": "ACTIVE",
```

```
        "tags": {}
      }
    ]
  }
```

Para obtener más información, consulte [Grabar en Amazon S3](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListRecordingConfigurations](#) la Referencia de AWS CLI comandos.

## list-stream-keys

En el siguiente ejemplo de código, se muestra cómo usar `list-stream-keys`.

### AWS CLI

Para obtener una lista de claves de transmisión

En el siguiente `list-stream-keys` ejemplo, se enumeran todas las claves de flujo de un determinado ARN (Amazon Resource Name).

```
aws ivs list-stream-keys \
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Salida:

```
{
  "streamKeys": [
    {
      "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/abcdABCDefgh",
      "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "tags": {}
    }
  ]
}
```

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [ListStreamKeys](#) la Referencia de AWS CLI comandos.

## list-stream-sessions

En el siguiente ejemplo de código, se muestra cómo usar `list-stream-sessions`.

### AWS CLI

Para obtener un resumen de las transmisiones actuales y anteriores de un canal específico en la AWS región actual

El siguiente `list-stream-sessions` ejemplo proporciona información resumida de las transmisiones de un canal específico ARN (Amazon Resource Name).

```
aws ivs list-stream-sessions \  
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \  
  --max-results 25 \  
  --next-token ""
```

Salida:

```
{  
  "nextToken": "set-2",  
  "streamSessions": [  
    {  
      "startTime": 1641578182,  
      "endTime": 1641579982,  
      "hasErrorEvent": false,  
      "streamId": "mystream"  
    }  
    ...  
  ]  
}
```

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [ListStreamSessions](#) la Referencia de AWS CLI comandos.

## list-streams

En el siguiente ejemplo de código, se muestra cómo usar `list-streams`.



## AWS CLI

Para obtener una lista de las transmisiones en directo y su estado

En el siguiente `list-streams` ejemplo, se enumeran todas las transmisiones en directo de tu AWS cuenta.

```
aws ivs list-streams
```

Salida:

```
{
  "streams": [
    {
      "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
      "state": "LIVE",
      "health": "HEALTHY",
      "streamId": "st-ABCDEFghij01234KLMN5678",
      "viewerCount": 1
    }
  ]
}
```

Para obtener más información, consulta [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [ListStreams](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar todas las etiquetas de un AWS recurso (por ejemplo: canal, clave de flujo)

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas de un recurso específico ARN (Amazon Resource Name).

```
aws ivs list-tags-for-resource \
  --resource-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Salida:

```
{
  "tags":
  {
    "key1": "value1",
    "key2": "value2"
  }
}
```

Para obtener más información, consulte [Etiquetado](#) en la APIreferencia de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListTagsForResource](#)la Referencia de AWS CLI comandos.

## put-metadata

En el siguiente ejemplo de código, se muestra cómo usar put-metadata.

AWS CLI

Para insertar metadatos en la transmisión activa de un canal específico

En el siguiente put-metadata ejemplo, se insertan los metadatos indicados en la transmisión del canal especificado.

```
aws ivs put-metadata \
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \
  --metadata '{"my": "metadata"}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [PutMetadata](#)la Referencia de AWS CLI comandos.

## start-viewer-session-revocation

En el siguiente ejemplo de código, se muestra cómo usar start-viewer-session-revocation.

## AWS CLI

Para revocar la sesión de un espectador para un par determinado de múltiples canales ARN e ID de espectador

En el siguiente `start-viewer-session-revocation` ejemplo, se inicia el proceso de revocación de la sesión de espectador asociada a un canal ARN y un ID de espectador específicos, hasta el número de versión de la sesión especificado, inclusive. Si no se proporciona la versión, el valor predeterminado es 0.

```
aws ivs batch-start-viewer-session-revocation \  
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \  
  --viewer-id abcdefg \  
  --viewer-session-versions-less-than-or-equal-to 1234567890
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de canales privados](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [StartViewerSessionRevocation](#) la Referencia de AWS CLI comandos.

## stop-stream

En el siguiente ejemplo de código, se muestra cómo usar `stop-stream`.

## AWS CLI

Para detener una transmisión específica

El siguiente `stop-stream` ejemplo detiene la transmisión en el canal especificado.

```
aws ivs stop-stream \  
  --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte [StopStream](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir o actualizar las etiquetas de un AWS recurso (por ejemplo: canal, clave de transmisión)

En el siguiente `tag-resource` ejemplo, se añaden o actualizan las etiquetas de un recurso específico ARN (Amazon Resource Name).

```
aws ivs tag-resource \  
  --resource-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \  
  --tags "tagkey1=tagvalue1, tagkey2=tagvalue2"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado](#) en la APIreferencia de Amazon Interactive Video Service.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar las etiquetas de un AWS recurso (por ejemplo: canal, clave de transmisión)

En el siguiente `untag-resource` ejemplo, se eliminan las etiquetas especificadas para un recurso específico ARN (Amazon Resource Name).

```
aws ivs untag-resource \  
  --resource-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \  
  --tag-keys "tagkey1, tagkey2"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado](#) en la APIreferencia de Amazon Interactive Video Service.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-channel

En el siguiente ejemplo de código, se muestra cómo usar `update-channel`.

### AWS CLI

Ejemplo 1: Para actualizar la información de configuración de un canal

El siguiente `update-channel` ejemplo actualiza la configuración del canal de un canal específico ARN para cambiar el nombre del canal. Esto no afecta a la transmisión en curso de este canal; debe detener y reiniciar la transmisión para que los cambios surtan efecto.

```
aws ivs update-channel \  
  --arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \  
  --name "channel-1" \  
  --insecure-ingest
```

Salida:

```
{  
  "channel": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
    "name": "channel-1",  
    "latencyMode": "LOW",  
    "type": "STANDARD",  
    "playbackRestrictionPolicyArn": "",  
    "recordingConfigurationArn": "",  
    "srt": {  
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",  
      "passphrase":  
"AB1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"  
    },  
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",  
    "insecureIngest": true,  
  }  
}
```

```

    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "preset": "",
    "authorized": false,
    "tags": {}
}

```

Para obtener más información, consulte [Crear un canal](#) en la Guía del usuario IVS de baja latencia.

Ejemplo 2: Para actualizar la configuración de un canal para permitir la grabación

El siguiente `update-channel` ejemplo actualiza la configuración del canal de un canal específico ARN para permitir la grabación. Esto no afecta a la transmisión en curso de este canal; debe detener y reiniciar la transmisión para que los cambios surtan efecto.

```

aws ivs update-channel \
  --arn "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh" \
  --no-insecure-ingest \
  --recording-configuration-arn "arn:aws:ivs:us-west-2:123456789012:recording-
configuration/ABCD12cdEFgh"

```

Salida:

```

{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "name": "test-channel-with-recording",
    "latencyMode": "LOW",
    "type": "STANDARD",
    "playbackRestrictionPolicyArn": "",
    "recordingConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:recording-
configuration/ABCD12cdEFgh",
    "srt": {
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
      "passphrase":
"BA1C2defGHijklMN03PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
    },
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": false,
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",

```

```

    "preset": "",
    "authorized": false,
    "tags": {}
  }
}

```

Para obtener más información, consulte [Grabar en Amazon S3](#) en la Guía del usuario IVS de baja latencia.

Ejemplo 3: Para actualizar la configuración de un canal para deshabilitar la grabación

El siguiente `update-channel` ejemplo actualiza la configuración del canal de un canal específico para inhabilitar ARN la grabación. Esto no afecta a la transmisión en curso de este canal; debe detener y reiniciar la transmisión para que los cambios surtan efecto.

```

aws ivs update-channel \
  --arn "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh" \
  --recording-configuration-arn ""

```

Salida:

```

{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "name": "test-channel-with-recording",
    "latencyMode": "LOW",
    "type": "STANDARD",
    "playbackRestrictionPolicyArn": "",
    "recordingConfigurationArn": "",
    "srt": {
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
      "passphrase":
"AB1C2edfGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijkLMN5opqrStuVWxyzAbCDEfghIJ"
    },
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": false,
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "preset": "",
    "authorized": false,
    "tags": {}
  }
}

```

```
}

```

Para obtener más información, consulte [Grabar en Amazon S3](#) en la Guía del usuario IVS de baja latencia.

Ejemplo 4: Para actualizar la configuración de un canal para habilitar la restricción de reproducción

El siguiente `update-channel` ejemplo actualiza la configuración del canal de un canal específico ARN para aplicar una política de restricción de reproducción. Esto no afecta a la transmisión en curso de este canal; debe detener y reiniciar la transmisión para que los cambios surtan efecto.

```
aws ivs update-channel \
  --arn "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh" \
  --no-insecure-ingest \
  --playback-restriction-policy-arn "arn:aws:ivs:us-west-2:123456789012:playback-
restriction-policy/ABCdef34ghIJ"
```

Salida:

```
{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "name": "test-channel-with-playback-restriction-policy",
    "latencyMode": "LOW",
    "type": "STANDARD",
    "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABCdef34ghIJ",
    "recordingConfigurationArn": "",
    "srt": {
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
      "passphrase":
"AB1C2defGHijklMN03PqQRstUvwxyzaCBDEfghh4ijklMN5opqrStuVWXYZAbCDEfghIJ"
    },
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": false,
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "preset": "",
    "authorized": false,
    "tags": {}
  }
}
```



```
}  
}
```

Para obtener más información, consulte [Contenido no deseado y espectadores](#) en la Guía del usuario IVS de baja latencia.

Ejemplo 5: Para actualizar la configuración de un canal para deshabilitar la restricción de reproducción

El siguiente `update-channel` ejemplo actualiza la configuración del canal de un canal específico para inhabilitar ARN la restricción de reproducción. Esto no afecta a la transmisión en curso de este canal; debe detener y reiniciar la transmisión para que los cambios surtan efecto.

```
aws ivs update-channel \  
  --arn "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh" \  
  --playback-restriction-policy-arn ""
```

Salida:

```
{  
  "channel": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
    "name": "test-channel-with-playback-restriction-policy",  
    "latencyMode": "LOW",  
    "type": "STANDARD",  
    "playbackRestrictionPolicyArn": "",  
    "recordingConfigurationArn": "",  
    "srt": {  
      "endpoint": "a1b2c3d4e5f6.srt.live-video.net",  
      "passphrase":  
"AB1C2defGHijkLMNo3PqQRstUvwxyzaBCDeFghh4ijk1MN5opqrStuVWxyzAbCDEfghIJ"  
    },  
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",  
    "insecureIngest": false,  
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/  
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",  
    "preset": "",  
    "authorized": false,  
    "tags": {}  
  }  
}
```

Para obtener más información, consulte [Contenido no deseado y espectadores](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte la Referencia [UpdateChannel](#) de AWS CLI comandos.

## update-playback-restriction-policy

En el siguiente ejemplo de código, se muestra cómo usar `update-playback-restriction-policy`.

### AWS CLI

Para actualizar una política de restricción de reproducción

El siguiente `update-playback-restriction-policy` ejemplo actualiza la política de restricción de reproducción con la política especificada ARN para deshabilitar la aplicación estricta de Origin. Esto no afecta a la transmisión en curso del canal asociado; debe detener y reiniciar la transmisión para que los cambios surtan efecto.

```
aws ivs update-playback-restriction-policy \  
  --arn "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/  
  ABcdef34ghIJ" \  
  --no-enable-strict-origin-enforcement
```

Salida:

```
{  
  "playbackRestrictionPolicy": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/  
  ABcdef34ghIJ",  
    "allowedCountries": [  
      "US",  
      "MX"  
    ],  
    "allowedOrigins": [  
      "https://www.website1.com",  
      "https://www.website2.com"  
    ],  
    "enableStrictOriginEnforcement": false,  
    "name": "test-playback-restriction-policy",  
    "tags": {
```

```
        "key1": "value1",
        "key2": "value2"
    }
}
```

Para obtener más información, consulte [Contenido no deseado y espectadores](#) en la Guía del usuario IVS de baja latencia.

- Para API obtener más información, consulte la Referencia [UpdatePlaybackRestrictionPolicy](#) de AWS CLI comandos.

## Ejemplos de Amazon IVS Chat que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface mediante Amazon IVS Chat.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-chat-token**

En el siguiente ejemplo de código, se muestra cómo usar `create-chat-token`.

AWS CLI

Para crear un token de chat

En el siguiente `create-chat-token` ejemplo, se crea un token de chat cifrado que se utiliza para establecer una WebSocket conexión individual con una sala. El token es válido durante un minuto y una conexión (sesión) establecida con el token es válida durante el tiempo especificado.

```
aws ivschat create-chat-token \
  --roomIdIdentifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6", \
  --userId "11231234" \
  --capabilities "SEND_MESSAGE", \
  --sessionDurationInMinutes 30
```

Salida:

```
{
  "token": "ACEGmnoq#1rstu2...BDFH3vxyw!4hlm!#5",
  "sessionExpirationTime": "2022-03-16T04:44:09+00:00"
  "state": "CREATING",
  "tokenExpirationTime": "2022-03-16T03:45:09+00:00"
}
```

Para obtener más información, consulte el [paso 3: Autenticar y autorizar los clientes de chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [CreateChatToken](#) la Referencia de AWS CLI comandos.

## create-logging-configuration

En el siguiente ejemplo de código, se muestra cómo usar `create-logging-configuration`.

AWS CLI

Para crear un `LoggingConfiguration` recurso de chat

El siguiente `create-logging-configuration` ejemplo crea un `LoggingConfiguration` recurso que permite a los clientes almacenar y grabar los mensajes enviados.

```
aws ivschat create-logging-configuration \
  --destination-configuration s3={bucketName=demo-logging-bucket} \
  --name "test-logging-config" \
  --tags "key1=value1, key2=value2"
```

Salida:

```
{
```

```

    "arn": "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/
    ABcdef34ghIJ",
    "createTime": "2022-09-14T17:48:00.653000+00:00",
    "destinationConfiguration": {
      "s3": {
        "bucketName": "demo-logging-bucket"
      }
    },
    "id": "ABcdef34ghIJ",
    "name": "test-logging-config",
    "state": "ACTIVE",
    "tags": { "key1" : "value1", "key2" : "value2" },
    "updateTime": "2022-09-14T17:48:01.104000+00:00"
  }

```

Para obtener más información, consulte [Introducción a Amazon IVS Chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [CreateLoggingConfiguration](#) la Referencia de AWS CLI comandos.

## create-room

En el siguiente ejemplo de código, se muestra cómo usar create-room.

### AWS CLI

Para crear una habitación

En el create-room ejemplo siguiente se crea una habitación nueva.

```

aws ivschat create-room \
  --name "test-room-1" \
  --logging-configuration-identifiers "arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABcdef34ghIJ" \
  --maximum-message-length 256 \
  --maximum-message-rate-per-second 5

```

Salida:

```

{
  "arn": "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6",

```

```
"id": "g1H2I3j4k5L6",
"createTime": "2022-03-16T04:44:09+00:00",
"loggingConfigurationIdentifiers": ["arn:aws:ivschat:us-west-2:123456789012:logging-configuration/ABcdef34ghIJ"],
"maximumMessageLength": 256,
"maximumMessageRatePerSecond": 5,
"name": "test-room-1",
"tags": {}
"updateTime": "2022-03-16T07:22:09+00:00"
}
```

Para obtener más información, consulte el [paso 2: Crear una sala de chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [CreateRoom](#) la Referencia de AWS CLI comandos.

## delete-logging-configuration

En el siguiente ejemplo de código, se muestra cómo usar delete-logging-configuration.

### AWS CLI

Para eliminar un LoggingConfiguration recurso de chat

En el siguiente delete-logging-configuration ejemplo, se elimina el LoggingConfiguration recurso del especificadoARN.

```
aws ivschat delete-logging-configuration \
  --identifier "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/
  ABcdef34ghIJ"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Introducción a Amazon IVS Chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [DeleteLoggingConfiguration](#) la Referencia de AWS CLI comandos.

## delete-message

En el siguiente ejemplo de código, se muestra cómo usar delete-message.

## AWS CLI

Para eliminar mensajes de una sala específica

En el siguiente `delete-message` ejemplo, se envía un par a la sala especificada, que indica a los clientes que eliminen el mensaje especificado, es decir, que lo retiren de la vista y que lo eliminen del historial de chat del cliente.

```
aws ivschat delete-message \  
  --roomIdentifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6" \  
  --id "ABC123def456" \  
  --reason "Message contains profanity"
```

Salida:

```
{  
  "id": "12345689012"  
}
```

Para obtener más información, consulte [Introducción a Amazon IVS Chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [DeleteMessage](#) la Referencia de AWS CLI comandos.

## `delete-room`

En el siguiente ejemplo de código, se muestra cómo usar `delete-room`.

## AWS CLI

Para eliminar una habitación

En el siguiente `delete-room` ejemplo, se elimina la habitación especificada. Los clientes conectados están desconectados. En caso de éxito, devuelve HTTP 204 con el cuerpo de respuesta vacío.

```
aws ivschat delete-room \  
  --identifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Introducción a Amazon IVS Chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [DeleteRoom](#) la Referencia de AWS CLI comandos.

## **disconnect-user**

En el siguiente ejemplo de código, se muestra cómo usar `disconnect-user`.

### AWS CLI

Para desconectar a un usuario de una sala

El siguiente `disconnect-user` ejemplo desconecta todas las conexiones del usuario especificado de la sala especificada. En caso de éxito, devuelve HTTP 200 con el cuerpo de respuesta vacío.

```
aws ivschat disconnect-user \  
  --roomIdIdentifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6" \  
  --userId "ABC123def456" \  
  --reason "Violated terms of service"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Introducción a Amazon IVS Chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [DisconnectUser](#) la Referencia de AWS CLI comandos.

## **get-logging-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `get-logging-configuration`.

### AWS CLI

Para obtener información sobre un `LoggingConfiguration` recurso

En el siguiente `get-logging-configuration` ejemplo, se obtiene información sobre el `LoggingConfiguration` recurso especificadoARN.



```
aws ivschat get-logging-configuration \  
  --identifier "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/  
ABcdef34ghIJ"
```

Salida:

```
{  
  "arn": "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/  
ABcdef34ghIJ",  
  "createTime": "2022-09-14T17:48:00.653000+00:00",  
  "destinationConfiguration": {  
    "s3": {  
      "bucketName": "demo-logging-bucket"  
    }  
  },  
  "id": "ABcdef34ghIJ",  
  "name": "test-logging-config",  
  "state": "ACTIVE",  
  "tags": { "key1" : "value1", "key2" : "value2" },  
  "updateTime": "2022-09-14T17:48:01.104000+00:00"  
}
```

Para obtener más información, consulte [Introducción a Amazon IVS Chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [GetLoggingConfiguration](#) la Referencia de AWS CLI comandos.

## get-room

En el siguiente ejemplo de código, se muestra cómo usar `get-room`.

### AWS CLI

Para obtener la habitación especificada

En el `get-room` ejemplo siguiente se obtiene información sobre la habitación especificada.

```
aws ivschat get-room \  
  --identifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6"
```

Salida:

```
{
  "arn": "arn:aws:ivschat:us-west-2:123456789012:room/g1H2I3j4k5L6",
  "createTime": "2022-03-16T04:44:09+00:00",
  "id": "g1H2I3j4k5L6",
  "loggingConfigurationIdentifiers": ["arn:aws:ivschat:us-west-2:123456789012:logging-configuration/ABCdef34ghIJ"],
  "maximumMessageLength": 256,
  "maximumMessageRatePerSecond": 5,
  "name": "test-room-1",
  "tags": {},
  "updateTime": "2022-03-16T07:22:09+00:00"
}
```

Para obtener más información, consulte [Introducción a Amazon IVS Chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [GetRoom](#) la Referencia de AWS CLI comandos.

## list-logging-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-logging-configurations`.

AWS CLI

Para obtener información resumida sobre todas las configuraciones de registro del usuario en la AWS región en la que se procesa la API solicitud

En el siguiente `list-logging-configurations` ejemplo, se muestra información sobre todos `LoggingConfiguration` los recursos del usuario en la AWS región en la que se procesa la API solicitud.

```
aws ivschat list-logging-configurations \
  --max-results 2 \
  --next-token ""
```

Salida:

```
{
  "nextToken": "set-2",
  "loggingConfigurations": [
```

```

    {
      "arn": "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/
ABCdef34ghIJ",
      "createTime": "2022-09-14T17:48:00.653000+00:00",
      "destinationConfiguration": {
        "s3": {
          "bucketName": "demo-logging-bucket"
        }
      },
      "id": "ABCdef34ghIJ",
      "name": "test-logging-config",
      "state": "ACTIVE",
      "tags": { "key1" : "value1", "key2" : "value2" },
      "updateTime": "2022-09-14T17:48:01.104000+00:00"
    }
    ...
  ]
}

```

Para obtener más información, consulte [Introducción a Amazon IVS Chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListLoggingConfigurations](#) la Referencia de AWS CLI comandos.

## list-rooms

En el siguiente ejemplo de código, se muestra cómo usar `list-rooms`.

### AWS CLI

Para obtener información resumida sobre todas las habitaciones de la región actual

En el siguiente `list-rooms` ejemplo, se obtiene información resumida sobre todas las habitaciones de la AWS región en la que se procesa la solicitud. Los resultados se ordenan en orden descendente de `updateTime`.

```

aws ivschat list-rooms \
  --logging-configuration-identifier "arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABCdef34ghIJ" \
  --max-results 10 \
  --next-token ""

```

Salida:

```
{
  "nextToken": "page3",
  "rooms": [
    {
      "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6",
      "createTime": "2022-03-16T04:44:09+00:00",
      "id": "g1H2I3j4k5L6",
      "loggingConfigurationIdentifiers": ["arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABCdef34ghIJ"],
      "name": "test-room-1",
      "tags": {},
      "updateTime": "2022-03-16T07:22:09+00:00"
    }
  ]
}
```

Para obtener más información, consulte [Introducción a Amazon IVS Chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListRooms](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

AWS CLI

Para enumerar todas las etiquetas de un AWS recurso (por ejemplo: Room)

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas de un recurso específico ARN (Amazon Resource Name).

```
aws ivschat list-tags-for-resource \
  --resource-arn arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6
```

Salida:

```
{
  "tags":
  {
```

```
    "key1": "value1",
    "key2": "value2"
  }
}
```

Para obtener más información, consulte [Etiquetado](#) en la APIreferencia de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## send-event

En el siguiente ejemplo de código, se muestra cómo usar send-event.

### AWS CLI

Para enviar un evento a una sala

El siguiente send-event ejemplo envía el evento dado a la sala especificada.

```
aws ivschat send-event \
  --roomIdentifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6" \
  --eventName "SystemMessage" \
  --attributes \
    "msgType"="user-notification", \
    "msgText"="This chat room will close in 15 minutes."
```

Salida:

```
{
  "id": "12345689012"
}
```

Para obtener más información, consulte [Introducción a Amazon IVS Chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [SendEvent](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

## AWS CLI

Para añadir o actualizar las etiquetas de un AWS recurso (por ejemplo: Room)

En el siguiente `tag-resource` ejemplo, se añaden o actualizan las etiquetas de un recurso específico ARN (Amazon Resource Name). En caso de éxito, devuelve HTTP 200 con el cuerpo de respuesta vacío.

```
aws ivschat tag-resource \  
  --resource-arn arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6 \  
  --tags "tagkey1=tagkeyvalue1, tagkey2=tagkeyvalue2"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado](#) en la APIreferencia de Amazon Interactive Video Service.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

## AWS CLI

Para eliminar las etiquetas de un AWS recurso (por ejemplo: Room)

En el siguiente `untag-resource` ejemplo, se eliminan las etiquetas especificadas para un recurso específico ARN (Amazon Resource Name). En caso de éxito, devuelve HTTP 200 con el cuerpo de respuesta vacío.

```
aws ivschat untag-resource \  
  --resource-arn arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6 \  
  --tag-keys "tagkey1, tagkey2"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado](#) en la APIreferencia de Amazon Interactive Video Service.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-logging-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-logging-configuration`.

### AWS CLI

Para actualizar la configuración de registro de una sala

En el siguiente `update-logging-configuration` ejemplo, se actualiza un `LoggingConfiguration` recurso con los datos proporcionados.

```
aws ivschat update-logging-configuration \
  --destination-configuration s3={bucketName=demo-logging-bucket} \
  --identifier "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/ABcdef34ghIJ" \
  --name "test-logging-config"
```

Salida:

```
{
  "arn": "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/ABcdef34ghIJ",
  "createTime": "2022-09-14T17:48:00.653000+00:00",
  "destinationConfiguration": {
    "s3": {
      "bucketName": "demo-logging-bucket"
    }
  },
  "id": "ABcdef34ghIJ",
  "name": "test-logging-config",
  "state": "ACTIVE",
  "tags": { "key1" : "value1", "key2" : "value2" },
  "updateTime": "2022-09-14T17:48:01.104000+00:00"
}
```

Para obtener más información, consulte [Introducción a Amazon IVS Chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [UpdateLoggingConfiguration](#) la Referencia de AWS CLI comandos.

## update-room

En el siguiente ejemplo de código, se muestra cómo usar `update-room`.

### AWS CLI

Para actualizar la configuración de una sala

El siguiente `update-room` ejemplo actualiza la configuración de la habitación especificada con los datos proporcionados.

```
aws ivschat update-room \  
  --identifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6" \  
  --logging-configuration-identifiers "arn:aws:ivschat:us-  
west-2:123456789012:logging-configuration/ABCdef34ghIJ" \  
  --name "chat-room-a" \  
  --maximum-message-length 256 \  
  --maximum-message-rate-per-second 5
```

Salida:

```
{  
  "arn": "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6",  
  "createTime": "2022-03-16T04:44:09+00:00",  
  "id": "g1H2I3j4k5L6",  
  "loggingConfigurationIdentifiers": ["arn:aws:ivschat:us-  
west-2:123456789012:logging-configuration/ABCdef34ghIJ"],  
  "maximumMessageLength": 256,  
  "maximumMessageRatePerSecond": 5,  
  "name": "chat-room-a",  
  "tags": {},  
  "updateTime": "2022-03-16T07:22:09+00:00"  
}
```

Para obtener más información, consulte [Introducción a Amazon IVS Chat](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [UpdateRoom](#) la Referencia de AWS CLI comandos.



## Ejemplos de Amazon IVS Real-Time Streaming utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante Amazon IVS Real-Time Streaming. AWS Command Line Interface

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-encoder-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `create-encoder-configuration`.

AWS CLI

Para crear una configuración de codificador de composición

El siguiente `create-encoder-configuration` ejemplo crea una configuración de codificador de composición con las propiedades especificadas.

```
aws ivs-realtime create-encoder-configuration \  
  --name test-ec --video bitrate=3500000,framerate=30.0,height=1080,width=1920
```

Salida:

```
{  
  "encoderConfiguration": {  
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/  
ABabCDcdEFef",  
    "name": "test-ec",  
    "tags": {},  
    "video": {
```

```
        "bitrate": 3500000,  
        "framerate": 30,  
        "height": 1080,  
        "width": 1920  
    }  
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [CreateEncoderConfiguration](#) la Referencia de AWS CLI comandos.

## create-participant-token

En el siguiente ejemplo de código, se muestra cómo usar `create-participant-token`.

### AWS CLI

Para crear un token de participante en el escenario

En el siguiente `create-participant-token` ejemplo, se crea un token de participante para la etapa especificada.

```
aws ivs-realtime create-participant-token \  
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \  
  --user-id bob
```

Salida:

```
{  
  "participantToken": {  
    "expirationTime": "2023-03-07T09:47:43+00:00",  
    "participantId": "ABCDEFghij01234KLMN6789",  
    "token": "abcd1234defg5678"  
  }  
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [CreateParticipantToken](#) la Referencia de AWS CLI comandos.

## create-stage

En el siguiente ejemplo de código, se muestra cómo usar `create-stage`.

### AWS CLI

Para crear un escenario

El siguiente `create-stage` ejemplo crea un escenario y un token de participante del escenario para un usuario específico.

```
aws ivs-realtime create-stage \  
  --name stage1 \  
  --participant-token-configurations userId=alice
```

Salida:

```
{  
  "participantTokens": [  
    {  
      "participantId": "ABCDEFghij01234KLMN5678",  
      "token": "a1b2c3d4567890ab",  
      "userId": "alice"  
    }  
  ],  
  "stage": {  
    "activeSessionId": "st-a1b2c3d4e5f6g",  
    "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",  
    "name": "stage1",  
    "tags": {}  
  }  
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [CreateStage](#) la Referencia de AWS CLI comandos.

## create-storage-configuration

En el siguiente ejemplo de código, se muestra cómo usar `create-storage-configuration`.

### AWS CLI

Para crear una configuración de almacenamiento de composiciones

El siguiente `create-storage-configuration` ejemplo crea una configuración de almacenamiento de composiciones con las propiedades especificadas.

```
aws ivs-realtime create-storage-configuration \  
  --name "test-sc" --s3 "bucketName=test-bucket-name"
```

Salida:

```
{  
  "storageConfiguration": {  
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/  
ABabCDcdEFef",  
    "name": "test-sc",  
    "s3": {  
      "bucketName": "test-bucket-name"  
    },  
    "tags": {}  
  }  
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [CreateStorageConfiguration](#) la Referencia de AWS CLI comandos.

## delete-encoder-configuration

En el siguiente ejemplo de código, se muestra cómo usar `delete-encoder-configuration`.

### AWS CLI

Para eliminar la configuración de un codificador de composición

Lo siguiente `delete-encoder-configuration` elimina la configuración del codificador de composición especificada por el dato (nombre de recurso de ARN Amazon).

```
aws ivs-realtime delete-encoder-configuration \  
  --arn "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/  
  ABabCDcdEFef"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [DeleteEncoderConfiguration](#) la Referencia de AWS CLI comandos.

## delete-stage

En el siguiente ejemplo de código, se muestra cómo usar `delete-stage`.

### AWS CLI

Para eliminar una etapa

En el siguiente `delete-stage` ejemplo, se elimina la etapa especificada.

```
aws ivs-realtime delete-stage \  
  --arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [DeleteStage](#) la Referencia de AWS CLI comandos.

## delete-storage-configuration

En el siguiente ejemplo de código, se muestra cómo usar `delete-storage-configuration`.

### AWS CLI

Para eliminar una configuración de almacenamiento de maquetación

Lo siguiente `delete-storage-configuration` elimina la configuración de almacenamiento de composición especificada por el ARN (nombre de recurso de Amazon) indicado.

```
aws ivs-realtime delete-storage-configuration \  
  --arn "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/  
  ABabCDcdEFef"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [DeleteStorageConfiguration](#) la Referencia de AWS CLI comandos.

## disconnect-participant

En el siguiente ejemplo de código, se muestra cómo usar `disconnect-participant`.

### AWS CLI

Para desconectar a un participante del escenario

El siguiente `disconnect-participant` ejemplo desconecta al participante especificado de la etapa especificada.

```
aws ivs-realtime disconnect-participant \  
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \  
  --participant-id ABCDEfghij01234KLMN5678
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [DisconnectParticipant](#) la Referencia de AWS CLI comandos.

## get-composition

En el siguiente ejemplo de código, se muestra cómo usar `get-composition`.

## AWS CLI

Ejemplo 1: Para obtener una composición con los ajustes de maquetación predeterminados

En el siguiente `get-composition` ejemplo, se obtiene la composición del ARN (Amazon Resource Name) especificado.

```
aws ivs-realtime get-composition \  
--arn "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh"
```

Salida:

```
{  
  "composition": {  
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh",  
    "destinations": [  
      {  
        "configuration": {  
          "channel": {  
            "channelArn": "arn:aws:ivs:ap-northeast-1:123456789012:channel/abcABCdefDEg",  
            "encoderConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"  
          },  
          "name": ""  
        },  
        "id": "AabBCcdDEefF",  
        "startTime": "2023-10-16T23:26:00+00:00",  
        "state": "ACTIVE"  
      },  
      {  
        "configuration": {  
          "name": "",  
          "s3": {  
            "encoderConfigurationArns": [  
              "arn:aws:ivs:arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"  
            ],  
            "recordingConfiguration": {  
              "format": "HLS"  
            },  
            "storageConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/FefABabCDcdE"  
          }  
        }  
      }  
    ]  
  }  
}
```

```

        }
      },
      "detail": {
        "s3": {
          "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
        }
      },
      "id": "GHFabcgefABC",
      "startTime": "2023-10-16T23:26:00+00:00",
      "state": "STARTING"
    }
  ],
  "layout": {
    "grid": {
      "featuredParticipantAttribute": ""
      "gridGap": 2,
      "omitStoppedVideo": false,
      "videoAspectRatio": "VIDEO",
      "videoFillMode": ""
    }
  },
  "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
  "startTime": "2023-10-16T23:24:00+00:00",
  "state": "ACTIVE",
  "tags": {}
}
}

```

Para obtener más información, consulte [Grabación compuesta \(transmisión en tiempo real\)](#) en la Guía del usuario de Amazon Interactive Video Service.

Ejemplo 2: Para obtener una composición con un diseño PiP

El siguiente `get-composition` ejemplo obtiene la composición del ARN (Amazon Resource Name) especificado, que utiliza el diseño PiP.

```

aws ivs-realtime get-composition \
  --arn "arn:aws:ivs:ap-northeast-1:123456789012:composition/wxyzWXYZpqrs"

```

Salida:

```

{
  "composition": {

```



```

"arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/wxyzWXYZpqrs",
"destinations": [
  {
    "configuration": {
      "channel": {
        "channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg",
        "encoderConfigurationArn": "arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
      },
      "name": ""
    },
    "id": "AabBCcdDEefF",
    "startTime": "2023-10-16T23:26:00+00:00",
    "state": "ACTIVE"
  },
  {
    "configuration": {
      "name": "",
      "s3": {
        "encoderConfigurationArns": [
          "arn:aws:ivs:arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
        ],
        "recordingConfiguration": {
          "format": "HLS"
        },
        "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE"
      }
    },
    "detail": {
      "s3": {
        "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
      }
    },
    "id": "GHFabcgefABC",
    "startTime": "2023-10-16T23:26:00+00:00",
    "state": "STARTING"
  }
],
"layout": {
  "pip": {

```

```

        "featuredParticipantAttribute": "abcdefg",
        "gridGap": 0,
        "omitStoppedVideo": false,
        "pipBehavior": "STATIC",
        "pipOffset": 0,
        "pipParticipantAttribute": "",
        "pipPosition": "BOTTOM_RIGHT",
        "videoFillMode": "COVER"
    }
},
"stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
"startTime": "2023-10-16T23:24:00+00:00",
"state": "ACTIVE",
"tags": {}
}
}

```

Para obtener más información, consulte [Grabación compuesta \(transmisión en tiempo real\)](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [GetComposition](#) la Referencia de AWS CLI comandos.

## get-encoder-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-encoder-configuration`.

### AWS CLI

Para obtener una configuración de codificador de composición

El siguiente `get-encoder-configuration` ejemplo obtiene la configuración del codificador de composición especificada por el dato ARN (Amazon Resource Name).

```

aws ivs-realtime get-encoder-configuration \
  --arn "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/
abcdABCDefgh"

```

Salida:

```

{
  "encoderConfiguration": {

```

```

    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/
abcdABCDefgh",
    "name": "test-ec",
    "tags": {},
    "video": {
        "bitrate": 3500000,
        "framerate": 30,
        "height": 1080,
        "width": 1920
    }
}
}
}

```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [GetEncoderConfiguration](#) la Referencia de AWS CLI comandos.

## get-participant

En el siguiente ejemplo de código, se muestra cómo usar `get-participant`.

### AWS CLI

Para conseguir un participante en el escenario

En el siguiente `get-participant` ejemplo, se obtiene el participante de la etapa para un ID de participante y un ID de sesión específicos en la etapa especificada ARN (Amazon Resource Name).

```

aws ivs-realtime get-participant \
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
  --session-id st-a1b2c3d4e5f6g \
  --participant-id abCDEf12GHIj

```

Salida:

```

{
  "participant": {
    "browserName", "Google Chrome",

```

```

    "browserVersion", "116",
    "firstJoinTime": "2023-04-26T20:30:34+00:00",
    "ispName", "Comcast",
    "osName", "Microsoft Windows 10 Pro",
    "osVersion", "10.0.19044",
    "participantId": "abCDEf12GHIj",
    "published": true,
    "sdkVersion", "",
    "state": "DISCONNECTED",
    "userId": ""
  }
}

```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [GetParticipant](#) la Referencia de AWS CLI comandos.

## get-stage-session

En el siguiente ejemplo de código, se muestra cómo usar `get-stage-session`.

### AWS CLI

Para obtener una sesión de escenario

En el siguiente `get-stage-session` ejemplo, se obtiene la sesión de la etapa para un ID de sesión específico de una etapa específica ARN (Amazon Resource Name).

```

aws ivs-realtime get-stage-session \
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
  --session-id st-a1b2c3d4e5f6g

```

Salida:

```

{
  "stageSession": {
    "endTime": "2023-04-26T20:36:29+00:00",
    "sessionId": "st-a1b2c3d4e5f6g",
    "startTime": "2023-04-26T20:30:29.602000+00:00"
  }
}

```

```
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [GetStageSession](#) la Referencia de AWS CLI comandos.

## get-stage

En el siguiente ejemplo de código, se muestra cómo usar `get-stage`.

### AWS CLI

Para obtener la información de configuración de una etapa

El siguiente `get-stage` ejemplo obtiene la configuración de la etapa para una etapa específica ARN (Amazon Resource Name).

```
aws ivs-realtime get-stage \  
  --arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh
```

Salida:

```
{  
  "stage": {  
    "activeSessionId": "st-a1b2c3d4e5f6g",  
    "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",  
    "name": "test",  
    "tags": {}  
  }  
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [GetStage](#) la Referencia de AWS CLI comandos.

## get-storage-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-storage-configuration`.

## AWS CLI

Para obtener una configuración de almacenamiento de composiciones

El siguiente `get-storage-configuration` ejemplo obtiene la configuración de almacenamiento de la composición especificada por el dato ARN (Amazon Resource Name).

```
aws ivs-realtime get-storage-configuration \
  --name arn "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/
abcdABCDefgh"
```

Salida:

```
{
  "storageConfiguration": {
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/
abcdABCDefgh",
    "name": "test-sc",
    "s3": {
      "bucketName": "test-bucket-name"
    },
    "tags": {}
  }
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [GetStorageConfiguration](#) la Referencia de AWS CLI comandos.

## list-compositions

En el siguiente ejemplo de código, se muestra cómo usar `list-compositions`.

## AWS CLI

Para obtener una lista de composiciones

A continuación `list-compositions` se muestran todas las composiciones de tu AWS cuenta en la AWS región en la que se procesa la API solicitud.

**aws ivs-realtime list-compositions**

Salida:

```
{
  "compositions": [
    {
      "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/
abcdABCDefgh",
      "destinations": [
        {
          "id": "AabBCcdDEeff",
          "startTime": "2023-10-16T23:25:23+00:00",
          "state": "ACTIVE"
        }
      ],
      "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/
defgABCDabcd",
      "startTime": "2023-10-16T23:25:21+00:00",
      "state": "ACTIVE",
      "tags": {}
    },
    {
      "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/
ABcdabCDefgh",
      "destinations": [
        {
          "endTime": "2023-10-16T23:25:00.786512+00:00",
          "id": "aABbcCDdeEFf",
          "startTime": "2023-10-16T23:24:01+00:00",
          "state": "STOPPED"
        },
        {
          "endTime": "2023-10-16T23:25:00.786512+00:00",
          "id": "deEFfaABbcCD",
          "startTime": "2023-10-16T23:24:01+00:00",
          "state": "STOPPED"
        }
      ],
      "endTime": "2023-10-16T23:25:00+00:00",
      "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/
efghabcdABCD",
      "startTime": "2023-10-16T23:24:00+00:00",
    }
  ]
}
```

```
        "state": "STOPPED",
        "tags": {}
      }
    ]
  }
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListCompositions](#) la Referencia de AWS CLI comandos.

## list-encoder-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-encoder-configurations`.

### AWS CLI

Para ver una lista de las configuraciones de los codificadores de composición

A continuación, se `list-encoder-configurations` enumeran todas las configuraciones de codificadores de composición de su AWS cuenta, en la AWS región en la que se procesa la API solicitud.

```
aws ivs-realtime list-encoder-configurations
```

Salida:

```
{
  "encoderConfigurations": [
    {
      "arn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/abcdABCDefgh",
      "name": "test-ec-1",
      "tags": {}
    },
    {
      "arn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/ABCefgEFGabc",
      "name": "test-ec-2",
      "tags": {}
    }
  ]
}
```



```
]
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListEncoderConfigurations](#) la Referencia de AWS CLI comandos.

## list-participant-events

En el siguiente ejemplo de código, se muestra cómo usar `list-participant-events`.

### AWS CLI

Para obtener una lista de los eventos participantes en el escenario

En el siguiente `list-participant-events` ejemplo, se enumeran todos los eventos de los participantes para un ID de participante y un ID de sesión específicos de una etapa específica ARN (Amazon Resource Name).

```
aws ivs-realtime list-participant-events \
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
  --session-id st-a1b2c3d4e5f6g \
  --participant-id abCDEf12GHIj
```

Salida:

```
{
  "events": [
    {
      "eventTime": "2023-04-26T20:36:28+00:00",
      "name": "LEFT",
      "participantId": "abCDEf12GHIj"
    },
    {
      "eventTime": "2023-04-26T20:36:28+00:00",
      "name": "PUBLISH_STOPPED",
      "participantId": "abCDEf12GHIj"
    },
    {
      "eventTime": "2023-04-26T20:30:34+00:00",
```

```

        "name": "JOINED",
        "participantId": "abCDEf12GHIj"
    },
    {
        "eventTime": "2023-04-26T20:30:34+00:00",
        "name": "PUBLISH_STARTED",
        "participantId": "abCDEf12GHIj"
    }
]
}

```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListParticipantEvents](#) la Referencia de AWS CLI comandos.

## list-participants

En el siguiente ejemplo de código, se muestra cómo usar `list-participants`.

### AWS CLI

Para obtener una lista de los participantes del escenario

El siguiente `list-participants` ejemplo muestra una lista de todos los participantes de un ID de sesión específico de una etapa específica ARN (Amazon Resource Name).

```

aws ivs-realtime list-participants \
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
  --session-id st-a1b2c3d4e5f6g

```

Salida:

```

{
  "participants": [
    {
      "firstJoinTime": "2023-04-26T20:30:34+00:00",
      "participantId": "abCDEf12GHIj",
      "published": true,
      "state": "DISCONNECTED",
      "userId": ""
    }
  ]
}

```

```
    }  
  ]  
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListParticipants](#) la Referencia de AWS CLI comandos.

## list-stage-sessions

En el siguiente ejemplo de código, se muestra cómo usar `list-stage-sessions`.

### AWS CLI

Para obtener una lista de las sesiones de escenario

En el siguiente `list-stage-sessions` ejemplo, se enumeran todas las sesiones de una etapa específica ARN (Amazon Resource Name).

```
aws ivs-realtime list-stage-sessions \  
  --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh
```

Salida:

```
{  
  "stageSessions": [  
    {  
      "endTime": "2023-04-26T20:36:29+00:00",  
      "sessionId": "st-a1b2c3d4e5f6g",  
      "startTime": "2023-04-26T20:30:29.602000+00:00"  
    }  
  ]  
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListStageSessions](#) la Referencia de AWS CLI comandos.

## list-stages

En el siguiente ejemplo de código, se muestra cómo usar `list-stages`.

### AWS CLI

Para obtener información resumida sobre todas las etapas

En el siguiente `list-stages` ejemplo, se enumeran todas las etapas de tu AWS cuenta, en la AWS región en la que se procesa la API solicitud.

```
aws ivs-realtime list-stages
```

Salida:

```
{
  "stages": [
    {
      "activeSessionId": "st-a1b2c3d4e5f6g",
      "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
      "name": "stage1",
      "tags": {}
    },
    {
      "activeSessionId": "st-a123bcd456efg",
      "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcd1234ABCD",
      "name": "stage2",
      "tags": {}
    },
    {
      "activeSessionId": "st-abcDEF1234ghi",
      "arn": "arn:aws:ivs:us-west-2:123456789012:stage/ABCD1234efgh",
      "name": "stage3",
      "tags": {}
    }
  ]
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListStages](#) la Referencia de AWS CLI comandos.

## list-storage-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-storage-configurations`.

### AWS CLI

Para ver una lista de las configuraciones de almacenamiento de composiciones

A continuación, se `list-storage-configurations` enumeran todas las configuraciones de almacenamiento de composición de su AWS cuenta, en la AWS región en la que se procesa la API solicitud.

```
aws ivs-realtime list-storage-configurations
```

Salida:

```
{
  "storageConfigurations": [
    {
      "arn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/abcdABCDefgh",
      "name": "test-sc-1",
      "s3": {
        "bucketName": "test-bucket-1-name"
      },
      "tags": {}
    },
    {
      "arn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/ABCefgEFGabc",
      "name": "test-sc-2",
      "s3": {
        "bucketName": "test-bucket-2-name"
      },
      "tags": {}
    }
  ]
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [ListStorageConfigurations](#) la Referencia de AWS CLI comandos.

## start-composition

En el siguiente ejemplo de código, se muestra cómo usar start-composition.

### AWS CLI

Ejemplo 1: Para iniciar una composición con los ajustes de maquetación predeterminados

En el siguiente start-composition ejemplo, se inicia una composición para el escenario especificado para que se transmita a las ubicaciones especificadas.

```
aws ivs-realtime start-composition \
  --stage-arn arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCdabcd \
  --destinations '[{"channel": {"channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg", \
  "encoderConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-
configuration/ABabCDcdEFef"}}, \
  {"s3":{"encoderConfigurationArns":["arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"], \
  "storageConfigurationArn":"arn:aws:ivs:ap-northeast-1:123456789012:storage-
configuration/FefABabCDcdE"}}]'
```

Salida:

```
{
  "composition": {
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh",
    "destinations": [
      {
        "configuration": {
          "channel": {
            "channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg",
            "encoderConfigurationArn": "arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
          },
          "name": ""
        },
        "id": "AabBCcdDEefF",
```

```

        "state": "STARTING"
      },
      {
        "configuration": {
          "name": "",
          "s3": {
            "encoderConfigurationArns": [
              "arn:aws:ivs:arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
            ],
            "recordingConfiguration": {
              "format": "HLS"
            },
            "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE"
          }
        },
        "detail": {
          "s3": {
            "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
          }
        },
        "id": "GHFabcgefABC",
        "state": "STARTING"
      }
    ],
    "layout": {
      "grid": {
        "featuredParticipantAttribute": ""
        "gridGap": 2,
        "omitStoppedVideo": false,
        "videoAspectRatio": "VIDEO",
        "videoFillMode": ""
      }
    },
    "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
    "startTime": "2023-10-16T23:24:00+00:00",
    "state": "STARTING",
    "tags": {}
  }
}

```

Para obtener más información, consulte [Grabación compuesta \(transmisión en tiempo real\)](#) en la Guía del usuario de Amazon Interactive Video Service.

Ejemplo 2: Para iniciar una composición con un diseño PiP

En el siguiente `start-composition` ejemplo, se inicia una composición para el escenario especificado para que se transmita a las ubicaciones especificadas mediante el diseño PiP.

```
aws ivs-realtime start-composition \
  --stage-arn arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd \
  --destinations '[{"channel": {"channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg", \
  "encoderConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-
configuration/ABabCDcdEFef"}}, \
  {"s3":{"encoderConfigurationArns":["arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"], \
  "storageConfigurationArn":"arn:aws:ivs:ap-northeast-1:123456789012:storage-
configuration/FefABabCDcdE"}}]' \
  --layout pip='{featuredParticipantAttribute="abcdefg"}'
```

Salida:

```
{
  "composition": {
    "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/wxyzWXYZpqrs",
    "destinations": [
      {
        "configuration": {
          "channel": {
            "channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg",
            "encoderConfigurationArn": "arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
          },
          "name": ""
        },
        "id": "AabBCcdDEefF",
        "state": "STARTING"
      },
      {
        "configuration": {
          "name": "",
          "s3": {
```



```

        "encoderConfigurationArns": [
            "arn:aws:ivs:arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
        ],
        "recordingConfiguration": {
            "format": "HLS"
        },
        "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE"
    }
},
"detail": {
    "s3": {
        "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
    }
},
"state": "STARTING"
}
],
"layout": {
    "pip": {
        "featuredParticipantAttribute": "abcdefg",
        "gridGap": 0,
        "omitStoppedVideo": false,
        "pipBehavior": "STATIC",
        "pipOffset": 0,
        "pipParticipantAttribute": "",
        "pipPosition": "BOTTOM_RIGHT",
        "videoFillMode": "COVER"
    }
},
"stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCdabcd",
"startTime": "2023-10-16T23:24:00+00:00",
"state": "STARTING",
"tags": {}
}
}

```

Para obtener más información, consulte [Grabación compuesta \(transmisión en tiempo real\)](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [StartComposition](#) la Referencia de AWS CLI comandos.

## stop-composition

En el siguiente ejemplo de código, se muestra cómo usar stop-composition.

### AWS CLI

Para detener una composición

Lo siguiente stop-composition detiene la composición especificada por el dato ARN (nombre del recurso de Amazon).

```
aws ivs-realtime stop-composition \  
  --arn "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [StopComposition](#) la Referencia de AWS CLI comandos.

## update-stage

En el siguiente ejemplo de código, se muestra cómo usar update-stage.

### AWS CLI

Para actualizar la configuración de una etapa

El siguiente update-stage ejemplo actualiza una etapa para una etapa específica ARN para actualizar el nombre de la etapa.

```
aws ivs-realtime update-stage \  
  --arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \  
  --name stage1a
```

Salida:

```
{
  "stage": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
    "name": "stage1a"
  }
}
```

Para obtener más información, consulte [Habilitar varios hosts en una IVS transmisión de Amazon](#) en la Guía del usuario de Amazon Interactive Video Service.

- Para API obtener más información, consulte [UpdateStage](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon Kendra que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Amazon Kendra.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-data-source**

En el siguiente ejemplo de código, se muestra cómo usar `create-data-source`.

#### AWS CLI

Para crear un conector de fuente de datos de Amazon Kendra

A continuación, se `create-data-source` crea y configura un conector de fuente de datos de Amazon Kendra. Puede usarlo `describe-data-source` para ver el estado de un conector de

fuente de datos y leer cualquier mensaje de error si el estado muestra un conector de fuente de datos «FAILED» para crearlo completamente.

```
aws kendra create-data-source \
  --name "example data source 1" \
  --description "Example data source 1 for example index 1 contains the first set
of example documents" \
  --tags '{"Key": "test resources", "Value": "kendra"}, {"Key": "test resources",
"Value": "aws"}' \
  --role-arn "arn:aws:iam::my-account-id:role/
KendraRoleForS3TemplateConfigDataSource" \
  --index-id exampleindex1 \
  --language-code "es" \
  --schedule "0 0 18 ? * TUE,MON,WED,THU,FRI,SAT *" \
  --configuration '{"TemplateConfiguration": {"Template": file://
s3schemaconfig.json}}' \
  --type "TEMPLATE" \
  --custom-document-enrichment-configuration '{"PostExtractionHookConfiguration":
{"LambdaArn": "arn:aws:iam::my-account-id:function/my-function-ocr-docs",
"S3Bucket": "s3://my-s3-bucket/scanned-image-text-example-docs"}, "RoleArn":
"arn:aws:iam:my-account-id:role/KendraRoleForCDE"}' \
  --vpc-configuration '{"SecurityGroupIds": ["sg-1234567890abcdef0"], "SubnetIds":
["subnet-1c234", "subnet-2b134"]}'
```

Salida:

```
{
  "Id": "exampledatasource1"
}
```

Para obtener más información, consulte [Introducción a un conector de índice y fuente de datos de Amazon Kendra](#) en la Guía para desarrolladores de Amazon Kendra.

- Para API obtener más información, consulte la Referencia [CreateDataSource](#) de AWS CLI comandos.

## create-index

En el siguiente ejemplo de código, se muestra cómo usar create-index.

## AWS CLI

Para crear un índice de Amazon Kendra

A continuación, se `create-index` crea y configura un índice de Amazon Kendra. Puede usarlo `describe-index` para ver el estado de un índice y leer cualquier mensaje de error si el estado muestra un índice «FAILED» para crearlo completamente.

```
aws kendra create-index \  
  --name "example index 1" \  
  --description "Example index 1 contains the first set of example documents" \  
  --tags '{"Key": "test resources", "Value": "kendra"}, {"Key": "test resources",  
"Value": "aws"}' \  
  --role-arn "arn:aws:iam::my-account-id:role/KendraRoleForExampleIndex" \  
  --edition "DEVELOPER_EDITION" \  
  --server-side-encryption-configuration '{"KmsKeyId": "my-kms-key-id"}' \  
  --user-context-policy "USER_TOKEN" \  
  --user-token-configurations '{"JsonTokenTypeConfiguration":  
{"GroupAttributeField": "groupNameField", "UserNameAttributeField":  
"userNameField"}}'
```

Salida:

```
{  
  "Id": index1  
}
```

Para obtener más información, consulte [Introducción a un conector de índice y fuente de datos de Amazon Kendra](#) en la Guía para desarrolladores de Amazon Kendra.

- Para API obtener más información, consulte la Referencia [CreateIndex](#) de AWS CLI comandos.

## describe-data-source

En el siguiente ejemplo de código, se muestra cómo usar `describe-data-source`.

## AWS CLI

Para obtener información sobre un conector de fuente de datos de Amazon Kendra

A continuación, `describe-data-source` se obtiene información sobre un conector de fuente de datos de Amazon Kendra. Puede ver la configuración de un conector de fuente de datos y leer

cualquier mensaje de error si el estado muestra un conector de fuente de datos «FAILED» para crearlo por completo.

```
aws kendra describe-data-source \  
  --id exampledatasource1 \  
  --index-id exampleindex1
```

Salida:

```
{  
  "Configuration": {  
    "TemplateConfiguration": {  
      "Template": {  
        "connectionConfiguration": {  
          "repositoryEndpointMetadata": {  
            "BucketName": "my-bucket"  
          }  
        },  
        "repositoryConfigurations": {  
          "document": {  
            "fieldMappings": [  
              {  
                "indexFieldName": "_document_title",  
                "indexFieldType": "STRING",  
                "dataSourceFieldName": "title"  
              },  
              {  
                "indexFieldName": "_last_updated_at",  
                "indexFieldType": "DATE",  
                "dataSourceFieldName": "modified_date"  
              }  
            ]  
          }  
        },  
        "additionalProperties": {  
          "inclusionPatterns": [  
            "*.txt",  
            "*.doc",  
            "*.docx"  
          ],  
          "exclusionPatterns": [  
            "*.json"  
          ],  
        }  
      }  
    }  
  }  
}
```

```

        "inclusionPrefixes": [
            "PublicExampleDocsFolder"
        ],
        "exclusionPrefixes": [
            "PrivateDocsFolder/private"
        ],
        "aclConfigurationFilePath": "ExampleDocsFolder/AclConfig.json",
        "metadataFilesPrefix": "metadata"
    },
    "syncMode": "FULL_CRAWL",
    "type": "S3",
    "version": "1.0.0"
}
}
},
"CreatedAt": 2024-02-25T13:30:10+00:00,
"CustomDocumentEnrichmentConfiguration": {
    "PostExtractionHookConfiguration": {
        "LambdaArn": "arn:aws:iam::my-account-id:function/my-function-ocr-docs",
        "S3Bucket": "s3://my-s3-bucket/scanned-image-text-example-docs/function"
    },
    "RoleArn": "arn:aws:iam:my-account-id:role/KendraRoleForCDE"
}
>Description": "Example data source 1 for example index 1 contains the first set
of example documents",
"Id": "exampledatasource1",
"IndexId": "exampleindex1",
"LanguageCode": "en",
"Name": "example data source 1",
"RoleArn": "arn:aws:iam::my-account-id:role/
KendraRoleForS3TemplateConfigDataSource",
"Schedule": "0 0 18 ? * TUE,MON,WED,THU,FRI,SAT *",
"Status": "ACTIVE",
"Type": "TEMPLATE",
"UpdatedAt": 1709163615,
"VpcConfiguration": {
    "SecurityGroupIds": ["sg-1234567890abcdef0"],
    "SubnetIds": ["subnet-1c234","subnet-2b134"]
}
}
}

```

Para obtener más información, consulte [Introducción a un conector de índice y fuente de datos de Amazon Kendra](#) en la Guía para desarrolladores de Amazon Kendra.

- Para API obtener más información, consulte la Referencia [DescribeDataSource](#) de AWS CLI comandos.

## describe-index

En el siguiente ejemplo de código, se muestra cómo usar `describe-index`.

### AWS CLI

Para obtener información sobre un índice de Amazon Kendra

A continuación, `describe-index` se obtiene información sobre un índice de Amazon Kendra. Puede ver la configuración de un índice y leer cualquier mensaje de error si el estado muestra un índice «FAILED» para crearlo completamente.

```
aws kendra describe-index \  
  --id exampleindex1
```

Salida:

```
{  
  "CapacityUnits": {  
    "QueryCapacityUnits": 0,  
    "StorageCapacityUnits": 0  
  },  
  "CreatedAt": 2024-02-25T12:30:10+00:00,  
  "Description": "Example index 1 contains the first set of example documents",  
  "DocumentMetadataConfigurations": [  
    {  
      "Name": "_document_title",  
      "Relevance": {  
        "Importance": 8  
      },  
      "Search": {  
        "Displayable": true,  
        "Facetable": false,  
        "Searchable": true,  
        "Sortable": false  
      },  
      "Type": "STRING_VALUE"  
    },  
    {
```



```
    "Name": "_document_body",
    "Relevance": {
      "Importance": 5
    },
    "Search": {
      "Displayable": true,
      "Facetable": false,
      "Searchable": true,
      "Sortable": false
    },
    "Type": "STRING_VALUE"
  },
  {
    "Name": "_last_updated_at",
    "Relevance": {
      "Importance": 6,
      "Duration": "2628000s",
      "Freshness": true
    },
    "Search": {
      "Displayable": true,
      "Facetable": false,
      "Searchable": true,
      "Sortable": true
    },
    "Type": "DATE_VALUE"
  },
  {
    "Name": "department_custom_field",
    "Relevance": {
      "Importance": 7,
      "ValueImportanceMap": {
        "Human Resources" : 4,
        "Marketing and Sales" : 2,
        "Research and innvoation" : 3,
        "Admin" : 1
      }
    },
    "Search": {
      "Displayable": true,
      "Facetable": true,
      "Searchable": true,
      "Sortable": true
    },
  },
```

```

        "Type": "STRING_VALUE"
    }
],
"Edition": "DEVELOPER_EDITION",
"Id": "index1",
"IndexStatistics": {
    "FaqStatistics": {
        "IndexedQuestionAnswersCount": 10
    },
    "TextDocumentStatistics": {
        "IndexedTextBytes": 1073741824,
        "IndexedTextDocumentsCount": 1200
    }
},
"Name": "example index 1",
"RoleArn": "arn:aws:iam::my-account-id:role/KendraRoleForExampleIndex",
"ServerSideEncryptionConfiguration": {
    "KmsKeyId": "my-kms-key-id"
},
"Status": "ACTIVE",
"UpdatedAt": 1709163615,
"UserContextPolicy": "USER_TOKEN",
"UserTokenConfigurations": [
    {
        "JsonTokenTypeConfiguration": {
            "GroupAttributeField": "groupNameField",
            "UserNameAttributeField": "userNameField"
        }
    }
]
}

```

Para obtener más información, consulte [Introducción a un conector de índice y fuente de datos de Amazon Kendra](#) en la Guía para desarrolladores de Amazon Kendra.

- Para API obtener más información, consulte la Referencia [DescribeIndex](#) de AWS CLI comandos.

## update-data-source

En el siguiente ejemplo de código, se muestra cómo usar update-data-source.

## AWS CLI

Para actualizar un conector de fuente de datos de Amazon Kendra

A continuación, se `update-data-source` actualiza la configuración de un conector de fuente de datos de Amazon Kendra. Si la acción se realiza correctamente, el servicio no devuelve ningún resultado, el código de HTTP estado 200 o el código de AWS CLI devolución 0. Se puede utilizar `describe-data-source` para ver la configuración y el estado de un conector de fuente de datos.

```
aws kendra update-data-source \
  --id exampledatasource1 \
  --index-id exampleindex1 \
  --name "new name for example data source 1" \
  --description "new description for example data source 1" \
  --role-arn arn:aws:iam::my-account-id:role/KendraNewRoleForExampleDataSource \
  --configuration '{"TemplateConfiguration": {"Template": file://
s3schemanewconfig.json}}' \
  --custom-document-enrichment-configuration '{"PostExtractionHookConfiguration":
  {"LambdaArn": "arn:aws:iam::my-account-id:function/my-function-ocr-docs",
  "S3Bucket": "s3://my-s3-bucket/scanned-image-text-example-docs", "RoleArn":
  "arn:aws:iam:my-account-id:role/KendraNewRoleForCDE"}' \
  --language-code "es" \
  --schedule "0 0 18 ? * MON,WED,FRI *" \
  --vpc-configuration '{"SecurityGroupIds": ["sg-1234567890abcdef0"], "SubnetIds":
  ["subnet-1c234", "subnet-2b134"]}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Introducción a un conector de índice y fuente de datos de Amazon Kendra](#) en la Guía para desarrolladores de Amazon Kendra.

- Para API obtener más información, consulte la Referencia [UpdateDataSource](#) de AWS CLI comandos.

## update-index

En el siguiente ejemplo de código, se muestra cómo usar `update-index`.

## AWS CLI

Para actualizar un índice de Amazon Kendra

A continuación, se `update-index` actualiza la configuración de un índice de Amazon Kendra. Si la acción se realiza correctamente, el servicio no devuelve ningún resultado, el código de HTTP estado 200 o el código de AWS CLI devolución 0. Se puede utilizar `describe-index` para ver la configuración y el estado de un índice.

```
aws kendra update-index \
  --id enterpriseindex1 \
  --name "new name for Enterprise Edition index 1" \
  --description "new description for Enterprise Edition index 1" \
  --role-arn arn:aws:iam::my-account-id:role/KendraNewRoleForEnterpriseIndex \
  --capacity-units '{"QueryCapacityUnits": 2, "StorageCapacityUnits": 1}' \
  --document-metadata-configuration-updates '{"Name": "_document_title",  
"Relevance": {"Importance": 6}}, {"Name": "_last_updated_at", "Relevance":  
{"Importance": 8}}' \
  --user-context-policy "USER_TOKEN" \
  --user-token-configurations '{"JsonTokenTypeConfiguration":  
{"GroupAttributeField": "groupNameField", "UserNameAttributeField":  
"userNameField"}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Introducción a un conector de índice y fuente de datos de Amazon Kendra](#) en la Guía para desarrolladores de Amazon Kendra.

- Para API obtener más información, consulte la Referencia [UpdateIndex](#) de AWS CLI comandos.

## Ejemplos de Kinesis que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el AWS Command Line Interface uso de Kinesis.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **add-tags-to-stream**

En el siguiente ejemplo de código, se muestra cómo usar `add-tags-to-stream`.

#### AWS CLI

Para añadir etiquetas a un flujo de datos

En el siguiente `add-tags-to-stream` ejemplo, se asigna una etiqueta con la clave `samplekey` y el valor `example` al flujo especificado.

```
aws kinesis add-tags-to-stream \  
  --stream-name samplestream \  
  --tags samplekey=example
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Cómo [etiquetar sus transmisiones](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia [AddTagsToStream](#) de AWS CLI comandos.

### **create-stream**

En el siguiente ejemplo de código, se muestra cómo usar `create-stream`.

#### AWS CLI

Creación de un flujo de datos

En el siguiente ejemplo de `create-stream` se crea un flujo de datos denominado `samplestream` con 3 particiones.

```
aws kinesis create-stream \  
  --stream-name samplestream \  
  --shard-count 3
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de una secuencia](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte [CreateStream](#) la Referencia de AWS CLI comandos.

## decrease-stream-retention-period

En el siguiente ejemplo de código, se muestra cómo usar `decrease-stream-retention-period`.

### AWS CLI

Para reducir el período de retención del flujo de datos

El siguiente `decrease-stream-retention-period` ejemplo reduce el período de retención (el tiempo durante el cual se puede acceder a los registros de datos después de añadirlos a la transmisión) de una transmisión denominada `samplestream` a 48 horas.

```
aws kinesis decrease-stream-retention-period \  
  --stream-name samplestream \  
  --retention-period-hours 48
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cambiar el período de retención de datos](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte [DecreaseStreamRetentionPeriod](#) la Referencia de AWS CLI comandos.

## delete-stream

En el siguiente ejemplo de código, se muestra cómo usar `delete-stream`.

### AWS CLI

Eliminación de un flujo de datos

En el siguiente ejemplo de `delete-stream` se elimina el flujo de datos especificado.

```
aws kinesis delete-stream \  
  --stream-name samplestream
```

```
--stream-name samplestream
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminación de una secuencia](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte [DeleteStream](#) la Referencia de AWS CLI comandos.

## deregister-stream-consumer

En el siguiente ejemplo de código, se muestra cómo usar `deregister-stream-consumer`.

### AWS CLI

Para anular el registro de un consumidor de flujos de datos

En el siguiente `deregister-stream-consumer` ejemplo, se anula el registro del consumidor especificado del flujo de datos especificado.

```
aws kinesis deregister-stream-consumer \  
  --stream-arn arn:aws:kinesis:us-west-2:123456789012:stream/samplestream \  
  --consumer-name KinesisConsumerApplication
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Desarrollo de consumidores con una distribución mejorada mediante Kinesis Data API Streams en la](#) Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia de [DeregisterStreamConsumer](#) comandos AWS CLI .

## describe-limits

En el siguiente ejemplo de código, se muestra cómo usar `describe-limits`.

### AWS CLI

Para describir los límites de los fragmentos

En el siguiente `describe-limits` ejemplo, se muestran los límites y el uso de los fragmentos de la cuenta corriente AWS .

```
aws kinesis describe-limits
```

Salida:

```
{
  "ShardLimit": 500,
  "OpenShardCount": 29
}
```

Para obtener más información, consulte [Repartición de una transmisión en la Guía para desarrolladores de Amazon Kinesis Data Streams](#).

- Para API obtener más información, consulte Command [DescribeLimits](#)Reference AWS CLI .

## `describe-stream-consumer`

En el siguiente ejemplo de código, se muestra cómo usar `describe-stream-consumer`.

AWS CLI

Para describir a un consumidor de flujos de datos

El siguiente `describe-stream-consumer` ejemplo devuelve la descripción del consumidor especificado, registrado en el flujo de datos especificado.

```
aws kinesis describe-stream-consumer \
  --stream-arn arn:aws:kinesis:us-west-2:012345678912:stream/samplestream \
  --consumer-name KinesisConsumerApplication
```

Salida:

```
{
  "ConsumerDescription": {
    "ConsumerName": "KinesisConsumerApplication",
    "ConsumerARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream/
consumer/KinesisConsumerApplication:1572383852",
    "ConsumerStatus": "ACTIVE",
    "ConsumerCreationTimestamp": 1572383852.0,
  }
}
```



```
    "StreamARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream"
  }
}
```

Para obtener más información, consulte [Lectura de datos de Amazon Kinesis Data Streams](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia [DescribeStreamConsumer](#) de AWS CLI comandos.

## describe-stream-summary

En el siguiente ejemplo de código, se muestra cómo usar `describe-stream-summary`.

### AWS CLI

Para describir un resumen de un flujo de datos

El siguiente `describe-stream-summary` ejemplo proporciona una descripción resumida (sin la lista de fragmentos) del flujo de datos especificado.

```
aws kinesis describe-stream-summary \
  --stream-name samplestream
```

Salida:

```
{
  "StreamDescriptionSummary": {
    "StreamName": "samplestream",
    "StreamARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream",
    "StreamStatus": "ACTIVE",
    "RetentionPeriodHours": 48,
    "StreamCreationTimestamp": 1572297168.0,
    "EnhancedMonitoring": [
      {
        "ShardLevelMetrics": []
      }
    ],
    "EncryptionType": "NONE",
    "OpenShardCount": 3,
    "ConsumerCount": 0
  }
}
```

```
}
```

Para obtener más información, consulte [Creación y administración de secuencias](#) en la Guía para desarrolladores Amazon Kinesis Data Streams.

- Para API obtener más información, consulte [DescribeStreamSummary](#) la Referencia de AWS CLI comandos.

## describe-stream

En el siguiente ejemplo de código, se muestra cómo usar `describe-stream`.

### AWS CLI

Descripción de un flujo de datos

En el siguiente ejemplo de `describe-stream` se devuelven los detalles del flujo de datos especificado.

```
aws kinesis describe-stream \  
  --stream-name samplestream
```

Salida:

```
{  
  "StreamDescription": {  
    "Shards": [  
      {  
        "ShardId": "shardId-000000000000",  
        "HashKeyRange": {  
          "StartingHashKey": "0",  
          "EndingHashKey": "113427455640312821154458202477256070484"  
        },  
        "SequenceNumberRange": {  
          "StartingSequenceNumber":  
"49600871682957036442365024926191073437251060580128653314"  
        }  
      },  
      {  
        "ShardId": "shardId-000000000001",  
        "HashKeyRange": {  
          "StartingHashKey": "113427455640312821154458202477256070485",
```

```

        "EndingHashKey": "226854911280625642308916404954512140969"
    },
    "SequenceNumberRange": {
        "StartingSequenceNumber":
"49600871682979337187563555549332609155523708941634633746"
    }
},
{
    "ShardId": "shardId-000000000002",
    "HashKeyRange": {
        "StartingHashKey": "226854911280625642308916404954512140970",
        "EndingHashKey": "340282366920938463463374607431768211455"
    },
    "SequenceNumberRange": {
        "StartingSequenceNumber":
"49600871683001637932762086172474144873796357303140614178"
    }
}
],
"StreamARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream",
"StreamName": "samplestream",
"StreamStatus": "ACTIVE",
"RetentionPeriodHours": 24,
"EnhancedMonitoring": [
    {
        "ShardLevelMetrics": []
    }
],
"EncryptionType": "NONE",
"KeyId": null,
"StreamCreationTimestamp": 1572297168.0
}
}

```

Para obtener más información, consulte [Creación y administración de secuencias](#) en la Guía para desarrolladores Amazon Kinesis Data Streams.

- Para API obtener más información, consulte [DescribeStream](#) la Referencia de AWS CLI comandos.

## **disable-enhanced-monitoring**

En el siguiente ejemplo de código, se muestra cómo usar `disable-enhanced-monitoring`.

## AWS CLI

Para deshabilitar la supervisión mejorada de las métricas a nivel de fragmento

El siguiente `disable-enhanced-monitoring` ejemplo desactiva la supervisión mejorada del flujo de datos de Kinesis para las métricas a nivel de fragmento.

```
aws kinesis disable-enhanced-monitoring \  
  --stream-name samplestream --shard-level-metrics ALL
```

Salida:

```
{  
  "StreamName": "samplestream",  
  "CurrentShardLevelMetrics": [  
    "IncomingBytes",  
    "OutgoingRecords",  
    "IteratorAgeMilliseconds",  
    "IncomingRecords",  
    "ReadProvisionedThroughputExceeded",  
    "WriteProvisionedThroughputExceeded",  
    "OutgoingBytes"  
  ],  
  "DesiredShardLevelMetrics": []  
}
```

Para obtener más información, consulte [Monitorización de transmisiones en Amazon Kinesis Data Streams](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia [DisableEnhancedMonitoring](#) de AWS CLI comandos.

## **enable-enhanced-monitoring**

En el siguiente ejemplo de código, se muestra cómo usar `enable-enhanced-monitoring`.

## AWS CLI

Para permitir una supervisión mejorada de las métricas a nivel de fragmento

El siguiente `enable-enhanced-monitoring` ejemplo permite una supervisión mejorada del flujo de datos de Kinesis para las métricas a nivel de fragmento.

```
aws kinesis enable-enhanced-monitoring \
  --stream-name samplestream \
  --shard-level-metrics ALL
```

Salida:

```
{
  "StreamName": "samplestream",
  "CurrentShardLevelMetrics": [],
  "DesiredShardLevelMetrics": [
    "IncomingBytes",
    "OutgoingRecords",
    "IteratorAgeMilliseconds",
    "IncomingRecords",
    "ReadProvisionedThroughputExceeded",
    "WriteProvisionedThroughputExceeded",
    "OutgoingBytes"
  ]
}
```

Para obtener más información, consulte [Monitorización de transmisiones en Amazon Kinesis Data Streams](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia [EnableEnhancedMonitoring](#) de AWS CLI comandos.

## get-records

En el siguiente ejemplo de código, se muestra cómo usar `get-records`.

### AWS CLI

Obtención de registros de una partición

En el siguiente ejemplo de `get-records` se obtienen registros de datos de la partición de un flujo de datos de Kinesis mediante el iterador de partición especificado.

```
aws kinesis get-records \
  --shard-iterator AAAAAAAAAAAF7/0mWD7IuHj1yGv/TKuNgx2ukD5xipCY4cy4gU96orWwZwcSXh3K9tAmGYe0ZyLZrvzze0FVf9iN99hUPw/w/
```

```
b0YWYeefNvnf1DYt5XpDJghLKr3DzgzknkTmMymDP3R+3wRKeuEw6/kdxY2yKJH0veaiekaVc4N2VwK/
GvaGP2Hh9Fg7N++q0Adg6fIDQPt4p8RpavDbk+A4sL9SWGE1
```

Salida:

```
{
  "Records": [],
  "MillisBehindLatest": 80742000
}
```

Para obtener más información, consulte [Desarrollo de consumidores que utilizan Kinesis Data API Streams con AWS SDK la versión para Java en la Guía para](#) desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia [GetRecords](#) de AWS CLI comandos.

## get-shard-iterator

En el siguiente ejemplo de código, se muestra cómo usar `get-shard-iterator`.

AWS CLI

Para obtener un iterador de fragmentos

En el siguiente `get-shard-iterator` ejemplo, se utiliza el tipo de `AT_SEQUENCE_NUMBER` iterador de fragmentos y se genera un iterador de fragmentos para empezar a leer los registros de datos exactamente desde la posición indicada por el número de secuencia especificado.

```
aws kinesis get-shard-iterator \
  --stream-name samplestream \
  --shard-id shardId-000000000001 \
  --shard-iterator-type LATEST
```

Salida:

```
{
  "ShardIterator": "AAAAAAAAAAFEvJjIYI+3jw/4aqqH9FifJ+n48XWTh/
IFIsbILP6o5eDueD39NXNBfpZ10WL5K6ADXk8w+5H+Qhd9cFA9k268CPXCz/kebq1TGYI7Vy
+1UkA9BuN3xvATxMBGxRY3zYK05gqgvaIRn9408SqeEqwhigwZxNWxID3Ej7YYYcxQi8Q/fIrCjGAy/
n2r5Z9G864YpWDFn9upNNQAR/ii0WKs"
}
```

Para obtener más información, consulte [Desarrollo de consumidores que utilizan Kinesis Data API Streams con AWS SDK la versión para Java en la Guía para desarrolladores de Amazon Kinesis Data Streams](#).

- Para API obtener más información, consulte la Referencia [GetShardIterator](#) de AWS CLI comandos.

## increase-stream-retention-period

En el siguiente ejemplo de código, se muestra cómo usar `increase-stream-retention-period`.

### AWS CLI

Para aumentar el período de retención del flujo de datos

El siguiente `increase-stream-retention-period` ejemplo aumenta el período de retención (el tiempo durante el cual se puede acceder a los registros de datos después de agregarlos a la transmisión) de la transmisión especificada a 168 horas.

```
aws kinesis increase-stream-retention-period \  
  --stream-name samplestream \  
  --retention-period-hours 168
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cambiar el período de retención de datos](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte [IncreaseStreamRetentionPeriod](#) la Referencia de AWS CLI comandos.

## list-shards

En el siguiente ejemplo de código, se muestra cómo usar `list-shards`.

### AWS CLI

Para enumerar los fragmentos de un flujo de datos

En el siguiente `list-shards` ejemplo, se enumeran todos los fragmentos de la secuencia especificada, empezando por el fragmento cuyo identificador sigue inmediatamente al especificado. `exclusive-start-shard-id shardId-000000000000`

```
aws kinesis list-shards \  
  --stream-name samplestream \  
  --exclusive-start-shard-id shardId-000000000000
```

Salida:

```
{  
  "Shards": [  
    {  
      "ShardId": "shardId-000000000001",  
      "HashKeyRange": {  
        "StartingHashKey": "113427455640312821154458202477256070485",  
        "EndingHashKey": "226854911280625642308916404954512140969"  
      },  
      "SequenceNumberRange": {  
        "StartingSequenceNumber":  
"49600871682979337187563555549332609155523708941634633746"  
      }  
    },  
    {  
      "ShardId": "shardId-000000000002",  
      "HashKeyRange": {  
        "StartingHashKey": "226854911280625642308916404954512140970",  
        "EndingHashKey": "340282366920938463463374607431768211455"  
      },  
      "SequenceNumberRange": {  
        "StartingSequenceNumber":  
"49600871683001637932762086172474144873796357303140614178"  
      }  
    }  
  ]  
}
```

Para obtener más información, consulte Cómo [publicar fragmentos](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia [ListShards](#) de AWS CLI comandos.



## list-streams

En el siguiente ejemplo de código, se muestra cómo usar `list-streams`.

### AWS CLI

Creación de una lista de flujos de datos

En el siguiente ejemplo de `list-streams` se enumeran todos los flujos de datos activos de la cuenta y la región actuales.

```
aws kinesis list-streams
```

Salida:

```
{
  "StreamNames": [
    "samplestream",
    "samplestream1"
  ]
}
```

Para obtener más información, consulte [Visualización de secuencias](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte [ListStreams](#) la Referencia de AWS CLI comandos.

## list-tags-for-stream

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-stream`.

### AWS CLI

Para enumerar las etiquetas de un flujo de datos

En el siguiente `list-tags-for-stream` ejemplo, se enumeran las etiquetas adjuntas al flujo de datos especificado.

```
aws kinesis list-tags-for-stream \
  --stream-name samplestream
```

Salida:

```
{
  "Tags": [
    {
      "Key": "samplekey",
      "Value": "example"
    }
  ],
  "HasMoreTags": false
}
```

Para obtener más información, consulte Cómo [etiquetar sus transmisiones](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia [ListTagsForStream](#) de AWS CLI comandos.

## merge-shards

En el siguiente ejemplo de código, se muestra cómo usar merge-shards.

### AWS CLI

Para fusionar fragmentos

En el siguiente merge-shards ejemplo, se combinan dos fragmentos adyacentes IDs de shardId -000000000000 y shardId -000000000001 en el flujo de datos especificado y los combina en un solo fragmento.

```
aws kinesis merge-shards \  
  --stream-name samplestream \  
  --shard-to-merge shardId-000000000000 \  
  --adjacent-shard-to-merge shardId-000000000001
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Fusión de dos fragmentos](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia de [MergeShards](#) comandos AWS CLI .

## put-record

En el siguiente ejemplo de código, se muestra cómo usar `put-record`.

### AWS CLI

Escritura de un registro en un flujo de datos

En el siguiente ejemplo de `put-record` se escribe un único registro de datos en el flujo de datos especificado mediante la clave de partición especificada.

```
aws kinesis put-record \  
  --stream-name samplestream \  
  --data sampledatarecord \  
  --partition-key samplepartitionkey
```

Salida:

```
{  
  "ShardId": "shardId-000000000009",  
  "SequenceNumber": "49600902273357540915989931256901506243878407835297513618",  
  "EncryptionType": "KMS"  
}
```

Para obtener más información, consulte [Desarrollo de productores que utilicen Amazon Kinesis Data API Streams con para Java en AWS SDK la Guía para](#) desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia [PutRecord](#) de AWS CLI comandos.

## put-records

En el siguiente ejemplo de código, se muestra cómo usar `put-records`.

### AWS CLI

Para escribir varios registros en un flujo de datos

El siguiente `put-records` ejemplo escribe un registro de datos con la clave de partición especificada y otro registro de datos con una clave de partición diferente en una sola llamada.

```
aws kinesis put-records \  
  --stream-name samplestream \  
  --data sampledatarecord \  
  --partition-key samplepartitionkey
```

```
--stream-name samplestream \  
--  
records Data=blob1,PartitionKey=partitionkey1 Data=blob2,PartitionKey=partitionkey2
```

Salida:

```
{  
  "FailedRecordCount": 0,  
  "Records": [  
    {  
      "SequenceNumber":  
"49600883331171471519674795588238531498465399900093808706",  
      "ShardId": "shardId-000000000004"  
    },  
    {  
      "SequenceNumber":  
"49600902273357540915989931256902715169698037101720764562",  
      "ShardId": "shardId-000000000009"  
    }  
  ],  
  "EncryptionType": "KMS"  
}
```

Para obtener más información, consulte [Desarrollo de productores que utilicen Amazon Kinesis Data API Streams con para Java en AWS SDK la Guía para](#) desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia [PutRecords](#) de AWS CLI comandos.

## register-stream-consumer

En el siguiente ejemplo de código, se muestra cómo usar `register-stream-consumer`.

### AWS CLI

Para registrar un consumidor de flujo de datos

El siguiente `register-stream-consumer` ejemplo registra a un consumidor llamado `KinesisConsumerApplication` con el flujo de datos especificado.

```
aws kinesis register-stream-consumer \  
--stream-arn arn:aws:kinesis:us-west-2:012345678912:stream/samplestream \  
--
```

```
--consumer-name KinesisConsumerApplication
```

Salida:

```
{
  "Consumer": {
    "ConsumerName": "KinesisConsumerApplication",
    "ConsumerARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream/
consumer/KinesisConsumerApplication:1572383852",
    "ConsumerStatus": "CREATING",
    "ConsumerCreationTimestamp": 1572383852.0
  }
}
```

Para obtener más información, consulte [Desarrollo de consumidores con una distribución mejorada mediante Kinesis Data API Streams en la Guía para desarrolladores de Amazon Kinesis Data Streams](#).

- Para API obtener más información, consulte la Referencia de [RegisterStreamConsumer](#) comandos AWS CLI .

## **remove-tags-from-stream**

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags-from-stream`.

### AWS CLI

Para eliminar etiquetas de un flujo de datos

En el siguiente `remove-tags-from-stream` ejemplo, se quita la etiqueta con la clave especificada del flujo de datos especificado.

```
aws kinesis remove-tags-from-stream \  
  --stream-name samplestream \  
  --tag-keys samplekey
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Cómo [etiquetar sus transmisiones](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia [RemoveTagsFromStream](#) de AWS CLI comandos.

## split-shard

En el siguiente ejemplo de código, se muestra cómo usar `split-shard`.

### AWS CLI

Para dividir fragmentos

En el siguiente `split-shard` ejemplo, se divide el fragmento especificado en dos fragmentos nuevos mediante una nueva clave hash inicial de 10.

```
aws kinesis split-shard \  
  --stream-name samplestream \  
  --shard-to-split shardId-000000000000 \  
  --new-starting-hash-key 10
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [División de un fragmento](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia de [SplitShard](#) comandos AWS CLI .

## start-stream-encryption

En el siguiente ejemplo de código, se muestra cómo usar `start-stream-encryption`.

### AWS CLI

Para habilitar el cifrado del flujo de datos

El siguiente `start-stream-encryption` ejemplo habilita el cifrado del lado del servidor para el flujo especificado mediante la clave especificada AWS KMS.

```
aws kinesis start-stream-encryption \  
  --encryption-type KMS \  
  --key-id arn:aws:kms:us-west-2:012345678912:key/a3c4a7cd-728b-45dd-  
b334-4d3eb496e452 \  
  --stream-name samplestream
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Protección de datos en Amazon Kinesis Data Streams](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia [StartStreamEncryption](#) de AWS CLI comandos.

## stop-stream-encryption

En el siguiente ejemplo de código, se muestra cómo usar `stop-stream-encryption`.

AWS CLI

Para deshabilitar el cifrado del flujo de datos

En el siguiente `stop-stream-encryption` ejemplo, se deshabilita el cifrado del lado del servidor para la transmisión especificada mediante la clave especificada. AWS KMS

```
aws kinesis start-stream-encryption \  
  --encryption-type KMS \  
  --key-id arn:aws:kms:us-west-2:012345678912:key/a3c4a7cd-728b-45dd-  
b334-4d3eb496e452 \  
  --stream-name samplestream
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Protección de datos en Amazon Kinesis Data Streams](#) en la Guía para desarrolladores de Amazon Kinesis Data Streams.

- Para API obtener más información, consulte la Referencia [StopStreamEncryption](#) de AWS CLI comandos.

## update-shard-count

En el siguiente ejemplo de código, se muestra cómo usar `update-shard-count`.

AWS CLI

Para actualizar el recuento de fragmentos en un flujo de datos

En el siguiente `update-shard-count` ejemplo, se actualiza el recuento de fragmentos del flujo de datos especificado a 6. En este ejemplo, se utiliza una escala uniforme, que crea fragmentos del mismo tamaño.

```
aws kinesis update-shard-count \  
  --stream-name samplestream \  
  --scaling-type UNIFORM_SCALING \  
  --target-shard-count 6
```

Salida:

```
{  
  "StreamName": "samplestream",  
  "CurrentShardCount": 3,  
  "TargetShardCount": 6  
}
```

Para obtener más información, consulte [Repartición de una transmisión en la Guía para desarrolladores de Amazon Kinesis Data Streams](#).

- Para API obtener más información, consulte Command [UpdateShardCount](#)Reference AWS CLI .

## AWS KMS ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS KMS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)



## Acciones

### **cancel-key-deletion**

En el siguiente ejemplo de código, se muestra cómo usar `cancel-key-deletion`.

#### AWS CLI

Para cancelar la eliminación programada de una KMS clave gestionada por el cliente

El siguiente `cancel-key-deletion` ejemplo cancela la eliminación programada de una KMS clave gestionada por el cliente.

```
aws kms cancel-key-deletion \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Salida:

```
{  
  "KeyId": "arn:aws:kms:us-  
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab"  
}
```

Cuando el `cancel-key-deletion` comando se ejecuta correctamente, se cancela la eliminación programada. Sin embargo, el estado de la KMS clave es `Disabled`, por lo que no puede utilizarla en KMS operaciones criptográficas. Para restaurar su funcionalidad, utilice el `enable-key` comando.

Para obtener más información, consulte [Programar y cancelar la eliminación de claves](#) en la Guía para desarrolladores del Servicio de administración de AWS claves.

- Para API obtener más información, consulte [CancelKeyDeletion](#) la Referencia de AWS CLI comandos.

### **connect-custom-key-store**

En el siguiente ejemplo de código, se muestra cómo usar `connect-custom-key-store`.

#### AWS CLI

Para conectar un almacén de claves personalizado

En el siguiente `connect-custom-key-store` ejemplo, se vuelve a conectar el almacén de claves personalizadas especificado. Puede utilizar un comando como este para conectar un almacén de claves personalizado por primera vez o para volver a conectar un almacén de claves que estaba desconectado.

Puedes usar este comando para conectar un almacén de HSM claves de AWS Cloud o un almacén de claves externo.

```
aws kms connect-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0
```

El comando no devuelve ningún resultado. Para comprobar que se aplicó el comando, utilice el comando `describe-custom-key-stores`.

Para obtener información sobre cómo conectar un almacén de HSM claves AWS en la nube, consulte [Conexión y desconexión de un almacén de HSM claves en la AWS nube](#) en la Guía para desarrolladores del servicio de administración de AWS claves.

Para obtener información sobre cómo conectar un almacén de claves externo, consulte [Conectar y desconectar un almacén de claves externo](#) en la Guía para desarrolladores del servicio de administración de AWS claves.

- Para API obtener más información, consulte [ConnectCustomKeyStore](#) la Referencia de AWS CLI comandos.

## create-alias

En el siguiente ejemplo de código, se muestra cómo usar `create-alias`.

### AWS CLI

Para crear un alias para una KMS clave

El siguiente `create-alias` comando crea un alias con el nombre `example-alias` de la KMS clave identificada por el ID de clave `1234abcd-12ab-34cd-56ef-1234567890ab`.

Los nombres del alias no debe empezar por `alias/`. No utilice nombres de alias que comiencen por `alias/aws`; están reservados para que los usen AWS.

```
aws kms create-alias \  
  --alias-name alias/example-alias \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

```
--target-key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Este comando no devuelve ningún resultado. Para ver el nuevo alias, utilice el comando `list-aliases`.

Para obtener más información, consulte [Uso de alias](#) en la AWS Guía para desarrolladores de Key Management Service.

- Para API obtener más información, consulte [CreateAlias](#) la Referencia de AWS CLI comandos.

## create-custom-key-store

En el siguiente ejemplo de código, se muestra cómo usar `create-custom-key-store`.

### AWS CLI

Ejemplo 1: Para crear un almacén de HSM claves AWS en la nube

En el siguiente `create-custom-key-store` ejemplo, se crea un almacén de HSM claves AWS en la AWS nube respaldado por un HSM clúster de la nube con los parámetros necesarios. También puedes añadir el `custom-key-store-type` parameter with the default value: `AWS_CLOUDHSM`.

Para especificar la entrada de archivo para el `trust-anchor-certificate` comando en AWS CLI, se requiere el `file://` prefijo.

```
aws kms create-custom-key-store \  
  --custom-key-store-name ExampleCloudHSMKeyStore \  
  --cloud-hsm-cluster-id cluster-1a23b4cdefg \  
  --key-store-password kmsPswd \  
  --trust-anchor-certificate file://customerCA.crt
```

Salida:

```
{  
  "CustomKeyStoreId": cks-1234567890abcdef0  
}
```

Para obtener más información, consulte [Creación de un almacén de HSM claves en AWS la nube](#) en la Guía para desarrolladores del servicio de administración de AWS claves.

Ejemplo 2: Para crear un almacén de claves externo con conectividad de punto final público

El siguiente `create-custom-key-store` ejemplo crea un almacén de claves externo (XKS) con el que se comunica AWS KMS a través de Internet.

En este ejemplo, `XksProxyUriPath` utiliza un prefijo opcional `deexample-prefix`.

NOTE: Si utiliza la AWS CLI versión 1.0, ejecute el siguiente comando antes de especificar un parámetro con un HTTPS valor HTTP o, como el `XksProxyUriEndpoint` parámetro.

```
aws configure set cli_follow_urlparam false
```

De lo contrario, la AWS CLI versión 1.0 reemplaza el valor del parámetro por el contenido que se encuentra en esa URI dirección.

```
aws kms create-custom-key-store \  
  --custom-key-store-name ExamplePublicEndpointXKS \  
  --custom-key-store-type EXTERNAL_KEY_STORE \  
  --xks-proxy-connectivity PUBLIC_ENDPOINT \  
  --xks-proxy-uri-endpoint "https://myproxy.xks.example.com" \  
  --xks-proxy-uri-path "/example-prefix/kms/xks/v1" \  
  --xks-proxy-authentication-credential "AccessKeyId=ABCDE12345670EXAMPLE,  
RawSecretAccessKey=DXjSUawneL2fr6SKC7G25CNxTyWKE5PF9XX6H/u9pSo="
```

Salida:

```
{  
  "CustomKeyId": cks-2234567890abcdef0  
}
```

Para obtener más información, consulte [Creación de un almacén de claves externo](#) en la Guía para desarrolladores del Servicio de administración de AWS claves.

Ejemplo 3: Para crear un almacén de claves externo con conectividad a VPC Endpoint Service

En el siguiente `create-custom-key-store` ejemplo, se crea un almacén de claves externo (XKS) que utiliza un Amazon VPC Endpoint Service para comunicarse AWS KMS.

NOTE: Si utiliza la AWS CLI versión 1.0, ejecute el siguiente comando antes de especificar un parámetro con un HTTPS valor HTTP o, como el `XksProxyUriEndpoint` parámetro.

```
aws configure set cli_follow_urlparam false
```

De lo contrario, la AWS CLI versión 1.0 reemplaza el valor del parámetro por el contenido que se encuentra en esa URI dirección.

```
aws kms create-custom-key-store \  
  --custom-key-store-name ExampleVPCEndpointXKS \  
  --custom-key-store-type EXTERNAL_KEY_STORE \  
  --xks-proxy-connectivity VPC_ENDPOINT_SERVICE \  
  --xks-proxy-uri-endpoint "https://myproxy-private.xks.example.com" \  
  --xks-proxy-uri-path "/kms/xks/v1" \  
  --xks-proxy-vpc-endpoint-service-name "com.amazonaws.vpce.us-east-1.vpce-svc-example1" \  
  --xks-proxy-authentication-credential "AccessKeyId=ABCDE12345670EXAMPLE,RawSecretAccessKey=DXjSUawne12fr6SKC7G25CNxTyWKE5PF9XX6H/u9pSo="
```

Salida:

```
{  
  "CustomKeyId": cks-3234567890abcdef0  
}
```

Para obtener más información, consulte [Creación de un almacén de claves externo](#) en la Guía para desarrolladores del Servicio de administración de AWS claves.

- Para API obtener más información, consulte [CreateCustomKeyStore](#) la Referencia de AWS CLI comandos.

## create-grant

En el siguiente ejemplo de código, se muestra cómo usar create-grant.

### AWS CLI

#### Creación de una concesión

En el siguiente create-grant ejemplo, se crea una concesión que permite al exampleUser usuario utilizar el decrypt comando de la KMS tecla de 1234abcd-12ab-34cd-56ef-1234567890ab ejemplo. El entidad principal que se va a dar de baja es el rol adminRole. La concesión utiliza la limitación de concesión EncryptionContextSubset para permitir este permiso solo cuando el contexto de cifrado de la solicitud decrypt incluye un par clave-valor "Department": "IT".

```
aws kms create-grant \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --grantee-principal arn:aws:iam::123456789012:user/exampleUser \  
  --operations Decrypt \  
  --constraints EncryptionContextSubset={Department=IT} \  
  --retiring-principal arn:aws:iam::123456789012:role/adminRole
```

Salida:

```
{  
  "GrantId": "1a2b3c4d2f5e69f440bae30eaec9570bb1fb7358824f9ddfa1aa5a0dab1a59b2",  
  "GrantToken": "<grant token here>"  
}
```

Para ver información detallada sobre la concesión, utilice el comando `list-grants`.

Para obtener más información, consulte [Grants in AWS KMS](#) en la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte [CreateGrant](#) la Referencia de AWS CLI comandos.

## create-key

En el siguiente ejemplo de código, se muestra cómo usar `create-key`.

### AWS CLI

Ejemplo 1: Para crear una KMS clave gestionada por el cliente en AWS KMS

El siguiente `create-key` ejemplo crea una KMS clave de cifrado simétrica.

Para crear la KMS clave básica, una clave de cifrado simétrica, no es necesario especificar ningún parámetro. Los valores predeterminados de esos parámetros crean una clave de cifrado simétrica.

Como este comando no especifica una política de claves, la KMS clave obtiene la [política de claves predeterminada para las claves](#) creadas mediante programación. KMS Utilice el comando `get-key-policy` para ver la política de claves. Utilice el comando `put-key-policy` para cambiar la política de claves.

```
aws kms create-key
```

El `create-key` comando devuelve los metadatos de la clave, incluidos el ID de la clave y ARN la nueva KMS clave. Puede utilizar estos valores para identificar la KMS clave en otras AWS KMS operaciones. La salida no incluye las etiquetas. Para ver las etiquetas de una KMS clave, utilice el `list-resource-tags` command.

Salida:

```
{
  "KeyMetadata": {
    "AWSAccountId": "111122223333",
    "Arn": "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "CreationDate": "2017-07-05T14:04:55-07:00",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "Description": "",
    "Enabled": true,
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "SYMMETRIC_DEFAULT",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": false,
    "Origin": "AWS_KMS"
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ]
  }
}
```

Nota: El `create-key` comando no permite especificar un alias. Para crear un alias para la nueva KMS clave, utilice el `create-alias` comando.

Para obtener más información, consulte [Creación de claves](#) en la Guía para desarrolladores de AWS Key Management Service.

Ejemplo 2: Para crear una RSA KMS clave asimétrica para el cifrado y el descifrado

El siguiente `create-key` ejemplo crea una KMS clave que contiene un par de RSA claves asimétricas para el cifrado y el descifrado.

```
aws kms create-key \
  --key-spec RSA_4096 \
```

```
--key-usage ENCRYPT_DECRYPT
```

Salida:

```
{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333",
    "CreationDate": "2021-04-05T14:04:55-07:00",
    "CustomerMasterKeySpec": "RSA_4096",
    "Description": "",
    "Enabled": true,
    "EncryptionAlgorithms": [
      "RSAES_OAEP_SHA_1",
      "RSAES_OAEP_SHA_256"
    ],
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "RSA_4096",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": false,
    "Origin": "AWS_KMS"
  }
}
```

Para obtener más información, consulte [Claves asimétricas AWS KMS en](#) la Guía para desarrolladores del Servicio de administración de AWS claves.

Ejemplo 3: Para crear una KMS clave de curva elíptica asimétrica para la firma y la verificación

Para crear una KMS clave asimétrica que contenga un ECC par de claves de curva elíptica asimétrica () para la firma y la verificación. El `--key-usage` parámetro es obligatorio aunque `SIGN_VERIFY` sea el único valor válido para las claves. ECC KMS

```
aws kms create-key \
  --key-spec ECC_NIST_P521 \
  --key-usage SIGN_VERIFY
```

Salida:



```
{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333",
    "CreationDate": "2019-12-02T07:48:55-07:00",
    "CustomerMasterKeySpec": "ECC_NIST_P521",
    "Description": "",
    "Enabled": true,
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "ECC_NIST_P521",
    "KeyState": "Enabled",
    "KeyUsage": "SIGN_VERIFY",
    "MultiRegion": false,
    "Origin": "AWS_KMS",
    "SigningAlgorithms": [
      "ECDSA_SHA_512"
    ]
  }
}
```

Para obtener más información, consulte [Claves asimétricas AWS KMS en](#) la Guía para desarrolladores del Servicio de administración de AWS claves.

#### Ejemplo 4: Para crear una clave HMAC KMS

El siguiente `create-key` ejemplo crea una clave de 384 bits HMACKMS. El `GENERATE_VERIFY_MAC` valor del `--key-usage` parámetro es obligatorio aunque sea el único valor válido para HMAC KMS las claves.

```
aws kms create-key \
  --key-spec HMAC_384 \
  --key-usage GENERATE_VERIFY_MAC
```

Salida:

```
{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
```

```

    "AWSAccountId": "111122223333",
    "CreationDate": "2022-04-05T14:04:55-07:00",
    "CustomerMasterKeySpec": "HMAC_384",
    "Description": "",
    "Enabled": true,
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "HMAC_384",
    "KeyState": "Enabled",
    "KeyUsage": "GENERATE_VERIFY_MAC",
    "MacAlgorithms": [
        "HMAC_SHA_384"
    ],
    "MultiRegion": false,
    "Origin": "AWS_KMS"
}
}

```

Para obtener más información, consulte [HMACClas claves AWS KMS en](#) la Guía AWS para desarrolladores del Servicio de administración de claves.

Ejemplo 4: Para crear una clave principal KMS multirregional

En el siguiente ejemplo de `create-key` se crea una clave principal de cifrado simétrica de varias regiones. Como los valores predeterminados de todos los parámetros crean una clave de cifrado simétrica, solo se necesita el `--multi-region` parámetro para esta KMS clave. En AWS CLI, para indicar que un parámetro booleano es verdadero, simplemente especifique el nombre del parámetro.

```

aws kms create-key \
  --multi-region

```

Salida:

```

{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-west-2:111122223333:key/mrk-1234abcd12ab34cd56ef12345678990ab",
    "AWSAccountId": "111122223333",
    "CreationDate": "2021-09-02T016:15:21-09:00",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "Description": "",

```

```

    "Enabled": true,
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ],
    "KeyId": "mrk-1234abcd12ab34cd56ef12345678990ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "SYMMETRIC_DEFAULT",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": true,
    "MultiRegionConfiguration": {
      "MultiRegionKeyType": "PRIMARY",
      "PrimaryKey": {
        "Arn": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef12345678990ab",
        "Region": "us-west-2"
      },
      "ReplicaKeys": []
    },
    "Origin": "AWS_KMS"
  }
}

```

Para obtener más información, consulte [Asymmetric Keys in AWS KMS en](#) la Guía para desarrolladores del servicio de administración de AWS claves.

Ejemplo 5: Para crear una KMS clave para el material clave importado

En el siguiente `create-key` ejemplo se crea una KMS clave sin material clave. Cuando se complete la operación, puede importar su propio material clave a la KMS clave. Para crear esta KMS clave, defina el `--origin` parámetro en `EXTERNAL`.

```

aws kms create-key \
  --origin EXTERNAL

```

Salida:

```

{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333",

```

```

    "CreationDate": "2019-12-02T07:48:55-07:00",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "Description": "",
    "Enabled": false,
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ],
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "SYMMETRIC_DEFAULT",
    "KeyState": "PendingImport",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": false,
    "Origin": "EXTERNAL"
  }
}

```

Para obtener más información, consulte [Importación de material clave en AWS KMS claves](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.

Ejemplo 6: Para crear una KMS clave en un almacén de HSM claves en AWS la nube

El siguiente `create-key` ejemplo crea y crea una KMS clave en el almacén de HSM claves de AWS Cloud especificado. La operación crea la KMS clave y sus metadatos AWS KMS y crea el material clave en el HSM clúster de AWS Cloud asociado al almacén de claves personalizado. Los parámetros `--custom-key-store-id` y `--origin` son obligatorios.

```

aws kms create-key \
  --origin AWS_CLOUDHSM \
  --custom-key-store-id cks-1234567890abcdef0

```

Salida:

```

{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333",
    "CloudHsmClusterId": "cluster-1a23b4cdefg",
    "CreationDate": "2019-12-02T07:48:55-07:00",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "CustomKeyId": "cks-1234567890abcdef0",
  }
}

```

```

    "Description": "",
    "Enabled": true,
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ],
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "SYMMETRIC_DEFAULT",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": false,
    "Origin": "AWS_CLOUDHSM"
  }
}

```

Para obtener más información, consulta los [almacenes de HSM claves en la AWS nube](#) en la Guía para desarrolladores del servicio de administración de AWS claves.

Ejemplo 7: Para crear una KMS clave en un almacén de claves externo

El siguiente `create-key` ejemplo crea y crea una KMS clave en el almacén de claves externo especificado. Los parámetros `--custom-key-store-id`, `--origin` y `--xks-key-id` son obligatorios en este comando.

El parámetro `--xks-key-id` especifica el ID de una clave de cifrado simétrica existente en el administrador de claves externo. Esta clave sirve como material de clave externa para la KMS clave. El valor del `--origin` parámetro debe ser `EXTERNAL_KEY_STORE`. El `custom-key-store-id` parámetro debe identificar un almacén de claves externo que esté conectado a su proxy de almacén de claves externo.

```

aws kms create-key \
  --origin EXTERNAL_KEY_STORE \
  --custom-key-store-id cks-9876543210fedcba9 \
  --xks-key-id bb8562717f809024

```

Salida:

```

{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",

```

```

    "AWSAccountId": "111122223333",
    "CreationDate": "2022-12-02T07:48:55-07:00",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "CustomKeyStoreId": "cks-9876543210fedcba9",
    "Description": "",
    "Enabled": true,
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ],
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "SYMMETRIC_DEFAULT",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": false,
    "Origin": "EXTERNAL_KEY_STORE",
    "XksKeyConfiguration": {
      "Id": "bb8562717f809024"
    }
  }
}

```

Para obtener más información, consulte [Almacenes de claves externos](#) en la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte [CreateKey](#) la Referencia de AWS CLI comandos.

## decrypt

En el siguiente ejemplo de código, se muestra cómo usar `decrypt`.

### AWS CLI

Ejemplo 1: Para descifrar un mensaje cifrado con una KMS clave simétrica (Linux y macOS)

El siguiente ejemplo de `decrypt` comando muestra la forma recomendada de descifrar datos con el AWS CLI. Esta versión muestra cómo descifrar datos con una clave simétrica KMS.

Proporcione el texto cifrado de un archivo. En el valor del `--ciphertext-blob` parámetro, utilice el `fileb://` prefijo, que indica que debe leer los datos de un archivo CLI binario. Si el archivo no se encuentra en el directorio actual, escriba la ruta completa al archivo. Para obtener más información sobre la lectura AWS CLI de los valores de AWS CLI los parámetros

de un archivo, consulte Carga de parámetros desde un archivo < <https://docs.aws.amazon.com/cli/latest/userguide/cli-usage-parameters-file.html>> en la Guía del usuario de la interfaz de línea de AWS comandos y Mejores prácticas para los parámetros de archivos locales< <https://aws.amazon.com/blogs/developer/best-practices-for-local-file-parameters/>> en el blog de herramientas de línea de AWS comandos. Especifique la KMS clave para descifrar el texto cifrado. El `--key-id` parámetro no es necesario cuando se descifra con una KMS clave simétrica. AWS KMS puede obtener el identificador de KMS clave de la clave que se utilizó para cifrar los datos de los metadatos del texto cifrado. Sin embargo, siempre se recomienda especificar la KMS clave que se está utilizando. Esta práctica garantiza el uso de la KMS clave deseada y evita que se descifre inadvertidamente un texto cifrado con una KMS clave en la que no se confíe. Solicite el resultado en texto plano como un valor de texto. El `--query Plaintext` El parámetro `--output` devuelve la salida como texto. Decodifique en base64 el texto sin formato y guárdelo en un archivo. El siguiente ejemplo canaliza (`|`) el valor del parámetro `Plaintext` a la utilidad `Base64`, que lo decodifica. A continuación, redirige (`>`) la salida descodificada al archivo `ExamplePlaintext`.

Antes de ejecutar este comando, sustituya el identificador de clave de ejemplo por un identificador de clave válido de su cuenta. AWS

```
aws kms decrypt \  
  --ciphertext-blob fileb://ExampleEncryptedFile \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --output text \  
  --query Plaintext | base64 \  
  --decode > ExamplePlaintextFile
```

Este comando no genera ninguna salida. La salida del comando `decrypt` se descodifica en base64 y se guarda en un archivo.

Para obtener más información, consulte [Descifrar](#) en la API referencia del servicio de administración de AWS claves.

Ejemplo 2: Para descifrar un mensaje cifrado con una KMS clave simétrica (línea de comandos de Windows)

El ejemplo siguiente es el mismo que el anterior, excepto que usa la utilidad `certutil` para descodificar en Base64 los datos de texto sin formato. Como se muestra en los siguientes ejemplos, este procedimiento requiere dos comandos.

Antes de ejecutar este comando, sustituya el identificador de clave de ejemplo por un identificador de clave válido de su AWS cuenta.

```
aws kms decrypt ^
  --ciphertext-blob fileb://ExampleEncryptedFile ^
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab ^
  --output text ^
  --query Plaintext > ExamplePlaintextFile.base64
```

Ejecute el comando `certutil`.

```
certutil -decode ExamplePlaintextFile.base64 ExamplePlaintextFile
```

Salida:

```
Input Length = 18
Output Length = 12
CertUtil: -decode command completed successfully.
```

Para obtener más información, consulte [Descifrar](#) en la APIreferencia del servicio de administración de AWS claves.

Ejemplo 3: Para descifrar un mensaje cifrado con una KMS clave asimétrica (Linux y macOS)

El siguiente ejemplo de `decrypt` comando muestra cómo descifrar los datos cifrados con una RSA clave asimétrica. KMS

Cuando se utiliza una KMS clave asimétrica, es obligatorio el `encryption-algorithm` parámetro, que especifica el algoritmo utilizado para cifrar el texto sin formato.

Antes de ejecutar este comando, sustituya la ID de clave de ejemplo por una ID de clave válida de su cuenta. AWS

```
aws kms decrypt \
  --ciphertext-blob fileb://ExampleEncryptedFile \
  --key-id 0987dcba-09fe-87dc-65ba-ab0987654321 \
  --encryption-algorithm RSAES_OAEP_SHA_256 \
  --output text \
  --query Plaintext | base64 \
  --decode > ExamplePlaintextFile
```



Este comando no genera ninguna salida. La salida del comando `decrypt` se descodifica en base64 y se guarda en un archivo.

Para obtener más información, consulte [Asymmetric Keys in AWS KMS en](#) la Guía para desarrolladores del servicio de administración de AWS claves.

- Para API obtener más información, consulte [Decrypt](#) in AWS CLI Command Reference.

## **delete-alias**

En el siguiente ejemplo de código, se muestra cómo usar `delete-alias`.

### AWS CLI

Para eliminar un alias AWS KMS

En el siguiente ejemplo de `delete-alias` se elimina el alias `alias/example-alias`. El nombre de alias debe empezar por `alias/`.

```
aws kms delete-alias \  
  --alias-name alias/example-alias
```

Este comando no genera ninguna salida. Utilice el comando `list-aliases` para encontrar el alias.

Para obtener más información, consulte [Eliminación de un alias](#) en la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte [DeleteAlias](#) la Referencia de AWS CLI comandos.

## **delete-custom-key-store**

En el siguiente ejemplo de código, se muestra cómo usar `delete-custom-key-store`.

### AWS CLI

Para eliminar un almacén de claves personalizado

En el siguiente `delete-custom-key-store` ejemplo, se elimina el almacén de claves personalizado especificado.

La eliminación de un almacén de HSM claves de AWS Cloud no afecta al HSM clúster de Cloud asociado. La eliminación de un almacén de claves externo no afecta al proxy del almacén de claves externo asociado, al administrador de claves externo ni a las claves externas.

NOTE: Antes de poder eliminar un almacén de claves personalizado, debe programar la eliminación de todas KMS las claves del almacén de claves personalizado y, a continuación, esperar a que se eliminen esas KMS claves. A continuación, debe desconectar el almacén de claves personalizado. Para obtener ayuda para encontrar las KMS claves en su almacén de claves personalizado, consulte [Eliminar un almacén de HSM claves en AWS la nube \(API\)](#) en la Guía para desarrolladores del servicio de administración de AWS claves.

```
delete-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0
```

El comando no devuelve ningún resultado. Para comprobar que se ha eliminado el almacén de claves personalizado, utilice el `describe-custom-key-stores` comando.

Para obtener información sobre cómo eliminar un almacén de HSM claves AWS en la nube, consulte [Eliminar un almacén de HSM claves en la AWS nube](#) en la Guía para desarrolladores del servicio de administración de AWS claves.

Para obtener información sobre la eliminación de almacenes de claves externos, consulte [Eliminar un almacén de claves externo](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.

- Para API obtener más información, consulte [DeleteCustomKeyStore](#) la Referencia de AWS CLI comandos.

## **delete-imported-key-material**

En el siguiente ejemplo de código, se muestra cómo usar `delete-imported-key-material`.

### AWS CLI

Para eliminar el material clave importado de una KMS clave

En el siguiente `delete-imported-key-material` ejemplo, se elimina el material clave que se ha importado a una KMS clave.

```
aws kms delete-imported-key-material \  
  --key-id   
  --key-id   
  --key-id
```

```
--key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Este comando no genera ninguna salida. Para comprobar que se ha eliminado el material clave, utilice el `describe-key` comando para buscar un estado clave igual `PendingImport` o `PendingDeletion`.

Para obtener más información, consulte [Eliminar el material clave importado](https://docs.aws.amazon.com/kms/latest/developerguide/importing-keys-delete-key-material.html) < <https://docs.aws.amazon.com/kms/latest/developerguide/importing-keys-delete-key-material.html> > en la Guía para desarrolladores de Key Management Service.AWS

- Para obtener API más información, [DeleteImportedKeyMaterial](#) consulte AWS CLI la Referencia de comandos.

## describe-custom-key-stores

En el siguiente ejemplo de código, se muestra cómo usar `describe-custom-key-stores`.

### AWS CLI

Ejemplo 1: Para obtener detalles sobre un almacén de HSM claves AWS en la nube

En el siguiente `describe-custom-key-store` ejemplo, se muestran detalles sobre el almacén de HSM claves de AWS Cloud especificado. El comando es el mismo para todos los tipos de almacenes de claves personalizados, pero el resultado difiere según el tipo de almacén de claves y, en el caso de un almacén de claves externo, su opción de conectividad.

De forma predeterminada, este comando muestra información sobre todos los almacenes de claves personalizados de la cuenta y la región. Para mostrar información sobre un almacén de claves personalizadas concreto, utilice el `custom-key-store-id` parámetro `custom-key-store-name` o.

```
aws kms describe-custom-key-stores \  
  --custom-key-store-name ExampleCloudHSMKeyStore
```

El resultado de este comando incluye detalles útiles sobre el almacén de HSM claves de AWS Cloud, incluido su estado de conexión (`ConnectionState`). Si el estado de la conexión es `FAILED`, el resultado incluye un `ConnectionErrorCode` campo que describe el problema.

Salida:

```
{
```

```
"CustomKeyStores": [  
  {  
    "CloudHsmClusterId": "cluster-1a23b4cdefg",  
    "ConnectionState": "CONNECTED",  
    "CreationDate": "2022-04-05T14:04:55-07:00",  
    "CustomKeyStoreId": "cks-1234567890abcdef0",  
    "CustomKeyStoreName": "ExampleExternalKeyStore",  
    "TrustAnchorCertificate": "<certificate appears here>"  
  }  
]  
}
```

Para obtener más información, consulte [Visualización de un almacén de HSM claves en la AWS nube](#) en la Guía para desarrolladores del servicio de administración de AWS claves.

Ejemplo 2: Para obtener detalles sobre un almacén de claves externo con conectividad de punto final público

En el siguiente `describe-custom-key-store` ejemplo, se muestran detalles sobre el almacén de claves externo especificado. El comando es el mismo para todos los tipos de almacenes de claves personalizados, pero el resultado difiere según el tipo de almacén de claves y, en el caso de un almacén de claves externo, su opción de conectividad.

De forma predeterminada, este comando muestra información sobre todos los almacenes de claves personalizados de la cuenta y la región. Para mostrar información sobre un almacén de claves personalizadas concreto, utilice el `custom-key-store-id` parámetro `custom-key-store-name` o.

```
aws kms describe-custom-key-stores \  
  --custom-key-store-id cks-9876543210fedcba9
```

El resultado de este comando incluye detalles útiles sobre el almacén de claves externo, incluido su estado de conexión (`ConnectionState`). Si el estado de la conexión es `FAILED`, el resultado incluye un `ConnectionErrorCode` campo que describe el problema.

Salida:

```
{  
  "CustomKeyStores": [  
    {  
      "CustomKeyStoreId": "cks-9876543210fedcba9",
```

```

    "CustomKeyStoreName": "ExampleXKS",
    "ConnectionState": "CONNECTED",
    "CreationDate": "2022-12-02T07:48:55-07:00",
    "CustomKeyStoreType": "EXTERNAL_KEY_STORE",
    "XksProxyConfiguration": {
      "AccessKeyId": "ABCDE12345670EXAMPLE",
      "Connectivity": "PUBLIC_ENDPOINT",
      "UriEndpoint": "https://myproxy.xks.example.com",
      "UriPath": "/example-prefix/kms/xks/v1"
    }
  ]
}

```

Para obtener más información, consulte [Visualización de un almacén de claves externo](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.

Ejemplo 3: Para obtener detalles sobre un almacén de claves externo con conectividad a VPC Endpoint Service

En el siguiente `describe-custom-key-store` ejemplo, se muestran detalles sobre el almacén de claves externo especificado. El comando es el mismo para todos los tipos de almacenes de claves personalizados, pero el resultado difiere según el tipo de almacén de claves y, en el caso de un almacén de claves externo, su opción de conectividad.

De forma predeterminada, este comando muestra información sobre todos los almacenes de claves personalizados de la cuenta y la región. Para mostrar información sobre un almacén de claves personalizadas concreto, utilice el `custom-key-store-id` parámetro `custom-key-store-name` o.

```

aws kms describe-custom-key-stores \
  --custom-key-store-id cks-2234567890abcdef0

```

El resultado de este comando incluye detalles útiles sobre el almacén de claves externo, incluido su estado de conexión (`ConnectionState`). Si el estado de la conexión es `FAILED`, el resultado incluye un `ConnectionErrorCode` campo que describe el problema.

Salida:

```

{
  "CustomKeyStores": [

```

```

    {
      "CustomKeyId": "cks-3234567890abcdef0",
      "CustomKeyName": "ExampleVPCEExternalKeyStore",
      "ConnectionState": "CONNECTED",
      "CreationDate": "2022-12-22T07:48:55-07:00",
      "CustomKeyType": "EXTERNAL_KEY_STORE",
      "XksProxyConfiguration": {
        "AccessKeyId": "ABCDE12345670EXAMPLE",
        "Connectivity": "VPC_ENDPOINT_SERVICE",
        "UriEndpoint": "https://myproxy-private.xks.example.com",
        "UriPath": "/kms/xks/v1",
        "VpcEndpointServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-
example1"
      }
    }
  ]
}

```

Para obtener más información, consulte [Visualización de un almacén de claves externo](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.

- Para API obtener más información, consulte [DescribeCustomKeyStores](#) la Referencia de AWS CLI comandos.

## describe-key

En el siguiente ejemplo de código, se muestra cómo usar `describe-key`.

### AWS CLI

Ejemplo 1: Para encontrar información detallada sobre una KMS clave

En el siguiente `describe-key` ejemplo, se obtiene información detallada sobre la clave AWS gestionada de Amazon S3 en la cuenta y la región de ejemplo. Puede usar este comando para obtener detalles sobre las claves AWS administradas y las claves administradas por el cliente.

Para especificar la KMS clave, utilice el `key-id` parámetro. En este ejemplo se utiliza un valor de nombre de alias, pero puede utilizar un identificador de clave, una claveARN, un nombre de alias o un alias ARN en este comando.

```

aws kms describe-key \
  --key-id alias/aws/s3

```

Salida:

```
{
  "KeyMetadata": {
    "AWSAccountId": "846764612917",
    "KeyId": "b8a9477d-836c-491f-857e-07937918959b",
    "Arn": "arn:aws:kms:us-west-2:846764612917:key/
b8a9477d-836c-491f-857e-07937918959b",
    "CreationDate": 2017-06-30T21:44:32.140000+00:00,
    "Enabled": true,
    "Description": "Default KMS key that protects my S3 objects when no other
key is defined",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "KeyState": "Enabled",
    "Origin": "AWS_KMS",
    "KeyManager": "AWS",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ]
  }
}
```

Para obtener más información, consulte [Visualización de claves](#) en la Guía para desarrolladores de AWS Key Management Service.

Ejemplo 2: Para obtener detalles sobre una clave RSA asimétrica KMS

En el siguiente `describe-key` ejemplo, se obtiene información detallada sobre una RSA KMS clave asimétrica utilizada para la firma y la verificación.

```
aws kms describe-key \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Salida:

```
{
  "KeyMetadata": {
    "AWSAccountId": "111122223333",
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "CreationDate": "2019-12-02T19:47:14.861000+00:00",
```

```

    "CustomerMasterKeySpec": "RSA_2048",
    "Enabled": false,
    "Description": "",
    "KeyState": "Disabled",
    "Origin": "AWS_KMS",
    "MultiRegion": false,
    "KeyManager": "CUSTOMER",
    "KeySpec": "RSA_2048",
    "KeyUsage": "SIGN_VERIFY",
    "SigningAlgorithms": [
      "RSASSA_PKCS1_V1_5_SHA_256",
      "RSASSA_PKCS1_V1_5_SHA_384",
      "RSASSA_PKCS1_V1_5_SHA_512",
      "RSASSA_PSS_SHA_256",
      "RSASSA_PSS_SHA_384",
      "RSASSA_PSS_SHA_512"
    ]
  }
}

```

### Ejemplo 3: Obtención de detalles sobre una clave de réplica de varias regiones

En el siguiente ejemplo de `describe-key` se obtienen los metadatos de una clave de réplica de varias regiones. Esta clave de varias regiones es una clave de cifrado simétrica. La salida de un comando `describe-key` para cualquier clave de varias regiones devuelve información sobre la clave principal y todas sus réplicas.

```

aws kms describe-key \
  --key-id arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab

```

Salida:

```

{
  "KeyMetadata": {
    "MultiRegion": true,
    "AWSAccountId": "111122223333",
    "Arn": "arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
    "CreationDate": "2021-06-28T21:09:16.114000+00:00",
    "Description": "",
    "Enabled": true,

```



```

    "KeyId": "mrk-1234abcd12ab34cd56ef1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "Origin": "AWS_KMS",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ],
    "MultiRegionConfiguration": {
      "MultiRegionKeyType": "PRIMARY",
      "PrimaryKey": {
        "Arn": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "Region": "us-west-2"
      },
      "ReplicaKeys": [
        {
          "Arn": "arn:aws:kms:eu-west-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
          "Region": "eu-west-1"
        },
        {
          "Arn": "arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
          "Region": "ap-northeast-1"
        },
        {
          "Arn": "arn:aws:kms:sa-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
          "Region": "sa-east-1"
        }
      ]
    }
  }
}
}
}

```

#### Ejemplo 4: Para obtener detalles sobre una clave HMAC KMS

En el siguiente `describe-key` ejemplo se obtiene información detallada sobre una HMAC KMS clave.

```
aws kms describe-key \
```

```
--key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Salida:

```
{
  "KeyMetadata": {
    "AWSAccountId": "123456789012",
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "Arn": "arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "CreationDate": "2022-04-03T22:23:10.194000+00:00",
    "Enabled": true,
    "Description": "Test key",
    "KeyUsage": "GENERATE_VERIFY_MAC",
    "KeyState": "Enabled",
    "Origin": "AWS_KMS",
    "KeyManager": "CUSTOMER",
    "CustomerMasterKeySpec": "HMAC_256",
    "MacAlgorithms": [
      "HMAC_SHA_256"
    ],
    "MultiRegion": false
  }
}
```

- Para API obtener más información, consulte [DescribeKey](#) la Referencia de AWS CLI comandos.

## disable-key-rotation

En el siguiente ejemplo de código, se muestra cómo usar `disable-key-rotation`.

### AWS CLI

Para deshabilitar la rotación automática de una KMS tecla

El siguiente `disable-key-rotation` ejemplo desactiva la rotación automática de una KMS clave gestionada por el cliente. Para volver a activar la rotación automática, utilice el `enable-key-rotation` comando.

```
aws kms disable-key-rotation \
  --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
```

Este comando no genera ninguna salida. Para comprobar que la rotación automática de la KMS tecla está desactivada, utilice el `get-key-rotation-status` comando.

Para obtener más información, consulte [Teclas giratorias](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.

- Para API obtener más información, consulte [DisableKeyRotation](#) la Referencia de AWS CLI comandos.

## **disable-key**

En el siguiente ejemplo de código, se muestra cómo usar `disable-key`.

### AWS CLI

Para deshabilitar temporalmente una KMS tecla

En el siguiente ejemplo, se utiliza el `disable-key` comando para deshabilitar una KMS clave gestionada por el cliente. Para volver a activar la KMS clave, utilice el `enable-key` comando.

```
aws kms disable-key \  
--key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Habilitar y deshabilitar claves](#) en la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte [DisableKey](#) la Referencia de AWS CLI comandos.

## **disconnect-custom-key-store**

En el siguiente ejemplo de código, se muestra cómo usar `disconnect-custom-key-store`.

### AWS CLI

Para desconectar un almacén de claves personalizado

En el siguiente `disconnect-custom-key-store` ejemplo, se desconecta un almacén de claves personalizado de su HSM clúster de AWS nube. Puedes desconectar un almacén de claves para solucionar un problema, actualizar su configuración o impedir que las claves del almacén de KMS claves se utilicen en operaciones criptográficas.

Este comando es el mismo para todos los almacenes de claves personalizados, incluidos los almacenes de claves AWS en la nube y los almacenes de HSM claves externos.

Antes de ejecutar este comando, reemplace el ID del almacén de claves personalizado de ejemplo por uno válido.

```
$ aws kms disconnect-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0
```

Este comando no produce ningún resultado. Compruebe que el comando ha sido efectivo y utilícelo. `describe-custom-key-stores`

Para obtener más información sobre cómo desconectar un almacén de HSM claves AWS en la nube, consulte [Conexión y desconexión de un almacén de HSM claves en AWS la nube en la Guía para desarrolladores](#) del servicio de administración de AWS claves.

Para obtener más información sobre cómo desconectar un almacén de claves externo, consulte [Conectar y desconectar un almacén de claves externo](#) en la Guía para desarrolladores del servicio de administración de AWS claves.

- Para API obtener más información, consulte la Referencia [DisconnectCustomKeyStore](#) de AWS CLI comandos.

## **enable-key-rotation**

En el siguiente ejemplo de código, se muestra cómo usar `enable-key-rotation`.

### AWS CLI

Para activar la rotación automática de una KMS tecla

El siguiente `enable-key-rotation` ejemplo permite la rotación automática de una KMS clave gestionada por el cliente con un período de rotación de 180 días. La KMS clave se rotará un año (aproximadamente 365 días) a partir de la fecha en que se complete este comando y cada año a partir de entonces.

El `--key-id` parámetro identifica la KMS clave. En este ejemplo se utiliza un ARN valor clave, pero se puede utilizar el identificador de clave o el ARN de la KMS clave. El `--rotation-period-in-days` parámetro especifica el número de días entre cada fecha de rotación. Especifique un valor entre 90 y 2560 días. Si no se especifica ningún valor, el valor predeterminado es 365 días.

```
aws kms enable-key-rotation \  
  --key-id arn:aws:kms:us-  
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab \  
  --rotation-period-in-days 180
```

Este comando no genera ninguna salida. Para comprobar que la KMS clave está habilitada, utilice el `get-key-rotation-status` comando.

Para obtener más información, consulte [Teclas giratorias](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.

- Para API obtener más información, consulte [EnableKeyRotation](#) la Referencia de AWS CLI comandos.

## enable-key

En el siguiente ejemplo de código, se muestra cómo usar `enable-key`.

### AWS CLI

Para activar una KMS clave

En el siguiente ejemplo de `enable-key` se habilita una clave administrada por el cliente. Puede utilizar un comando como este para activar una KMS tecla que haya desactivado temporalmente mediante el `disable-key` comando. También puede usarlo para habilitar una KMS tecla que esté deshabilitada porque estaba programada para su eliminación y la eliminación se canceló.

Para especificar la KMS clave, utilice el `key-id` parámetro. En este ejemplo se utiliza un valor de identificador de clave, pero puede utilizar un identificador o un ARN valor de clave en este comando.

Antes de ejecutar este comando, reemplace el ID de claves de ejemplo por uno válido.

```
aws kms enable-key \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Este comando no genera ninguna salida. Para comprobar que la KMS clave está habilitada, utilice el `describe-key` comando. Consulte los valores de los campos `KeyState` y `Enabled` en la salida `describe-key`.

Para obtener más información, consulte [Habilitar y deshabilitar claves](#) en la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte [EnableKey](#) la Referencia de AWS CLI comandos.

## encrypt

En el siguiente ejemplo de código, se muestra cómo usar encrypt.

### AWS CLI

Ejemplo 1: Cifrado del contenido de un archivo en Linux o MacOS

El siguiente encrypt comando muestra la forma recomendada de cifrar datos con el AWS CLI.

```
aws kms encrypt \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --plaintext fileb://ExamplePlaintextFile \  
  --output text \  
  --query CiphertextBlob | base64 \  
  --decode > ExampleEncryptedFile
```

El comando hace varias cosas:

Utiliza el parámetro `--plaintext` para indicar los datos que se van a cifrar. El valor de este parámetro debe estar codificado en base64. El valor del `plaintext` parámetro debe estar codificado en base64 o debe utilizar el `fileb://` prefijo, que le indica que lea los datos binarios del archivo. Si el AWS CLI archivo no está en el directorio actual, escriba la ruta completa del archivo. Por ejemplo: `fileb:///var/tmp/ExamplePlaintextFile` o `fileb://C:\Temp\ExamplePlaintextFile`. Para obtener más información sobre la lectura de los valores de los AWS CLI parámetros de un archivo, consulte [Carga de parámetros desde un archivo](#) en la AWS Guía del usuario de la interfaz de línea de AWS comandos y en el blog [Prácticas recomendadas para los parámetros de archivos locales](#) en la herramienta de línea de comandos. Utiliza `--query` los parámetros `--output` y para controlar la salida del comando. Estos parámetros extraen los datos cifrados, denominados texto cifrado, de la salida del comando. Para obtener más información sobre el control de la salida, consulte [Control Resultado del comando](#) en la guía del usuario de la interfaz de línea de AWS comandos. Utiliza la `base64` utilidad para decodificar el resultado extraído en datos binarios. El texto cifrado que devuelve un `encrypt` comando correcto es texto codificado en base64. Debe decodificar este texto AWS CLI para poder utilizarlo para descifrarlo. Guarda el texto cifrado binario en un archivo. La parte final del comando `()` guarda

el texto cifrado binario en un archivo para facilitar el descifrado. > `ExampleEncryptedFile`  
 Para ver un comando de ejemplo que utiliza el `aws kms decrypt` para descifrar datos, consulte los ejemplos de descifrado. AWS CLI

### Ejemplo 2: Utilización del AWS CLI para cifrar datos en Windows

Este ejemplo es igual al anterior, excepto que utiliza la herramienta `certutil` en lugar de `base64`. Como se muestra en el siguiente ejemplo, este procedimiento requiere dos comandos.

```
aws kms encrypt \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --plaintext fileb://ExamplePlaintextFile \
  --output text \
  --query CiphertextBlob > C:\Temp\ExampleEncryptedFile.base64

certutil -decode C:\Temp\ExampleEncryptedFile.base64 C:\Temp\ExampleEncryptedFile
```

### Ejemplo 3: Cifrar con una clave asimétrica KMS

El siguiente `encrypt` comando muestra cómo cifrar texto sin formato con una clave asimétrica. KMS El parámetro `--encryption-algorithm` es obligatorio. Como en todos los `encrypt` CLI comandos, el `plaintext` parámetro debe estar codificado en `base64` o debe utilizarse el `fileb://` prefijo, que indica que debe leer los datos binarios del archivo AWS CLI.

```
aws kms encrypt \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --encryption-algorithm RSAES_OAEP_SHA_256 \
  --plaintext fileb://ExamplePlaintextFile \
  --output text \
  --query CiphertextBlob | base64 \
  --decode > ExampleEncryptedFile
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [Cifrar](#) en la referencia de comandos.AWS CLI

## generate-data-key-pair-without-plaintext

En el siguiente ejemplo de código, se muestra cómo usar `generate-data-key-pair-without-plaintext`.

## AWS CLI

Para generar un par de ECC NIST claves de datos asimétricas P384

El siguiente `generate-data-key-pair-without-plaintext` ejemplo solicita un par de claves ECC NIST P384 para su uso fuera de AWS.

El comando devuelve una clave pública de texto simple y una copia de la clave privada cifrada con la clave especificada KMS. No devuelve una clave privada de texto simple. Puede almacenar de forma segura la clave privada cifrada con los datos cifrados y llamar AWS KMS para descifrar la clave privada cuando necesite utilizarla.

Para solicitar un key pair de claves de datos asimétricas ECC NIST P384, utilice el `key-pair-spec` parámetro con un valor de `ECC_NIST_P384`

La KMS clave que especifique debe ser una KMS clave de cifrado simétrica, es decir, una KMS clave con un `KeySpec` valor de `SYMMETRIC_DEFAULT`

NOTE: Los valores de la salida de este ejemplo están truncados para mostrarlos.

```
aws kms generate-data-key-pair-without-plaintext \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --key-pair-spec ECC_NIST_P384
```

Salida:

```
{  
  "PrivateKeyCiphertextBlob": "AQIDAHi6LtupRpdK12aJTzkk6Fbh0tQkM1QJJH3PdtHvS/y  
+hAFFxmiD134doUDzMGmfCEtcAAAHaTCCB2UGCSqGSIb3DQEHbqCCB1...",  
  "PublicKey":  
  "MIIBojANBgkqhkiG9w0BAQEFAAOCAY8AMIIBigKCAYEA3A3eGMyPrivSn7+Ld1JE1oUoQV5HpEuHAVbd0yND  
+NmYDH/mL10SIEuLrcdZ5hrMH4pk83r401...",  
  "KeyId": "arn:aws:kms:us-  
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",  
  "KeySpec": "ECC_NIST_P384"  
}
```

Los valores `PublicKey` y `PrivateKeyCiphertextBlob` se devuelven en formato codificado en base64.

Para obtener más información, consulte [los pares de claves de datos](#) en la Guía para desarrolladores del Servicio de administración de AWS claves.



- Para API obtener más información, consulte [GenerateDataKeyPairWithoutPlaintext](#) Referencia de AWS CLI comandos.

## generate-data-key-pair

En el siguiente ejemplo de código, se muestra cómo usar `generate-data-key-pair`.

### AWS CLI

Para generar un par de claves de datos RSA asimétricas de 2048 bits

El siguiente `generate-data-key-pair` ejemplo solicita un par de claves de datos RSA asimétricas de 2048 bits para su uso fuera de. AWS El comando devuelve una clave pública de texto sin formato y una clave privada de texto sin formato para su uso y eliminación inmediatos, así como una copia de la clave privada cifrada con la clave especificada. KMS Puede almacenar de forma segura la clave privada cifrada con los datos cifrados.

Para solicitar un par de claves de datos RSA asimétricas de 2048 bits, utilice el `key-pair-spec` parámetro con un valor de. `RSA_2048`

La KMS clave que especifique debe ser una clave de cifrado KMS simétrica, es decir, una KMS clave con un valor de. `KeySpec SYMMETRIC_DEFAULT`

NOTE: Los valores de la salida de este ejemplo están truncados para mostrarlos.

```
aws kms generate-data-key-pair \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --key-pair-spec RSA_2048
```

Salida:

```
{  
  "PrivateKeyCiphertextBlob": "AQIDAHi6LtupRpdK12aJTzkK6Fbh0tQkM1QJJH3PdtHvS/y  
+hAFFxmiD134doUDzMGmfCEtcAAAHaTCCB2UGCSqGSIB3DQEHbqCCB1...",  
  "PrivateKeyPlaintext": "MIIG/  
QIBADANBgkqhkiG9w0BAQEFAASCBUcwggbjAgEAAoIBgQDcDd4YzI  
+u9Kfv4t2UkTWhShBXkekS4cBVt07I0P42ZgMf+YvU5IgS4ut...",  
  "PublicKey":  
  "MIIBojANBgkqhkiG9w0BAQEFAAOCAY8AMIIBigKCAYEA3A3eGMyPrivSn7+Ld1JE1oUoQV5HpEuHAVbd0yND  
+NmYDH/mL10SIEuLrzdZ5hrMH4pk83r401...",
```

```
"KeyId": "arn:aws:kms:us-  
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",  
  "KeyPairSpec": "RSA_2048"  
}
```

Los valores `PublicKeyPrivateKeyPlaintext`, y `PrivateKeyCiphertextBlob` se devuelven en formato codificado en base64.

Para obtener más información, consulte [los pares de claves de datos](#) en la Guía para desarrolladores del Servicio de administración de AWS claves.

- Para API obtener más información, consulte [GenerateDataKeyPair](#) la Referencia de AWS CLI comandos.

## generate-data-key-without-plaintext

En el siguiente ejemplo de código, se muestra cómo usar `generate-data-key-without-plaintext`.

### AWS CLI

Generación de una clave de datos simétrica de 256 bits sin una clave de texto no cifrado

En el siguiente ejemplo de `generate-data-key-without-plaintext` se solicita una copia cifrada de una clave de datos simétrica de 256 bits para usarla fuera de AWS. Puede llamar AWS KMS para descifrar la clave de datos cuando esté listo para usarla.

Para solicitar una clave de datos de 256 bits, utilice el parámetro `key-spec` con un valor de `AES_256`. Para solicitar una clave de datos de 128 bits, utilice el parámetro `key-spec` con un valor de `AES_128`. Para el resto longitudes de clave de datos, utilice el parámetro `number-of-bytes`.

La KMS clave que especifique debe ser una KMS clave de cifrado simétrica, es decir, una KMS clave con un valor de especificación de clave de `_`. `SYMMETRIC DEFAULT`

```
aws kms generate-data-key-without-plaintext \  
  --key-id "arn:aws:kms:us-  
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab" \  
  --key-spec AES_256
```

Salida:

```
{
  "CiphertextBlob":
  "AQEDAHjRYf5WytIc0C857tFSnBaPn2F8DgfmThbJlGfR8P3WlwAAAH4wfAYJKoZIhvcNAQcGoG8wbQIBADBoBgkqhkiG9w0BAQsD
  "KeyId": "arn:aws:kms:us-
  east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
}
```

CiphertextBlob (clave de datos cifrada) se devuelve en formato codificado en base64.

Para obtener más información, consulte [Claves de datos](#) en la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte la Referencia [GenerateDataKeyWithoutPlaintext](#) de AWS CLI comandos.

## generate-data-key

En el siguiente ejemplo de código, se muestra cómo usar generate-data-key.

### AWS CLI

Ejemplo 1: Generación de una clave de datos simétrica de 256 bits

En el siguiente generate-data-key ejemplo, se solicita una clave de datos simétrica de 256 bits para utilizarla fuera de. AWS El comando devuelve una clave de datos en texto plano para su uso y eliminación inmediatos, y una copia de esa clave de datos cifrada con la clave especificada. KMS Puede almacenar de forma segura la clave de datos cifrada con los datos cifrados.

Para solicitar una clave de datos de 256 bits, utilice el parámetro key-spec con un valor de AES\_256. Para solicitar una clave de datos de 128 bits, utilice el parámetro key-spec con un valor de AES\_128. Para el resto longitudes de clave de datos, utilice el parámetro number-of-bytes.

La KMS clave que especifique debe ser una KMS clave de cifrado simétrica, es decir, una KMS clave con un valor de especificación de clave de `_`. SYMMETRIC DEFAULT

```
aws kms generate-data-key \
  --key-id alias/ExampleAlias \
  --key-spec AES_256
```

Salida:

```
{
  "Plaintext": "VdzKNHGzUAzJeRBVY+uUmofUGGiDzyB3+i9fVkh3piw=",
  "KeyId": "arn:aws:kms:us-east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "CiphertextBlob":
  "AQEDAHjRYf5WytIc0C857tFSnBaPn2F8DgfmThbJlGfR8P3WlwAAAH4wfAYJKoZIHvcNAQcGoG8wbQIBADBoBqkqhki+YdhV8MrkBQPeac0ReRVNDt9q1eAt+SHgIRF8P0H+7U="
}
```

Plaintext (clave de datos de texto sin formato) y CiphertextBlob (clave de datos cifrada) se devuelven en formato codificado en base64.

Para obtener más información, consulte Data keys < <https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html#data-keys> en la Guía para desarrolladores del servicio de administración de AWS claves.

Ejemplo 2: Generación de una clave de datos simétrica de 512 bits

En el siguiente ejemplo de `generate-data-key` se solicita una clave de datos simétrica de 512 bits para cifrado y descifrado. El comando devuelve una clave de datos en texto plano para su uso y eliminación inmediatos, y una copia de esa clave de datos cifrada con la clave especificada KMS. Puede almacenar de forma segura la clave de datos cifrada con los datos cifrados.

Para solicitar una longitud de clave distinta de 128 o 256 bits, utilice el parámetro `number-of-bytes`. Para solicitar una clave de datos de 512 bits, en el siguiente ejemplo se utiliza el parámetro `number-of-bytes` con un valor de 64 (bytes).

La KMS clave que especifique debe ser una KMS clave de cifrado simétrica, es decir, una KMS clave con un valor de especificación de clave de `_SYMMETRIC_DEFAULT`.

NOTE: Los valores de la salida de este ejemplo están truncados para mostrarlos.

```
aws kms generate-data-key \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --number-of-bytes 64
```

Salida:

```
{
  "CiphertextBlob": "AQIBAHi6LtupRpdK12aJTzkK6Fbh0tQkMlQJJH3PdtHvS/y+hAEnX/
QQNmMwDfg2koιNMEc8AAACaDCCAmQGCSqGSiB3DQEHBqCCA1UwggJRAgEAMIICSgYJKoZ...",
  "Plaintext": "ty8Lr0Bk60F07M2Bwt6qbFdNB
+G00ZLtf5MSEb4a13R2UKWG0p06njAwy2n72VRm2m7z/
Pm9Wpbvttz6a4lSo9hgPvKhZ5y6RTm40ovEXiVfBveyX3DQxDzRSwbKDPk/...",
  "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
}
```

`Plaintext` (clave de datos de texto sin formato) y `CiphertextBlob` (clave de datos cifrada) se devuelven en formato codificado en base64.

Para obtener más información, consulte [Data keys < https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html#data-keys](https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html#data-keys) en la Guía para desarrolladores del servicio de administración de AWS claves.

- Para API obtener más información, consulte [GenerateDataKey](#) la Referencia de AWS CLI comandos.

## generate-random

En el siguiente ejemplo de código, se muestra cómo usar `generate-random`.

### AWS CLI

Ejemplo 1: Para generar una cadena de bytes aleatorios de 256 bits (Linux o macOS)

En el siguiente ejemplo de `generate-random` se genera una cadena de bytes aleatoria de 256 bits (32 bytes) codificada en base64. El ejemplo decodifica la cadena de bytes y la guarda en el archivo aleatorio.

Al ejecutar este comando, debe usar el parámetro `number-of-bytes` para especificar la longitud del valor aleatorio en bytes.

No se especifica ninguna KMS clave al ejecutar este comando. La cadena de bytes aleatorios no está relacionada con ninguna KMS clave.

De forma predeterminada, AWS KMS genera el número aleatorio. Sin embargo, si especificas un almacén de claves personalizado < <https://docs.aws.amazon.com/kms/latest/developerguide/>

custom-key-store-overview .html>, la cadena de bytes aleatoria se genera en el clúster de Cloud asociado al AWS almacén de claves personalizado. HSM

Este ejemplo utiliza los parámetros y valores siguientes:

Utiliza el `--number-of-bytes` parámetro necesario con un valor de 32 para solicitar una cadena de 32 bytes (256 bits). Utiliza el `--output` parámetro con un valor de `text` para indicar que devuelva la AWS CLI salida como texto, en lugar de JSON. Utiliza el `--query` parameter para extraer el valor de la `Plaintext` propiedad de la respuesta. Canaliza (`|`) la salida del comando a la `base64` utilidad, que decodifica la salida extraída. Utiliza el operador de redirección (`>`) para guardar la cadena de bytes decodificada en `ExampleRandom` File e.it usa el operador de redirección (`>`) para guardar el texto cifrado binario en un archivo.

```
aws kms generate-random \  
  --number-of-bytes 32 \  
  --output text \  
  --query Plaintext | base64 --decode > ExampleRandom
```

Este comando no genera ninguna salida.

Para obtener más información, consulte la referencia del [GenerateRandom](#) servicio de administración de AWS claves. API

Ejemplo 2: Generación de un número aleatorio de 256 bits (símbolo del sistema de Windows)

En el siguiente ejemplo se utiliza el comando `generate-random` para generar una cadena de bytes aleatoria codificada en base64 de 256 bits (32 bytes). El ejemplo decodifica la cadena de bytes y la guarda en el archivo aleatorio. Este ejemplo es igual al anterior, excepto que usa la utilidad `certutil` de Windows para decodificar en base64 la cadena de bytes aleatoria antes de guardarla en un archivo.

En primer lugar, genere una cadena de bytes aleatoria codificada en base64 y guárdela en un archivo temporal, `ExampleRandom.base64`.

```
aws kms generate-random \  
  --number-of-bytes 32 \  
  --output text \  
  --query Plaintext > ExampleRandom.base64
```

Puesto que el resultado del comando `generate-random` se guarda en un archivo, este ejemplo no produce ningún resultado.

Ahora utilice el comando `certutil -decode` para decodificar la cadena de bytes codificada en base64 del archivo `ExampleRandom.base64`. A continuación, guarda la cadena de bytes decodificada en el archivo `ExampleRandom`.

```
certutil -decode ExampleRandom.base64 ExampleRandom
```

Salida:

```
Input Length = 18
Output Length = 12
CertUtil: -decode command completed successfully.
```

Para obtener más información, consulte [GenerateRandom](#) la API referencia del servicio de administración de AWS claves.

- Para API obtener más información, consulte [GenerateRandom](#) la Referencia de AWS CLI comandos.

## get-key-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-key-policy`.

### AWS CLI

Para copiar una política de claves de una KMS clave a otra KMS

En el siguiente `get-key-policy` ejemplo, se obtiene la política de claves de una KMS clave y se guarda en un archivo de texto. A continuación, reemplaza la política de una KMS clave diferente utilizando el archivo de texto como entrada de política.

Como el `--policy` parámetro de `put-key-policy` requiere una cadena, debe usar la `--output text` opción para devolver el resultado como una cadena de texto en lugar de hacerlo JSON.

```
aws kms get-key-policy \
  --policy-name default \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --query Policy \
  --output text > policy.txt
```

```
aws kms put-key-policy \  
  --policy-name default \  
  --key-id 0987dcba-09fe-87dc-65ba-ab0987654321 \  
  --policy file://policy.txt
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [PutKeyPolicy](#) la AWS KMS API Referencia.

- Para API obtener más información, consulte [GetKeyPolicy](#) la Referencia de AWS CLI comandos.

## get-key-rotation-status

En el siguiente ejemplo de código, se muestra cómo usar `get-key-rotation-status`.

### AWS CLI

Para recuperar el estado de rotación de una KMS llave.

El siguiente `get-key-rotation-status` ejemplo devuelve información sobre el estado de rotación de la KMS clave especificada, que incluye si la rotación automática está habilitada, el período de rotación y la próxima fecha de rotación programada. Puede usar este comando en KMS las claves administradas por el cliente y en las KMS claves AWS administradas. Sin embargo, todas KMS las claves AWS gestionadas se rotan automáticamente cada año.

```
aws kms get-key-rotation-status \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Salida:

```
{  
  "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",  
  "KeyRotationEnabled": true,  
  "NextRotationDate": "2024-02-14T18:14:33.587000+00:00",  
  "RotationPeriodInDays": 365  
}
```

Para obtener más información, consulte [Rotación de claves](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.



- Para API obtener más información, consulte [GetKeyRotationStatus](#) la Referencia de AWS CLI comandos.

## get-parameters-for-import

En el siguiente ejemplo de código, se muestra cómo usar `get-parameters-for-import`.

### AWS CLI

Para obtener los elementos necesarios para importar material clave a una KMS clave

En el siguiente `get-parameters-for-import` ejemplo, se obtienen la clave pública y el token de importación que necesita para importar el material clave a una KMS clave. Cuando utilice el `import-key-material` comando, asegúrese de utilizar el token de importación y el material clave cifrados con la clave pública que se devolvieron en el mismo `get-parameters-for-import` comando. Además, el algoritmo de empaquetado que especifique en este comando debe ser el que utilice para cifrar el material de la clave con la clave pública.

Para especificar la KMS clave, utilice el `key-id` parámetro. En este ejemplo se utiliza un identificador de clave, pero puede utilizar un identificador de clave o una clave ARN en este comando.

```
aws kms get-parameters-for-import \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --wrapping-algorithm RSAES_OAEP_SHA_256 \  
  --wrapping-key-spec RSA_2048
```

Salida:

```
{  
  "KeyId": "arn:aws:kms:us-  
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",  
  "PublicKey": "<public key base64 encoded data>",  
  "ImportToken": "<import token base64 encoded data>",  
  "ParametersValidTo": 1593893322.32  
}
```

Para obtener más información, consulte [Descargar la clave pública y el token de importación](#) en la Guía para desarrolladores del Servicio de administración de AWS claves.

- Para API obtener más información, consulte [GetParametersForImport](#) la Referencia de AWS CLI comandos.

## get-public-key

En el siguiente ejemplo de código, se muestra cómo usar `get-public-key`.

### AWS CLI

Ejemplo 1: Para descargar la clave pública de una clave asimétrica KMS

En el siguiente `get-public-key` ejemplo, se descarga la clave pública de una clave asimétrica KMS.

Además de devolver la clave pública, el resultado incluye la información necesaria para utilizarla de forma segura fuera de ella AWS KMS, incluidos el uso de la clave y los algoritmos de cifrado compatibles.

```
aws kms get-public-key \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Salida:

```
{
  "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "PublicKey": "jANBgkqhkiG9w0BAQEFAAOCAg8AMIICCgKCAgEA15epvg1/
QtJhxSi2g9SDEVg8QV/...",
  "CustomerMasterKeySpec": "RSA_4096",
  "KeyUsage": "ENCRYPT_DECRYPT",
  "EncryptionAlgorithms": [
    "RSAES_OAEP_SHA_1",
    "RSAES_OAEP_SHA_256"
  ]
}
```

Para obtener más información sobre el uso de KMS claves asimétricas en AWS KMS, consulte [Uso de claves simétricas y asimétricas en la referencia del servicio de administración de AWS claves](#). API

Ejemplo 2: Para convertir una clave pública a DER formato (Linux y macOS)

El siguiente `get-public-key` ejemplo descarga la clave pública de una KMS clave asimétrica y la guarda en un DER archivo.

Al utilizar el `get-public-key` comando en AWS CLI, devuelve una clave pública X.509 DER codificada en Base64. En este ejemplo, se obtiene el valor de la propiedad en forma de texto. `PublicKey` Lo decodifica en Base64 `PublicKey` y lo guarda en el archivo. `public_key.der` El `output` parámetro devuelve la salida como texto, en lugar de. JSON El `--query` parámetro obtiene solo la `PublicKey` propiedad, no las propiedades fuera de las cuales necesita usar la clave pública de forma segura AWS KMS.

Antes de ejecutar este comando, sustituya el identificador de clave de ejemplo por un identificador de clave válido de su AWS cuenta.

```
aws kms get-public-key \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --output text \  
  --query PublicKey | base64 --decode > public_key.der
```

Este comando no genera ninguna salida.

Para obtener más información sobre el uso de KMS claves asimétricas en AWS KMS, consulte [Uso de claves simétricas y asimétricas en la referencia del servicio de administración de AWS claves](#). API

- Para API obtener más información, consulte la Referencia de [GetPublicKey](#) comandos AWS CLI .

## **import-key-material**

En el siguiente ejemplo de código, se muestra cómo usar `import-key-material`.

### AWS CLI

Para importar material clave a una KMS clave

En el siguiente `import-key-material` ejemplo, se carga el material clave en una KMS clave que se creó sin material clave. El estado clave de la KMS clave debe ser `PendingImport`.

Este comando utiliza material clave que usted cifró con la clave pública que devolvió el `get-parameters-for-import` comando. También usa el token de importación del mismo `get-parameters-for-import` comando.

El `expiration-model` parámetro indica que el material clave caduca automáticamente en la fecha y hora especificadas por el `valid-to` parámetro. Cuando el material clave caduca, lo AWS KMS elimina, el estado de la KMS clave cambia a `Pending import` y la KMS clave queda inutilizable. Para restaurar la KMS clave, debe volver a importar el mismo material clave. Para utilizar un material clave diferente, debe crear una KMS clave nueva.

Antes de ejecutar este comando, sustituya el identificador de clave del ejemplo por un identificador de clave válido o una clave ARN de su AWS cuenta.

```
aws kms import-key-material \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --encrypted-key-material fileb://EncryptedKeyMaterial.bin \
  --import-token fileb://ImportToken.bin \
  --expiration-model KEY_MATERIAL_EXPIRES \
  --valid-to 2021-09-21T19:00:00Z
```

Este comando no genera ninguna salida.

Para obtener más información sobre la importación de material clave, consulte [Importación de material clave](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.

- Para API obtener más información, consulte [ImportKeyMaterial](#) la Referencia de AWS CLI comandos.

## list-aliases

En el siguiente ejemplo de código, se muestra cómo usar `list-aliases`.

### AWS CLI

Ejemplo 1: Para enumerar todos los alias de una AWS cuenta y una región

En el siguiente ejemplo, se utiliza el `list-aliases` comando para enumerar todos los alias de la región predeterminada de la AWS cuenta. El resultado incluye los alias asociados a las claves administradas y a KMS las claves AWS administradas por el cliente. KMS

```
aws kms list-aliases
```

Salida:

```
{
  "Aliases": [
    {
      "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/testKey",
      "AliasName": "alias/testKey",
      "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    {
      "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/FinanceDept",
      "AliasName": "alias/FinanceDept",
      "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321"
    },
    {
      "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/aws/dynamodb",
      "AliasName": "alias/aws/dynamodb",
      "TargetKeyId": "1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
    },
    {
      "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/aws/ebs",
      "AliasName": "alias/aws/ebs",
      "TargetKeyId": "0987ab65-43cd-21ef-09ab-87654321cdef"
    },
    ...
  ]
}
```

## Ejemplo 2: Para enumerar todos los alias de una clave concreta KMS

En el siguiente ejemplo, se utiliza el `list-aliases` comando y su `key-id` parámetro para enumerar todos los alias que están asociados a una clave concreta KMS.

Cada alias está asociado a una sola KMS clave, pero una KMS clave puede tener varios alias. Este comando es muy útil porque la AWS KMS consola solo muestra un alias para cada KMS clave. Para buscar todos los alias de una KMS clave, debe usar el `list-aliases` comando.

En este ejemplo se utiliza el identificador de clave de la KMS clave del `--key-id` parámetro, pero puede utilizar un identificador de clave, una claveARN, un nombre de alias o un alias ARN en este comando.

```
aws kms list-aliases --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Salida:

```
{
  "Aliases": [
    {
      "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
      "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/oregon-test-key",
      "AliasName": "alias/oregon-test-key"
    },
    {
      "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
      "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/project121-test",
      "AliasName": "alias/project121-test"
    }
  ]
}
```

Para obtener más información, consulte [Uso de alias](#) en la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte [ListAliases](#) la Referencia de AWS CLI comandos.

## list-grants

En el siguiente ejemplo de código, se muestra cómo usar `list-grants`.

### AWS CLI

Para ver las concesiones en una AWS KMS clave

En el siguiente `list-grants` ejemplo, se muestran todas las concesiones de la KMS clave AWS gestionada especificada para Amazon DynamoDB en su cuenta. Esta concesión permite a DynamoDB utilizar KMS la clave en su nombre para cifrar una tabla de DynamoDB antes de escribirla en el disco. Puede usar un comando como este para ver las concesiones de las claves administradas y KMS las claves AWS administradas por el cliente en la cuenta y la KMS región.

AWS

Este comando usa el `key-id` parámetro con un identificador de clave para identificar la KMS clave. Puede utilizar un identificador de clave o una clave ARN para identificar la KMS clave. Para obtener el identificador de clave o ARN la clave de una KMS clave AWS gestionada, utilice el `list-aliases` comando `list-keys` o.

```
aws kms list-grants \
```

```
--key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

El resultado muestra que la concesión otorga a Amazon DynamoDB permiso para usar KMS la clave en operaciones criptográficas y le da permiso para ver detalles sobre KMS la clave DescribeKey () y para retirar grants (). RetireGrant La restricción EncryptionContextSubset limita estos permisos a las solicitudes que incluyen los pares de contexto de cifrado especificados. Como resultado, los permisos de la concesión solo son efectivos en la cuenta y la tabla de DynamoDB especificadas.

```
{
  "Grants": [
    {
      "Constraints": {
        "EncryptionContextSubset": {
          "aws:dynamodb:subscriberId": "123456789012",
          "aws:dynamodb:tableName": "Services"
        }
      },
      "IssuingAccount": "arn:aws:iam::123456789012:root",
      "Name": "8276b9a6-6cf0-46f1-b2f0-7993a7f8c89a",
      "Operations": [
        "Decrypt",
        "Encrypt",
        "GenerateDataKey",
        "ReEncryptFrom",
        "ReEncryptTo",
        "RetireGrant",
        "DescribeKey"
      ],
      "GrantId":
        "1667b97d27cf748cf05b487217dd4179526c949d14fb3903858e25193253fe59",
      "KeyId": "arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
      "RetiringPrincipal": "dynamodb.us-west-2.amazonaws.com",
      "GranteePrincipal": "dynamodb.us-west-2.amazonaws.com",
      "CreationDate": "2021-05-13T18:32:45.144000+00:00"
    }
  ]
}
```

Para obtener más información, consulte las [subvenciones AWS KMS en](#) la Guía AWS para desarrolladores de Key Management Service.

- Para API obtener más información, consulte [ListGrants](#) la Referencia de AWS CLI comandos.

## list-key-policies

En el siguiente ejemplo de código, se muestra cómo usar `list-key-policies`.

### AWS CLI

Para obtener los nombres de las políticas clave de una KMS clave

En el siguiente ejemplo de `list-key-policies` se obtienen los nombres de las políticas de clave de una clave administrada por el cliente en la cuenta y la región del ejemplo. Puede usar este comando para buscar los nombres de las políticas clave para las claves AWS administradas y las claves administradas por el cliente.

Este comando no resulta útil, ya que el único nombre de la política de claves válido es `default`.

Para especificar la KMS clave, utilice el `key-id` parámetro. En este ejemplo se utiliza un valor de identificador de clave, pero puede utilizar un identificador de clave o una clave ARN en este comando.

```
aws kms list-key-policies \  
--key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Salida:

```
{  
  "PolicyNames": [  
    "default"  
  ]  
}
```

Para obtener más información sobre las políticas AWS KMS clave, consulte [Uso de políticas clave AWS KMS en](#) la Guía AWS para desarrolladores del Servicio de administración de claves.

- Para API obtener más información, consulte [ListKeyPolicies](#) la Referencia de AWS CLI comandos.

## list-key-rotations

En el siguiente ejemplo de código, se muestra cómo usar `list-key-rotations`.



## AWS CLI

Para recuperar información sobre todas las rotaciones de materiales clave completadas

En el siguiente `list-key-rotations` ejemplo, se muestra información sobre todas las rotaciones de material clave completadas para la clave especificada KMS.

```
aws kms list-key-rotations \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Salida:

```
{
  "Rotations": [
    {
      "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
      "RotationDate": "2024-03-02T10:11:36.564000+00:00",
      "RotationType": "AUTOMATIC"
    },
    {
      "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
      "RotationDate": "2024-04-05T15:14:47.757000+00:00",
      "RotationType": "ON_DEMAND"
    }
  ],
  "Truncated": false
}
```

Para obtener más información, consulte [Rotación de claves](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.

- Para API obtener más información, consulte [ListKeyRotations](#) la Referencia de AWS CLI comandos.

## list-keys

En el siguiente ejemplo de código, se muestra cómo usar `list-keys`.

## AWS CLI

Para obtener las KMS claves de una cuenta y una región

En el siguiente `list-keys` ejemplo, se obtienen las KMS claves de una cuenta y una región. Este comando devuelve tanto las claves AWS administradas como las claves administradas por el cliente.

```
aws kms list-keys
```

Salida:

```
{
  "Keys": [
    {
      "KeyArn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
      "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    {
      "KeyArn": "arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321",
      "KeyId": "0987dcba-09fe-87dc-65ba-ab0987654321"
    },
    {
      "KeyArn": "arn:aws:kms:us-
east-2:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d",
      "KeyId": "1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
    }
  ]
}
```

Para obtener más información, consulte [Visualización de claves](#) en la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte [ListKeys](#) la Referencia de AWS CLI comandos.

## **list-resource-tags**

En el siguiente ejemplo de código, se muestra cómo usar `list-resource-tags`.

### AWS CLI

Para obtener las etiquetas de una KMS clave

En el siguiente `list-resource-tags` ejemplo, se obtienen las etiquetas de una KMS clave. Para añadir o reemplazar etiquetas de recursos en KMS las claves, utilice el `tag-resource` comando. El resultado muestra que esta KMS clave tiene dos etiquetas de recursos, cada una de las cuales tiene una clave y un valor.

Para especificar la KMS clave, utilice el `key-id` parámetro. En este ejemplo se utiliza un valor de identificador de clave, pero puede utilizar un identificador de clave o una clave ARN en este comando.

```
aws kms list-resource-tags \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Salida:

```
{  
  "Tags": [  
    {  
      "TagKey": "Dept",  
      "TagValue": "IT"  
    },  
    {  
      "TagKey": "Purpose",  
      "TagValue": "Test"  
    }  
  ],  
  "Truncated": false  
}
```

Para obtener más información sobre el uso de etiquetas en AWS KMS, consulte [Etiquetar claves](#) en la Guía para desarrolladores del Servicio de administración de AWS claves.

- Para API obtener más información, consulte [ListResourceTags](#) la Referencia de AWS CLI comandos.

## **list-retirable-grants**

En el siguiente ejemplo de código, se muestra cómo usar `list-retirable-grants`.

AWS CLI

Para ver las subvenciones que un director puede retirar

En el siguiente `list-retirable-grants` ejemplo, se muestran todas las concesiones que el `ExampleAdmin` usuario puede retirar en relación con las KMS claves de una AWS cuenta y una región. Puede usar un comando como este para ver las concesiones que cualquier titular de cuenta puede retirar en relación con KMS las claves de la AWS cuenta y la región.

El valor del `retiring-principal` parámetro obligatorio debe ser el nombre del recurso de Amazon (ARN) de una cuenta, usuario o rol.

No puede especificar un servicio con el valor de `retiring-principal` en este comando, aunque un servicio pueda ser el principal que se retira. Para encontrar las concesiones en las que un servicio concreto es el principal que se retira, utilice el `list-grants` comando.

El resultado muestra que el `ExampleAdmin` usuario tiene permiso para retirar las subvenciones en dos KMS claves diferentes de la cuenta y la región. Además del principal que se retira, la cuenta tiene permiso para retirar cualquier subvención de la cuenta.

```
aws kms list-retirable-grants \  
--retiring-principal arn:aws:iam::111122223333:user/ExampleAdmin
```

Salida:

```
{  
  "Grants": [  
    {  
      "KeyId": "arn:aws:kms:us-  
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",  
      "GrantId":  
"156b69c63cb154aa21f59929ff19760717be8d9d82b99df53e18b94a15a5e88e",  
      "Name": "",  
      "CreationDate": 2021-01-14T20:17:36.419000+00:00,  
      "GranteePrincipal": "arn:aws:iam::111122223333:user/ExampleUser",  
      "RetiringPrincipal": "arn:aws:iam::111122223333:user/ExampleAdmin",  
      "IssuingAccount": "arn:aws:iam::111122223333:root",  
      "Operations": [  
        "Encrypt"  
      ],  
      "Constraints": {  
        "EncryptionContextSubset": {  
          "Department": "IT"  
        }  
      }  
    },  
  ],  
}
```

```

    {
      "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321",
      "GrantId":
"8c94d1f12f5e69f440bae30eaec9570bb1fb7358824f9ddfa1aa5a0dab1a59b2",
      "Name": "",
      "CreationDate": "2021-02-02T19:49:49.638000+00:00",
      "GranteePrincipal": "arn:aws:iam::111122223333:role/ExampleRole",
      "RetiringPrincipal": "arn:aws:iam::111122223333:user/ExampleAdmin",
      "IssuingAccount": "arn:aws:iam::111122223333:root",
      "Operations": [
        "Decrypt"
      ],
      "Constraints": {
        "EncryptionContextSubset": {
          "Department": "IT"
        }
      }
    }
  ],
  "Truncated": false
}

```

Para obtener más información, consulte [las subvenciones AWS KMS en](#) la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte [ListRetirableGrants](#) la Referencia de AWS CLI comandos.

## put-key-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-key-policy`.

### AWS CLI

Para cambiar la política de claves de una KMS clave

En el siguiente ejemplo de `put-key-policy` se cambia la política de claves para una clave administrada por el cliente.

Para empezar, cree una política de claves y guárdela en un JSON archivo local. En este ejemplo, el archivo es `key_policy.json`. También puede especificar la política de claves como un valor de cadena del parámetro `policy`.

La primera declaración de esta política clave da permiso a la AWS cuenta para usar IAM políticas que controlen el acceso a la KMS clave. La segunda sentencia otorga al `test-user` usuario permiso para ejecutar los `list-keys` comandos `describe-key` y de la KMS tecla.

Contenidos de `key_policy.json`:

```
{
  "Version" : "2012-10-17",
  "Id" : "key-default-1",
  "Statement" : [
    {
      "Sid" : "Enable IAM User Permissions",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "arn:aws:iam::111122223333:root"
      },
      "Action" : "kms:*",
      "Resource" : "*"
    },
    {
      "Sid" : "Allow Use of Key",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "arn:aws:iam::111122223333:user/test-user"
      },
      "Action" : [
        "kms:DescribeKey",
        "kms:ListKeys"
      ],
      "Resource" : "*"
    }
  ]
}
```

Para identificar la KMS clave, en este ejemplo se utiliza el identificador de clave, pero también se puede utilizar una claveARN. El comando utiliza el parámetro `policy` para especificar la políticas de claves. Para indicar que la política se encuentra en un archivo, utiliza el prefijo `file://` necesario. Este prefijo es necesario para identificar los archivos de todos los sistemas operativos compatibles. Por último, el comando usa el parámetro `policy-name` con un valor de `default`. Si no se especifica ningún nombre de política, el valor predeterminado es `default`. El único valor válido es `default`.

```
aws kms put-key-policy \  
  --policy-name default \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --policy file://key_policy.json
```

Este comando no produce ninguna salida. Para comprobar que se aplicó el comando, utilice el comando `get-key-policy`. El siguiente comando de ejemplo obtiene la política clave para la misma KMS clave. El parámetro `output` con un valor de `text` devuelve un formato de texto que es fácil de leer.

```
aws kms get-key-policy \  
  --policy-name default \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --output text
```

Salida:

```
{  
  "Version" : "2012-10-17",  
  "Id" : "key-default-1",  
  "Statement" : [  
    {  
      "Sid" : "Enable IAM User Permissions",  
      "Effect" : "Allow",  
      "Principal" : {  
        "AWS" : "arn:aws:iam::111122223333:root"  
      },  
      "Action" : "kms:*",  
      "Resource" : "*"   
    },  
    {  
      "Sid" : "Allow Use of Key",  
      "Effect" : "Allow",  
      "Principal" : {  
        "AWS" : "arn:aws:iam::111122223333:user/test-user"  
      },  
      "Action" : [ "kms:Describe", "kms:List" ],  
      "Resource" : "*"   
    }   
  ]   
}
```

Para obtener más información, consulte [Cambio de una política de claves](#) en la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte [PutKeyPolicy](#) la Referencia de AWS CLI comandos.

## re-encrypt

En el siguiente ejemplo de código, se muestra cómo usar `re-encrypt`.

### AWS CLI

Ejemplo 1: volver a cifrar un mensaje cifrado con una KMS clave simétrica diferente (Linux y macOS).

El siguiente ejemplo de `re-encrypt` comando muestra la forma recomendada de volver a cifrar los datos con. AWS CLI

Proporcione el texto cifrado de un archivo. En el valor del `--ciphertext-blob` parámetro, utilice el `fileb://` prefijo, que indica que debe leer los datos de un archivo CLI binario. Si el archivo no se encuentra en el directorio actual, escriba la ruta completa al archivo. Para obtener más información sobre la lectura AWS CLI de los valores de los AWS CLI parámetros de un archivo, consulte Carga de parámetros desde un archivo < <https://docs.aws.amazon.com/cli/latest/userguide/cli-usage-parameters-file.html> > en la Guía del usuario de la interfaz de línea de AWS comandos y Prácticas recomendadas para los parámetros de archivos locales < <https://aws.amazon.com/blogs/developer/best-practices-for-local-file-parameters/> > en el blog de herramientas de línea de AWS comandos. Especifique la KMS clave fuente, que descifra el texto cifrado. El `--source-key-id` parámetro no es necesario cuando se descifra con simétrico KMS AWS KMS puede obtener la KMS clave que se utilizó para cifrar los datos de los metadatos del blob de texto cifrado. Sin embargo, siempre se recomienda especificar la KMS clave que se está utilizando. Esta práctica garantiza que utiliza la KMS clave deseada y evita que descifre inadvertidamente un texto cifrado con una clave en la que no confíe. Especifique la KMS clave de destino, que volverá a cifrar los datos. El parámetro siempre KMS es obligatorio. `--destination-key-id` En este ejemplo se utiliza una claveARN, pero se puede utilizar cualquier identificador de clave válido. Solicite la salida en texto simple como un valor de texto. El parámetro le indica a la que debe obtener únicamente el valor del campo de la salida. `--query CLI Plaintext` El parámetro `--output` devuelve la salida como texto. Decodifique en base64 el texto sin formato y guárdelo en un archivo. El siguiente ejemplo canaliza (`()`) el valor



del parámetro `Plaintext` a la utilidad `Base64`, que lo decodifica. A continuación, redirige (`>`) la salida descodificada al archivo `ExamplePlaintext`.

Antes de ejecutar este comando, sustituya la clave de ejemplo por identificadores de clave IDs válidos de su cuenta. AWS

```
aws kms re-encrypt \  
  --ciphertext-blob fileb://ExampleEncryptedFile \  
  --source-key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --destination-key-id 0987dcba-09fe-87dc-65ba-ab0987654321 \  
  --query CiphertextBlob \  
  --output text | base64 --decode > ExampleReEncryptedFile
```

Este comando no genera ninguna salida. La salida del comando `re-encrypt` se descodifica en `base64` y se guarda en un archivo.

Para obtener más información, consulte `ReEncrypt` < [https://docs.aws.amazon.com/kms/APIReference/latest/API\\_ReEncrypt.html](https://docs.aws.amazon.com/kms/APIReference/latest/API_ReEncrypt.html) en la referencia del servicio de administración de AWS claves. API

Ejemplo 2: volver a cifrar un mensaje cifrado con una KMS clave simétrica diferente (línea de comandos de Windows).

El siguiente comando `re-encrypt` de ejemplo es el mismo que el anterior, excepto que usa la utilidad `certutil` para descodificar en `Base64` los datos de texto sin formato. Como se muestra en los siguientes ejemplos, este procedimiento requiere dos comandos.

Antes de ejecutar este comando, sustituya el identificador de clave de ejemplo por un identificador de clave válido de su cuenta. AWS

```
aws kms re-encrypt ^  
  --ciphertext-blob fileb://ExampleEncryptedFile ^  
  --source-key-id 1234abcd-12ab-34cd-56ef-1234567890ab ^  
  --destination-key-id 0987dcba-09fe-87dc-65ba-ab0987654321 ^  
  --query CiphertextBlob ^  
  --output text > ExampleReEncryptedFile.base64
```

A continuación, use la utilidad `certutil`

```
certutil -decode ExamplePlaintextFile.base64 ExamplePlaintextFile
```

**Salida:**

```
Input Length = 18
Output Length = 12
CertUtil: -decode command completed successfully.
```

Para obtener más información, consulte [ReEncrypt](https://docs.aws.amazon.com/kms/APIReference/latest/API_ReEncrypt.html) < [https://docs.aws.amazon.com/kms/APIReference/latest/API\\_ReEncrypt.html](https://docs.aws.amazon.com/kms/APIReference/latest/API_ReEncrypt.html) en la referencia del servicio de administración de AWS claves. API

- Para API obtener más información, consulte la Referencia de [ReEncrypt](#) comandos AWS CLI .

**retire-grant**

En el siguiente ejemplo de código, se muestra cómo usar `retire-grant`.

## AWS CLI

Retirada de una concesión en una clave maestra de cliente

En el siguiente `retire-grant` ejemplo, se elimina una concesión de una KMS clave.

El siguiente comando de ejemplo especifica los parámetros `grant-id` y `key-id`. El valor del `key-id` parámetro debe ser la clave ARN de la KMS clave.

```
aws kms retire-grant \
  --grant-id 1234a2345b8a4e350500d432bccf8ecd6506710e1391880c4f7f7140160c9af3 \
  --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
```

Este comando no genera ninguna salida. Utilice el comando `list-grants` para confirmar que la concesión se ha retirado.

Para obtener más información, consulte [Retiro y revocación de concesiones](#) en la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte [RetireGrant](#) la Referencia de AWS CLI comandos.

**revoke-grant**

En el siguiente ejemplo de código, se muestra cómo usar `revoke-grant`.

## AWS CLI

Revocación de una concesión en una clave maestra de cliente

En el siguiente `revoke-grant` ejemplo, se elimina una concesión de una KMS clave. El siguiente comando de ejemplo especifica los parámetros `grant-id` y `key-id`. El valor del `key-id` parámetro puede ser el identificador de clave o la clave ARN de la KMS clave.

```
aws kms revoke-grant \  
  --grant-id 1234a2345b8a4e350500d432bccf8ecd6506710e1391880c4f7f7140160c9af3 \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Este comando no genera ninguna salida. Utilice el comando `list-grants` para confirmar que la concesión se ha revocado.

Para obtener más información, consulte [Retiro y revocación de concesiones](#) en la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte [RevokeGrant](#) la Referencia de AWS CLI comandos.

## **rotate-key-on-demand**

En el siguiente ejemplo de código, se muestra cómo usar `rotate-key-on-demand`.

## AWS CLI

Para realizar la rotación de una KMS tecla bajo demanda

El siguiente `rotate-key-on-demand` ejemplo inicia inmediatamente la rotación del material clave para la KMS clave especificada.

```
aws kms rotate-key-on-demand \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Salida:

```
{  
  "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"  
}
```

Para obtener más información, consulte [Cómo realizar la rotación de claves bajo demanda](#) en la Guía para desarrolladores del Servicio de administración de AWS claves.

- Para API obtener más información, consulte [RotateKeyOnDemand](#) la Referencia de AWS CLI comandos.

## schedule-key-deletion

En el siguiente ejemplo de código, se muestra cómo usar `schedule-key-deletion`.

### AWS CLI

Para programar la eliminación de una KMS clave gestionada por el cliente.

En el siguiente `schedule-key-deletion` ejemplo, se programa la eliminación de la KMS clave gestionada por el cliente especificada en 15 días.

El `--key-id` parámetro identifica la KMS clave. En este ejemplo se utiliza un ARN valor clave, pero se puede utilizar el identificador de clave o el ARN de la KMS clave. El `--pending-window-in-days` parámetro especifica la duración del período de espera de 7 a 30 días. De forma predeterminada, el periodo de espera es de 30 días. En este ejemplo se especifica un valor de 15, que indica que hay que AWS eliminar permanentemente la KMS clave 15 días después de que se complete el comando.

```
aws kms schedule-key-deletion \
  --key-id arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab \
  --pending-window-in-days 15
```

La respuesta incluye la claveARN, el estado de la clave, el período de espera (`PendingWindowInDays`) y la fecha de eliminación en tiempo de Unix. Para ver la fecha de eliminación en hora local, utilice la AWS KMS consola. KMS las claves en el estado `PendingDeletion` clave no se pueden usar en operaciones criptográficas.

```
{
  "KeyId": "arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "DeletionDate": "2022-06-18T23:43:51.272000+00:00",
  "KeyState": "PendingDeletion",
  "PendingWindowInDays": 15
```

```
}
```

Para obtener más información, consulte [Eliminación de claves](#) en la Guía para desarrolladores de AWS Key Management Service.

- Para API obtener más información, consulte [ScheduleKeyDeletion](#) la Referencia de AWS CLI comandos.

## sign

En el siguiente ejemplo de código, se muestra cómo usar `sign`.

### AWS CLI

Ejemplo 1: Para generar una firma digital para un mensaje

El siguiente `sign` ejemplo genera una firma criptográfica para un mensaje corto. El resultado del comando incluye un `Signature` campo codificado en base 64 que se puede comprobar mediante el comando `verify`

Debe especificar el mensaje que desee firmar y un algoritmo de firma compatible con la clave asimétrica KMS. Para obtener los algoritmos de firma de la KMS clave, utilice el `describe-key` comando.

En la AWS CLI versión 2.0, el valor del `message` parámetro debe estar codificado en Base64. O bien, puede guardar el mensaje en un archivo y usar el `fileb://` prefijo, que indica que deben leer los AWS CLI datos binarios del archivo.

Antes de ejecutar este comando, sustituya el identificador de clave de ejemplo por un identificador de clave válido de su AWS cuenta. El identificador de clave debe representar una KMS clave asimétrica con un uso de clave de `SIGN_VERIFY`.

```
msg=(echo 'Hello World' | base64)

aws kms sign \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --message fileb://UnsignedMessage \
  --message-type RAW \
  --signing-algorithm RSASSA_PKCS1_V1_5_SHA_256
```

Salida:

```
{
  "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "Signature": "ABCDEFhpyVYyTxbafE74ccSvEJLJr3zuoV1Hfymz4qv+/
fxmxNLA7SE1SiF8lHw80fKZZ3bJ...",
  "SigningAlgorithm": "RSASSA_PKCS1_V1_5_SHA_256"
}
```

Para obtener más información sobre el uso de KMS claves asimétricas en AWS KMS, consulte [Claves asimétricas AWS KMS en la Guía para desarrolladores del Servicio de administración de AWS claves](#).

Ejemplo 2: Para guardar una firma digital en un archivo (Linux y) macOS

El siguiente `sign` ejemplo genera una firma criptográfica para un mensaje corto almacenado en un archivo local. El comando también obtiene la `Signature` propiedad de la respuesta, la decodifica en Base64 y la guarda en el archivo. `ExampleSignature` Puede usar el archivo de firma en un `verify` comando que verifique la firma.

El `sign` comando requiere un mensaje codificado en Base64 y un algoritmo de firma compatible con la clave asimétrica. KMS Para obtener los algoritmos de firma compatibles con la KMS clave, utilice el comando. `describe-key`

Antes de ejecutar este comando, sustituya el identificador de clave del ejemplo por un identificador de clave válido de su AWS cuenta. El identificador de clave debe representar una KMS clave asimétrica con un uso de clave de `SIGN_VERIFY`.

```
echo 'hello world' | base64 > EncodedMessage

aws kms sign \
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
  --message fileb://EncodedMessage \
  --message-type RAW \
  --signing-algorithm RSASSA_PKCS1_V1_5_SHA_256 \
  --output text \
  --query Signature | base64 --decode > ExampleSignature
```

Este comando no genera ninguna salida. En este ejemplo, se extrae la `Signature` propiedad de la salida y se guarda en un archivo.

Para obtener más información sobre el uso de KMS claves asimétricas en AWS KMS, consulte [Claves asimétricas AWS KMS en la Guía para desarrolladores del Servicio de administración de AWS claves](#).

- Para API obtener más información, consulte la Referencia de AWS CLI comandos de [inicio de sesión](#).

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir una etiqueta a una KMS clave

En el siguiente `tag-resource` ejemplo, se añaden `"Purpose": "Test"` `"Dept": "IT"` etiquetas a una KMS clave gestionada por el cliente. Puede usar etiquetas como estas para etiquetar KMS las claves y crear categorías de KMS claves para los permisos y la auditoría.

Para especificar la KMS clave, utilice el `key-id` parámetro. En este ejemplo se utiliza un valor de identificador de clave, pero puede utilizar un identificador de clave o una clave ARN en este comando.

```
aws kms tag-resource \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --tags TagKey='Purpose',TagValue='Test' TagKey='Dept',TagValue='IT'
```

Este comando no genera ninguna salida. Para ver las etiquetas de una AWS KMS KMS tecla, usa el `list-resource-tags` comando.

Para obtener más información sobre el uso de etiquetas en AWS KMS, consulte [Etiquetado de claves](#) en la Guía para desarrolladores del Servicio de administración de AWS claves.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

## AWS CLI

Para eliminar una etiqueta de una KMS clave

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta con la "Purpose" clave de una KMS clave gestionada por el cliente.

Para especificar la KMS clave, utilice el `key-id` parámetro. En este ejemplo se utiliza un valor de identificador de clave, pero puede utilizar un identificador de clave o una clave ARN en este comando. Antes de ejecutar este comando, sustituya el identificador de clave del ejemplo por un identificador de clave válido de su AWS cuenta.

```
aws kms untag-resource \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --tag-key 'Purpose'
```

Este comando no genera ninguna salida. Para ver las etiquetas de una AWS KMS KMS tecla, usa el `list-resource-tags` comando.

Para obtener más información sobre el uso de etiquetas en AWS KMS, consulte [Etiquetado de claves](#) en la Guía para desarrolladores del Servicio de administración de AWS claves.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-alias

En el siguiente ejemplo de código, se muestra cómo usar `update-alias`.

## AWS CLI

Para asociar un alias a una KMS clave diferente

En el siguiente `update-alias` ejemplo, se asocia el alias `alias/test-key` a una KMS clave diferente.

El parámetro `--alias-name` especifica el alias. El valor del nombre del alias debe empezar por `alias/`. El `--target-key-id` parámetro especifica la KMS clave que se va a asociar al alias. No es necesario especificar la KMS clave actual del alias.

```
aws kms update-alias \  
  --alias-name alias/test-key \  
  --target-key-id
```



```
--target-key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Este comando no genera ninguna salida. Utilice el comando `list-aliases` para encontrar el alias.

Para obtener más información, consulte [Actualización de alias](#) en la AWS Guía para desarrolladores de Key Management Service.

- Para API obtener más información, consulte [UpdateAlias](#) la Referencia de AWS CLI comandos.

## update-custom-key-store

En el siguiente ejemplo de código, se muestra cómo usar `update-custom-key-store`.

### AWS CLI

Ejemplo 1: Para editar el nombre descriptivo de un almacén de claves personalizado

El siguiente `update-custom-key-store` ejemplo cambia el nombre del almacén de claves personalizado. Este ejemplo funciona para un almacén de HSM claves AWS en la nube o un almacén de claves externo.

Use el `custom-key-store-id` para identificar el almacén de claves. Utilice el `new-custom-key-store-name` parámetro para especificar el nuevo nombre descriptivo.

Para actualizar el nombre descriptivo de un almacén de HSM claves AWS en la nube, primero debe desconectar el almacén de claves, por ejemplo, mediante el `disconnect-custom-key-store` comando. Puede actualizar el nombre descriptivo de un almacén de claves externo mientras esté conectado o desconectado. Para averiguar el estado de conexión de su almacén de claves personalizado, utilice el `describe-custom-key-store` comando.

```
aws kms update-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0 \  
  --new-custom-key-store-name ExampleKeyStore
```

Este comando no devuelve ningún dato. Para comprobar que el comando ha funcionado, utilice un `describe-custom-key-stores` comando.

Para obtener más información sobre la actualización de un almacén de HSM claves AWS en la nube, consulte [Edición de la configuración del almacén de HSM claves en la AWS nube](#) en la Guía AWS para desarrolladores del servicio de administración de claves.

Para obtener más información sobre la actualización de un almacén de claves externo, consulte [Edición de las propiedades del almacén de claves externo](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.

Ejemplo 2: Para editar la contraseña `kmsuser` de un almacén de claves AWS en la nube HSM

En el siguiente `update-custom-key-store` ejemplo, se actualiza el valor de la `kmsuser` contraseña por la contraseña actual del `kmsuser` HSM clúster de nube asociado al almacén de claves especificado. Este comando no cambia la `kmsuser` contraseña del clúster. Simplemente indica AWS KMS la contraseña actual. Si KMS no tiene la `kmsuser` contraseña actual, no podrá conectarse al almacén de HSM claves de la AWS nube.

NOTE: Antes de actualizar un almacén de HSM claves AWS en la nube, debes desconectarlo. Utilice el comando `disconnect-custom-key-store`. Cuando se complete el comando, podrás volver a conectar el almacén de HSM claves de AWS Cloud. Utilice el comando `connect-custom-key-store`.

```
aws kms update-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0 \  
  --key-store-password ExamplePassword
```

El comando no devuelve ningún resultado. Para comprobar que el cambio se ha realizado correctamente, utiliza un `describe-custom-key-stores` comando.

Para obtener más información sobre la actualización de un almacén de HSM claves AWS en la nube, consulte [Edición de la configuración del almacén de HSM claves en la AWS nube](#) en la Guía AWS para desarrolladores del servicio de administración de claves.

Ejemplo 3: Para editar el HSM clúster AWS en la nube de un almacén de HSM claves AWS en la nube

En el siguiente ejemplo, se cambia el HSM clúster de AWS nube que está asociado a un almacén de HSM claves de la AWS nube por un clúster relacionado, por ejemplo, una copia de seguridad diferente del mismo clúster.

NOTE: Antes de actualizar un almacén de HSM claves AWS en la nube, debes desconectarlo. Utilice el comando `disconnect-custom-key-store`. Cuando se complete el comando, podrás volver a conectar el almacén de HSM claves de AWS Cloud. Utilice el comando `connect-custom-key-store`.

```
aws kms update-custom-key-store \  
  --key-store-id cks-1234567890abcdef0 \  
  --key-store-name ExampleKeyStoreName \  
  --key-store-type ExternalKeyStore \  
  --key-store-arn arn:aws:kms:us-east-1:123456789012:key-store:ExternalKeyStore \  
  --key-store-password ExamplePassword
```

```
--custom-key-store-id cks-1234567890abcdef0 \  
--cloud-hsm-cluster-id cluster-1a23b4cdefg
```

El comando no devuelve ningún resultado. Para comprobar que el cambio se ha realizado correctamente, utiliza un `describe-custom-key-stores` comando.

Para obtener más información sobre la actualización de un almacén de HSM claves AWS en la nube, consulte [Edición de la configuración del almacén de HSM claves en la AWS nube](#) en la Guía AWS para desarrolladores del servicio de administración de claves.

Ejemplo 4: Para editar la credencial de autenticación proxy de un almacén de claves externo

En el siguiente ejemplo, se actualiza la credencial de autenticación proxy del almacén de claves externo. Debe especificar tanto el como el `raw-secret-access-keyaccess-key-id`, incluso si va a cambiar solo uno de los valores. Puede utilizar esta función para corregir una credencial no válida o cambiarla cuando el proxy del almacén de claves externo la rota.

Establezca la credencial de autenticación del proxy AWS KMS en su almacén de claves externo. A continuación, utilice este comando para proporcionar la credencial a. AWS KMS AWS KMSutiliza esta credencial para firmar sus solicitudes en el proxy del almacén de claves externo.

Puede actualizar la credencial de autenticación del proxy mientras el almacén de claves externo esté conectado o desconectado. Para averiguar el estado de la conexión de su almacén de claves personalizado, utilice el `describe-custom-key-store` comando.

```
aws kms update-custom-key-store \  
--custom-key-store-id cks-1234567890abcdef0 \  
--xks-proxy-authentication-credential "AccessKeyId=ABCDE12345670EXAMPLE,  
RawSecretAccessKey=DXjSUawne12fr6SKC7G25CNxTyWKE5PF9XX6H/u9pSo="
```

El comando no devuelve ningún resultado. Para comprobar que el cambio se ha realizado correctamente, utilice un `describe-custom-key-stores` comando.

Para obtener más información sobre la actualización de un almacén de claves externo, consulte [Edición de las propiedades del almacén de claves externo](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.

Ejemplo 5: Para editar la conectividad proxy de un almacén de claves externo

El siguiente ejemplo cambia la opción de conectividad proxy del almacén de claves externo de la conectividad de punto final público a la conectividad de servicio de VPC punto final. Además de

cambiar el `xks-proxy-connectivity` valor, debe cambiarlo para que refleje el DNS nombre privado asociado al servicio de VPC punto final. `xks-proxy-uri-endpoint` También debe añadir un `xks-proxy-vpc-endpoint-service-name` valor.

NOTE: Antes de actualizar la conectividad proxy de un almacén externo, debe desconectarlo. Utilice el comando `disconnect-custom-key-store`. Una vez finalizado el comando, puede volver a conectar el almacén de claves externo mediante el `connect-custom-key-store` comando.

```
aws kms update-custom-key-store \  
  --custom-key-store-id cks-1234567890abcdef0 \  
  --xks-proxy-connectivity VPC_ENDPOINT_SERVICE \  
  --xks-proxy-uri-endpoint "https://myproxy-private.xks.example.com" \  
  --xks-proxy-vpc-endpoint-service-name "com.amazonaws.vpce.us-east-1.vpce-svc-  
example"
```

El comando no devuelve ningún resultado. Para comprobar que el cambio se ha realizado correctamente, utilice un `describe-custom-key-stores` comando.

Para obtener más información sobre la actualización de un almacén de claves externo, consulte [Edición de las propiedades del almacén de claves externo](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.

- Para API obtener más información, consulte [UpdateCustomKeyStore](#) la Referencia de AWS CLI comandos.

## update-key-description

En el siguiente ejemplo de código, se muestra cómo usar `update-key-description`.

### AWS CLI

Ejemplo 1: Para añadir o cambiar la descripción de una KMS clave gestionada por el cliente

En el siguiente `update-key-description` ejemplo, se añade una descripción a una KMS clave gestionada por el cliente. Puede usar el mismo comando para cambiar una descripción existente.

El `--key-id` parámetro identifica la KMS clave del comando. En este ejemplo se utiliza un ARN valor de clave, pero se puede utilizar el identificador de la clave o la clave ARN de la KMS

clave. El `--description` parámetro especifica la nueva descripción. El valor de este parámetro reemplaza a la descripción actual de la KMS clave, si la hubiera.

```
aws kms update-key-description \  
  --key-id arn:aws:kms:us-  
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab \  
  --description "IT Department test key"
```

Este comando no genera ninguna salida. Para ver la descripción de una KMS clave, utilice el `describe-key` comando.

Para obtener más información, consulte [UpdateKeyDescription](#) la API referencia del servicio de administración de AWS claves.

Ejemplo 2: Para eliminar la descripción de una KMS clave gestionada por el cliente

En el siguiente `update-key-description` ejemplo, se elimina la descripción de una KMS clave gestionada por el cliente.

El `--key-id` parámetro identifica la KMS clave del comando. En este ejemplo se utiliza un valor de identificador de clave, pero se puede utilizar el identificador de clave o la clave ARN de la KMS clave. El `--description` parámetro con un valor de cadena vacío («») elimina la descripción existente.

```
aws kms update-key-description \  
  --key-id 0987dcba-09fe-87dc-65ba-ab0987654321 \  
  --description ''
```

Este comando no genera ninguna salida. Para ver la descripción de una KMS clave, utilice el comando `describe-key`.

Para obtener más información, consulte la referencia del [UpdateKeyDescription](#) servicio de administración de AWS claves. API

- Para API obtener más información, consulte [UpdateKeyDescription](#) la Referencia de AWS CLI comandos.

## verify

En el siguiente ejemplo de código, se muestra cómo usar `verify`.

## AWS CLI

Para verificar una firma digital

El siguiente `verify` ejemplo verifica la firma criptográfica de un mensaje corto codificado en Base64. El identificador de clave, el mensaje, el tipo de mensaje y el algoritmo de firma deben ser los mismos que se usaron para firmar el mensaje. La firma que especifique no puede estar codificada en base64. Para obtener ayuda para decodificar la firma que devuelve el `sign` comando, consulte los `sign` ejemplos de comandos.

El resultado del comando incluye un `SignatureValid` campo booleano que indica que se verificó la firma. Si se produce un error en la validación de la firma, también se produce un error en el `verify` comando.

Antes de ejecutar este comando, sustituya el identificador de clave de ejemplo por un identificador de clave válido de su AWS cuenta.

```
aws kms verify \  
  --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \  
  --message fileb://EncodedMessage \  
  --message-type RAW \  
  --signing-algorithm RSASSA_PKCS1_V1_5_SHA_256 \  
  --signature fileb://ExampleSignature
```

Salida:

```
{  
  "KeyId": "arn:aws:kms:us-  
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",  
  "SignatureValid": true,  
  "SigningAlgorithm": "RSASSA_PKCS1_V1_5_SHA_256"  
}
```

Para obtener más información sobre el uso de KMS claves asimétricas en AWS KMS, consulte [Uso de claves asimétricas](#) en la Guía para desarrolladores del Servicio de administración de AWS claves.

- Para API obtener más información, consulte [Verificar en la referencia](#) de AWS CLI comandos.

## Ejemplos de Lake Formation usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Lake Formation.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **add-lf-tags-to-resource**

En el siguiente ejemplo de código, se muestra cómo usar `add-lf-tags-to-resource`.

AWS CLI

Para adjuntar una o más etiquetas LF a un recurso existente

En el siguiente `add-lf-tags-to-resource` ejemplo, se adjunta una etiqueta LF determinada al recurso de la tabla.

```
aws lakeformation add-lf-tags-to-resource \  
--cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "CatalogId": "123456789111",  
  "Resource": {  
    "Table": {  
      "CatalogId": "123456789111",  
      "DatabaseName": "tpc",  
      "Name": "dl_tpc_promotion"  
    }  
  }  
}
```

```

    },
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "analyst"
      ]
    }]
  }
}

```

Salida:

```

{
  "Failures": []
}

```

Para obtener más información, consulte [Asignación de etiquetas LF a los recursos del catálogo de datos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte la Referencia de [AddLfTagsToResource](#) comandos AWS CLI .

## batch-grant-permissions

En el siguiente ejemplo de código, se muestra cómo usar batch-grant-permissions.

AWS CLI

Para conceder permisos de forma masiva sobre los recursos a los directores

En el siguiente batch-grant-permissions ejemplo, se concede acceso masivo a los principales a recursos específicos.

```

aws lakeformation batch-grant-permissions \
  --cli-input-json file://input.json

```

Contenidos de input.json:

```

{
  "CatalogId": "123456789111",
  "Entries": [{
    "Id": "1",

```



```
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
developer"
    },
    "Resource": {
      "Table": {
        "CatalogId": "123456789111",
        "DatabaseName": "tpc",
        "Name": "dl_tpc_promotion"
      }
    },
    "Permissions": [
      "ALL"
    ],
    "PermissionsWithGrantOption": [
      "ALL"
    ]
  },
  {
    "Id": "2",
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
developer"
    },
    "Resource": {
      "Table": {
        "CatalogId": "123456789111",
        "DatabaseName": "tpc",
        "Name": "dl_tpc_customer"
      }
    },
    "Permissions": [
      "ALL"
    ],
    "PermissionsWithGrantOption": [
      "ALL"
    ]
  },
  {
    "Id": "3",
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
business-analyst"
    },
```

```

    "Resource": {
      "Table": {
        "CatalogId": "123456789111",
        "DatabaseName": "tpc",
        "Name": "dl_tpc_promotion"
      }
    },
    "Permissions": [
      "ALL"
    ],
    "PermissionsWithGrantOption": [
      "ALL"
    ]
  },
  {
    "Id": "4",
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
developer"
    },
    "Resource": {
      "DataCellsFilter": {
        "TableCatalogId": "123456789111",
        "DatabaseName": "tpc",
        "TableName": "dl_tpc_item",
        "Name": "developer_item"
      }
    },
    "Permissions": [
      "SELECT"
    ],
    "PermissionsWithGrantOption": []
  }
]
}

```

Salida:

```

{
  "Failures": []
}

```

Para obtener más información, consulte [Concesión y revocación de permisos sobre los recursos del catálogo de datos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [BatchGrantPermissions](#) la Referencia de AWS CLI comandos.

## batch-revoke-permissions

En el siguiente ejemplo de código, se muestra cómo usar `batch-revoke-permissions`.

### AWS CLI

Para revocar de forma masiva los permisos de los directores sobre los recursos

En el siguiente `batch-revoke-permissions` ejemplo, se revoca de forma masiva el acceso de los principales a recursos específicos.

```
aws lakeformation batch-revoke-permissions \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "CatalogId": "123456789111",  
  "Entries": [{  
    "Id": "1",  
    "Principal": {  
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-  
developer"  
    },  
    "Resource": {  
      "Table": {  
        "CatalogId": "123456789111",  
        "DatabaseName": "tpc",  
        "Name": "dl_tpc_promotion"  
      }  
    },  
    "Permissions": [  
      "ALL"  
    ],  
    "PermissionsWithGrantOption": [  
      "ALL"  
    ]  
  }  
]
```

```
    },
    {
      "Id": "2",
      "Principal": {
        "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
business-analyst"
      },
      "Resource": {
        "Table": {
          "CatalogId": "123456789111",
          "DatabaseName": "tpc",
          "Name": "dl_tpc_promotion"
        }
      },
      "Permissions": [
        "ALL"
      ],
      "PermissionsWithGrantOption": [
        "ALL"
      ]
    }
  ]
}
```

Salida:

```
{
  "Failures": []
}
```

Para obtener más información, consulte [Concesión y revocación de permisos sobre los recursos del catálogo de datos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [BatchRevokePermissions](#) la Referencia de AWS CLI comandos.

## cancel-transaction

En el siguiente ejemplo de código, se muestra cómo usar `cancel-transaction`.

### AWS CLI

Para cancelar una transacción

En el siguiente `cancel-transaction` ejemplo, se cancela la transacción.

```
aws lakeformation cancel-transaction \  
  --transaction-id='b014d972ca8347b89825e33c5774aec4'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Leer y escribir en el lago de datos dentro de las transacciones](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [CancelTransaction](#) la Referencia de AWS CLI comandos.

## **commit-transaction**

En el siguiente ejemplo de código, se muestra cómo usar `commit-transaction`.

### AWS CLI

Para confirmar la transacción

El siguiente `commit-transaction` ejemplo confirma la transacción.

```
aws lakeformation commit-transaction \  
  --transaction-id='b014d972ca8347b89825e33c5774aec4'
```

Salida:

```
{  
  "TransactionStatus": "committed"  
}
```

Para obtener más información, consulte [Leer y escribir en el lago de datos dentro de las transacciones](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [CommitTransaction](#) la Referencia de AWS CLI comandos.

## **create-data-cells-filter**

En el siguiente ejemplo de código, se muestra cómo usar `create-data-cells-filter`.

## AWS CLI

Ejemplo 1: Para crear un filtro de celdas de datos

El siguiente `create-data-cells-filter` ejemplo crea un filtro de celdas de datos para permitir el acceso a determinadas columnas en función del estado de la fila.

```
aws lakeformation create-data-cells-filter \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "TableData": {  
    "ColumnNames": ["p_channel_details", "p_start_date_sk", "p_promo_name"],  
    "DatabaseName": "tpc",  
    "Name": "developer_promotion",  
    "RowFilter": {  
      "FilterExpression": "p_promo_name='ese'"  
    },  
    "TableCatalogId": "123456789111",  
    "TableName": "dl_tpc_promotion"  
  }  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Filtrado de datos y seguridad a nivel de celda en Lake Formation en](#) la Guía para desarrolladores de AWS Lake Formation.

Ejemplo 2: Para crear un filtro de columnas

El siguiente `create-data-cells-filter` ejemplo crea un filtro de datos para permitir el acceso a determinadas columnas.

```
aws lakeformation create-data-cells-filter \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "TableData": {
```

```

    "ColumnNames": ["p_channel_details", "p_start_date_sk", "p_promo_name"],
    "DatabaseName": "tpc",
    "Name": "developer_promotion_allrows",
    "RowFilter": {
      "AllRowsWildcard": {}
    },
    "TableCatalogId": "123456789111",
    "TableName": "dl_tpc_promotion"
  }
}

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Filtrado de datos y seguridad a nivel de celda en Lake Formation en](#) la Guía para desarrolladores de AWS Lake Formation.

Ejemplo 3: Para crear un filtro de datos con columnas excluidas

El siguiente `create-data-cells-filter` ejemplo crea un filtro de datos para permitir el acceso a todas las columnas excepto a las mencionadas.

```

aws lakeformation create-data-cells-filter \
  --cli-input-json file://input.json

```

Contenidos de `input.json`:

```

{
  "TableData": {
    "ColumnWildcard": {
      "ExcludedColumnNames": ["p_channel_details", "p_start_date_sk"]
    },
    "DatabaseName": "tpc",
    "Name": "developer_promotion_excludecolumn",
    "RowFilter": {
      "AllRowsWildcard": {}
    },
    "TableCatalogId": "123456789111",
    "TableName": "dl_tpc_promotion"
  }
}

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Filtrado de datos y seguridad a nivel de celda en Lake Formation en](#) la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte la Referencia [CreateDataCellsFilter](#) de AWS CLI comandos.

## **create-lf-tag**

En el siguiente ejemplo de código, se muestra cómo usar `create-lf-tag`.

### AWS CLI

Para crear una etiqueta LF

En el siguiente `create-lf-tag` ejemplo, se crea una etiqueta LF con el nombre y los valores especificados.

```
aws lakeformation create-lf-tag \  
  --catalog-id '123456789111' \  
  --tag-key 'usergroup' \  
  --tag-values '["developer","analyst","campaign"]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de etiquetas LF para el control de acceso a los metadatos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte la Referencia [CreateLfTag](#) de AWS CLI comandos.

## **delete-data-cells-filter**

En el siguiente ejemplo de código, se muestra cómo usar `delete-data-cells-filter`.

### AWS CLI

Para eliminar el filtro de celdas de datos

El siguiente `delete-data-cells-filter` ejemplo elimina un filtro de celda de datos determinado.

```
aws lakeformation delete-data-cells-filter \  
  --cli-input-json file://input.json
```



Contenidos de `input.json`:

```
{
  "TableCatalogId": "123456789111",
  "DatabaseName": "tpc",
  "TableName": "dl_tpc_promotion",
  "Name": "developer_promotion"
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Filtrado de datos y seguridad a nivel de celda en Lake Formation en](#) la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte la Referencia [DeleteDataCellsFilter](#) de AWS CLI comandos.

## **delete-lf-tag**

En el siguiente ejemplo de código, se muestra cómo usar `delete-lf-tag`.

AWS CLI

Para eliminar la definición de etiqueta LF

En el siguiente `delete-lf-tag` ejemplo, se elimina la definición de etiqueta LF.

```
aws lakeformation delete-lf-tag \
  --catalog-id '123456789111' \
  --tag-key 'usergroup'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de etiquetas LF para el control de acceso a los metadatos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte la Referencia [DeleteLfTag](#) de AWS CLI comandos.

## **delete-objects-on-cancel**

En el siguiente ejemplo de código, se muestra cómo usar `delete-objects-on-cancel`.

## AWS CLI

Para eliminar un objeto cuando se cancela la transacción

En el siguiente `delete-objects-on-cancel` ejemplo, se elimina el objeto `s3` de la lista cuando se cancela la transacción.

```
aws lakeformation delete-objects-on-cancel \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "CatalogId": "012345678901",  
  "DatabaseName": "tpc",  
  "TableName": "dl_tpc_household_demographics_gov",  
  "TransactionId": "1234d972ca8347b89825e33c5774aec4",  
  "Objects": [{  
    "Uri": "s3://lf-data-lake-012345678901/target/  
dl_tpc_household_demographics_gov/run-unnamed-1-part-block-0-r-00000-snappy-  
ff26b17504414fe88b302cd795eabd00.parquet",  
    "ETag": "1234ab1fc50a316b149b4e1f21a73800"  
  }]  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Leer y escribir en el lago de datos dentro de las transacciones](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [DeleteObjectsOnCancel](#) Referencia de AWS CLI comandos.

## **deregister-resource**

En el siguiente ejemplo de código, se muestra cómo usar `deregister-resource`.

## AWS CLI

Para anular el registro del almacenamiento de data lake

En el siguiente `deregister-resource` ejemplo se anula el registro del recurso tal y como lo administra Lake Formation.

```
aws lakeformation deregister-resource \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "ResourceArn": "arn:aws:s3:::lf-emr-athena-result-123"  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir una ubicación de Amazon S3 a su lago de datos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [DeregisterResource](#) la Referencia de AWS CLI comandos.

## **describe-transaction**

En el siguiente ejemplo de código, se muestra cómo usar `describe-transaction`.

### AWS CLI

Para recuperar los detalles de una transacción

El siguiente `describe-transaction` ejemplo devuelve los detalles de una sola transacción.

```
aws lakeformation describe-transaction \  
  --transaction-id='8cb4b1a7cc8d486fbaca9a64e7d9f5ce'
```

Salida:

```
{  
  "TransactionDescription": {  
    "TransactionId": "12345972ca8347b89825e33c5774aec4",  
    "TransactionStatus": "committed",  
    "TransactionStartTime": "2022-08-10T14:29:04.046000+00:00",  
    "TransactionEndTime": "2022-08-10T14:29:09.681000+00:00"  
  }  
}
```

Para obtener más información, consulte [Leer y escribir en el lago de datos dentro de las transacciones](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [DescribeTransaction](#) la Referencia de AWS CLI comandos.

## **extend-transaction**

En el siguiente ejemplo de código, se muestra cómo usar `extend-transaction`.

### AWS CLI

Para extender una transacción

El siguiente `extend-transaction` ejemplo amplía la transacción.

```
aws lakeformation extend-transaction \  
  --transaction-id='8cb4b1a7cc8d486fbaca9a64e7d9f5ce'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Leer y escribir en el lago de datos dentro de las transacciones](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [ExtendTransaction](#) la Referencia de AWS CLI comandos.

## **get-data-lake-settings**

En el siguiente ejemplo de código, se muestra cómo usar `get-data-lake-settings`.

### AWS CLI

Para recuperar la configuración de un AWS lago de datos gestionado por Lake Formation

En el siguiente `get-data-lake-settings` ejemplo, se recupera la lista de administradores de lagos de datos y otros ajustes de los lagos de datos.

```
aws lakeformation get-data-lake-settings \  
  --cli-input-json file://input.json
```

## Contenidos de `input.json`:

```
{
  "CatalogId": "123456789111"
}
```

## Salida:

```
{
  "DataLakeSettings": {
    "DataLakeAdmins": [{
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-admin"
    }],
    "CreateDatabaseDefaultPermissions": [],
    "CreateTableDefaultPermissions": [
      {
        "Principal": {
          "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
        },
        "Permissions": [
          "ALL"
        ]
      }
    ],
    "TrustedResourceOwners": [],
    "AllowExternalDataFiltering": true,
    "ExternalDataFilteringAllowList": [{
      "DataLakePrincipalIdentifier": "123456789111"
    }],
    "AuthorizedSessionTagValueList": [
      "Amazon EMR"
    ]
  }
}
```

Para obtener más información, consulte [Cambiar la configuración de seguridad predeterminada de su lago de datos en la Guía para desarrolladores de AWS Lake Formation](#).

- Para API obtener más información, consulte [GetDataLakeSettings](#) la Referencia de AWS CLI comandos.

## get-effective-permissions-for-path

En el siguiente ejemplo de código, se muestra cómo usar `get-effective-permissions-for-path`.

### AWS CLI

Para recuperar los permisos de los recursos ubicados en una ruta específica

El siguiente `get-effective-permissions-for-path` ejemplo devuelve los permisos de Lake Formation para un recurso de tabla o base de datos especificado ubicado en una ruta de Amazon S3.

```
aws lakeformation get-effective-permissions-for-path \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "CatalogId": "123456789111",  
  "ResourceArn": "arn:aws:s3:::lf-data-lake-123456789111"  
}
```

Salida:

```
{  
  "Permissions": [{  
    "Principal": {  
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-  
campaign-manager"  
    },  
    "Resource": {  
      "Database": {  
        "Name": "tpc"  
      }  
    },  
    "Permissions": [  
      "DESCRIBE"  
    ],  
    "PermissionsWithGrantOption": []  
  }],  
  {
```

```
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/EMR-
RuntimeRole"
    },
    "Resource": {
      "Database": {
        "Name": "tpc"
      }
    },
    "Permissions": [
      "ALL"
    ],
    "PermissionsWithGrantOption": []
  },
  {
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:saml-
provider/oktaSAMLProvider:user/emr-developer"
    },
    "Resource": {
      "Database": {
        "Name": "tpc"
      }
    },
    "Permissions": [
      "ALL",
      "DESCRIBE"
    ],
    "PermissionsWithGrantOption": []
  },
  {
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
admin"
    },
    "Resource": {
      "Database": {
        "Name": "tpc"
      }
    },
    "Permissions": [
      "ALL",
      "ALTER",
      "CREATE_TABLE",
```

```

        "DESCRIBE",
        "DROP"
    ],
    "PermissionsWithGrantOption": [
        "ALL",
        "ALTER",
        "CREATE_TABLE",
        "DESCRIBE",
        "DROP"
    ]
},
{
    "Principal": {
        "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/LF-
GlueServiceRole"
    },
    "Resource": {
        "Database": {
            "Name": "tpc"
        }
    },
    "Permissions": [
        "CREATE_TABLE"
    ],
    "PermissionsWithGrantOption": []
}
],
"NextToken":
"E5S1JDSTZ1eUp6SWpvaU9UQTN0RE0zTXpFeE5Ua3pJbjE5TENKbGVIQnBjbUYwYVc5dU1qcDdJbk5sWTI5dVpITW1P
}

```

Para obtener más información, consulte [Gestión de los permisos de Lake Formation](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [GetEffectivePermissionsForPath](#) la Referencia de AWS CLI comandos.

## get-lf-tag

En el siguiente ejemplo de código, se muestra cómo usar `get-lf-tag`.



## AWS CLI

Para recuperar la definición de etiqueta LF

En el siguiente `get-lf-tag` ejemplo, se recupera la definición de la etiqueta LF.

```
aws lakeformation get-lf-tag \  
  --catalog-id '123456789111' \  
  --tag-key 'usergroup'
```

Salida:

```
{  
  "CatalogId": "123456789111",  
  "TagKey": "usergroup",  
  "TagValues": [  
    "analyst",  
    "campaign",  
    "developer"  
  ]  
}
```

Para obtener más información, consulte [Administración de etiquetas LF para el control de acceso a los metadatos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte la Referencia [GetLfTag](#) de AWS CLI comandos.

## get-query-state

En el siguiente ejemplo de código, se muestra cómo usar `get-query-state`.

### AWS CLI

Para recuperar el estado de una consulta enviada

El siguiente `get-query-state` ejemplo devuelve el estado de una consulta enviada anteriormente.

```
aws lakeformation get-query-state \  
  --query-id='1234273f-4a62-4cda-8d98-69615ee8be9b'
```

Salida:

```
{
  "State": "FINISHED"
}
```

Para obtener más información, consulte [Operaciones de datos transaccionales](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [GetQueryState](#) la Referencia de AWS CLI comandos.

## get-query-statistics

En el siguiente ejemplo de código, se muestra cómo usar `get-query-statistics`.

AWS CLI

Para recuperar las estadísticas de las consultas

En el siguiente `get-query-statistics` ejemplo, se recuperan las estadísticas sobre la planificación y la ejecución de una consulta.

```
aws lakeformation get-query-statistics \
  --query-id='1234273f-4a62-4cda-8d98-69615ee8be9b'
```

Salida:

```
{
  "ExecutionStatistics": {
    "AverageExecutionTimeMillis": 0,
    "DataScannedBytes": 0,
    "WorkUnitsExecutedCount": 0
  },
  "PlanningStatistics": {
    "EstimatedDataToScanBytes": 43235,
    "PlanningTimeMillis": 2377,
    "QueueTimeMillis": 440,
    "WorkUnitsGeneratedCount": 1
  },
  "QuerySubmissionTime": "2022-08-11T02:14:38.641870+00:00"
}
```

```
}
```

Para obtener más información, consulte [Operaciones de datos transaccionales](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [GetQueryStatistics](#) la Referencia de AWS CLI comandos.

## get-resource-lf-tags

En el siguiente ejemplo de código, se muestra cómo usar `get-resource-lf-tags`.

### AWS CLI

Para enumerar las etiquetas LF

El siguiente `list-lf-tags` ejemplo devuelve una lista de etiquetas LF que el solicitante tiene permiso para ver.

```
aws lakeformation list-lf-tags \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "CatalogId": "123456789111",  
  "ResourceShareType": "ALL",  
  "MaxResults": 2  
}
```

Salida:

```
{  
  "LFTags": [{  
    "CatalogId": "123456789111",  
    "TagKey": "category",  
    "TagValues": [  
      "private",  
      "public"  
    ]  
  }],  
}
```

```
{
  "CatalogId": "123456789111",
  "TagKey": "group",
  "TagValues": [
    "analyst",
    "campaign",
    "developer"
  ]
}],
"NextToken": "kIiwiZXhwaXJhdGlvbiI6eyJzZWVvbmRzIjoxNjYwMDY4dCI6ZmFsc2V9"
}
```

Para obtener más información, consulte [Administración de etiquetas LF para el control de acceso a los metadatos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte la Referencia [GetResourceLfTags](#) de AWS CLI comandos.

## get-table-objects

En el siguiente ejemplo de código, se muestra cómo usar `get-table-objects`.

### AWS CLI

Para enumerar los objetos de la tabla gobernada

El siguiente `get-table-objects` ejemplo devuelve el conjunto de objetos de Amazon S3 que componen la tabla gobernada especificada.

```
aws lakeformation get-table-objects \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "CatalogId": "012345678901",
  "DatabaseName": "tpc",
  "TableName": "dl_tpc_household_demographics_gov",
  "QueryAsOfTime": "2022-08-10T15:00:00"
}
```

Salida:

```
{
  "Objects": [{
    "PartitionValues": [],
    "Objects": [{
      "Uri": "s3://lf-data-lake-012345678901/target/
dl_tpc_household_demographics_gov/run-unnamed-1-part-block-0-r-00000-snappy-
ff26b17504414fe88b302cd795eabd00.parquet",
      "ETag": "12345b1fc50a316b149b4e1f21a73800",
      "Size": 43235
    }]
  }]
}
```

Para obtener más información, consulte [Leer y escribir en el lago de datos dentro de las transacciones](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [GetTableObjects](#) la Referencia de AWS CLI comandos.

## get-work-unit-results

En el siguiente ejemplo de código, se muestra cómo usar `get-work-unit-results`.

### AWS CLI

Para recuperar las unidades de trabajo de una consulta determinada

El siguiente `get-work-unit-results` ejemplo devuelve las unidades de trabajo resultantes de la consulta.

```
aws lakeformation get-work-units \
  --query-id='1234273f-4a62-4cda-8d98-69615ee8be9b' \
  --work-unit-id '0' \
  --work-unit-token 'B2fMSdmQXe9umX8Ux8XCo4=' outfile
```

Salida:

```
outfile with Blob content.
```

Para obtener más información, consulte [Operaciones de datos transaccionales](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [GetWorkUnitResults](#) la Referencia de AWS CLI comandos.

## get-work-units

En el siguiente ejemplo de código, se muestra cómo usar `get-work-units`.

### AWS CLI

Para recuperar unidades de trabajo

En el siguiente `get-work-units` ejemplo, se recuperan las unidades de trabajo generadas por la `StartQueryPlanning` operación.

```
aws lakeformation get-work-units \
  --query-id='1234273f-4a62-4cda-8d98-69615ee8be9b'
```

Salida:

```
{
  "WorkUnitRanges": [{
    "WorkUnitIdMax": 0,
    "WorkUnitIdMin": 0,
    "WorkUnitToken":
      "1234eMAk4kL04umqEL4Z5WuxL04AXwABABVhd3MtY3J5cHRvLXB1YmxpYy1rZXkAREEwYm9QbkhINmFYTWphbmMxZW
      +f88jzGrYq22gE6jkQlp0B
      +0et2eqNumFudAAAAfjB8BgkqhkiG9w0BBwagbzBtAgEAMGgGCSqGSIb3DQEHAeBg1ghkgBZQMEAS4wEQQMCOEWRda
      wAAAAEAAAAAAAAAAAAAAAAEAAACX3/w5h75QAPomfKH+cyEKYU1yccUmBl
      +VSojiG0tdsUk7vcjYXUUboYm3dvqRqX2s4gROM0n
      +Ij8R0/8jYmnHkpvYAFNVRPyETyIKg7k5Z9+5I1c2d3446Jw/moWGGxjH8AEG9h27ytm0hozxDOEi/
      F2ZoXz6w1GDfGUo/2WxCkY0hTyNaw6TM
      +7drTM7yrW4iNVLUM0LX0xnFjIAhLhooWJek6vjQZUAZzB1AjBH8okRtYP8R7AY2W1s/
      hqFBhG0V4142AC0LxsuZbMQrE2SzwZUZ0E9Uew7/n0cyX4CMQDR79INyv4ysMByW9kKGGKyba+cCNk1ExMR
      +btBQBmMuB2fMSdmQXe9umX8Ux8XCo4="
  }],
  "QueryId": "1234273f-4a62-4cda-8d98-69615ee8be9b"
}
```

Para obtener más información, consulte [Operaciones de datos transaccionales](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [GetWorkUnits](#) la Referencia de AWS CLI comandos.

## grant-permissions

En el siguiente ejemplo de código, se muestra cómo usar `grant-permissions`.

### AWS CLI

Ejemplo 1: Para conceder permisos al director sobre los recursos mediante etiquetas LF

El siguiente `grant-permissions` ejemplo concede ALL permisos al principal sobre un recurso de base de datos que coincide con la política de etiquetas LF.

```
aws lakeformation grant-permissions \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "CatalogId": "123456789111",  
  "Principal": {  
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-admin"  
  },  
  "Resource": {  
    "LFTagPolicy": {  
      "CatalogId": "123456789111",  
      "ResourceType": "DATABASE",  
      "Expression": [{  
        "TagKey": "usergroup",  
        "TagValues": [  
          "analyst",  
          "developer"  
        ]  
      }]  
    }  
  },  
  "Permissions": [  
    "ALL"  
  ],  
  "PermissionsWithGrantOption": [  
    "ALL"  
  ]  
}
```

```
]
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Concesión y revocación de permisos sobre los recursos del catálogo de datos](#) en la Guía para desarrolladores de AWS Lake Formation.

Ejemplo 2: Para conceder permisos a nivel de columna al director

El siguiente `grant-permissions` ejemplo otorga permiso al principal para seleccionar una columna específica.

```
aws lakeformation grant-permissions \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "CatalogId": "123456789111",
  "Principal": {
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
  },
  "Resource": {
    "TableWithColumns": {
      "CatalogId": "123456789111",
      "ColumnNames": ["p_end_date_sk"],
      "DatabaseName": "tpc",
      "Name": "dl_tpc_promotion"
    }
  },
  "Permissions": [
    "SELECT"
  ],
  "PermissionsWithGrantOption": []
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Concesión y revocación de permisos sobre los recursos del catálogo de datos](#) en la Guía para desarrolladores de AWS Lake Formation.



### Ejemplo 3: Para conceder permisos de tabla al director

El siguiente `grant-permissions` ejemplo concede al director el permiso de selección en todas las tablas de una base de datos determinada.

```
aws lakeformation grant-permissions \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "CatalogId": "123456789111",  
  "Principal": {  
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"  
  },  
  "Resource": {  
    "Table": {  
      "CatalogId": "123456789111",  
      "DatabaseName": "tpc",  
      "TableWildcard": {}  
    }  
  },  
  "Permissions": [  
    "SELECT"  
  ],  
  "PermissionsWithGrantOption": []  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Concesión y revocación de permisos sobre los recursos del catálogo de datos](#) en la Guía para desarrolladores de AWS Lake Formation.

### Ejemplo 4: Para conceder permisos sobre las etiquetas LF al director

En el siguiente `grant-permissions` ejemplo, se concede al director un permiso de asociación sobre las etiquetas LF.

```
aws lakeformation grant-permissions \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "CatalogId": "123456789111",
  "Principal": {
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
  },
  "Resource": {
    "LFTag": {
      "CatalogId": "123456789111",
      "TagKey": "category",
      "TagValues": [
        "private", "public"
      ]
    }
  },
  "Permissions": [
    "ASSOCIATE"
  ],
  "PermissionsWithGrantOption": []
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Concesión y revocación de permisos sobre los recursos del catálogo de datos](#) en la Guía para desarrolladores de AWS Lake Formation.

Ejemplo 5: Otorgar permisos sobre las ubicaciones de los datos al director

El siguiente `grant-permissions` ejemplo concede permiso sobre la ubicación de los datos al principal.

```
aws lakeformation grant-permissions \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "CatalogId": "123456789111",
  "Principal": {
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
  },
}
```

```
"Resource": {
  "DataLocation": {
    "CatalogId": "123456789111",
    "ResourceArn": "arn:aws:s3:::lf-data-lake-123456789111"
  }
},
"Permissions": [
  "DATA_LOCATION_ACCESS"
],
"PermissionsWithGrantOption": []
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Concesión y revocación de permisos sobre los recursos del catálogo de datos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [GrantPermissions](#) la Referencia de AWS CLI comandos.

## list-data-cells-filter

En el siguiente ejemplo de código, se muestra cómo usar `list-data-cells-filter`.

### AWS CLI

Para enumerar los filtros de celdas de datos

El siguiente `list-data-cells-filter` ejemplo muestra el filtro de celda de datos para una tabla determinada.

```
aws lakeformation list-data-cells-filter \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "MaxResults": 2,
  "Table": {
    "CatalogId": "123456789111",
    "DatabaseName": "tpc",
```

```
    "Name": "dl_tpc_promotion"
  }
}
```

**Salida:**

```
{
  "DataCellsFilters": [{
    "TableCatalogId": "123456789111",
    "DatabaseName": "tpc",
    "TableName": "dl_tpc_promotion",
    "Name": "developer_promotion",
    "RowFilter": {
      "FilterExpression": "p_promo_name='ese'"
    },
    "ColumnNames": [
      "p_channel_details",
      "p_start_date_sk",
      "p_purpose",
      "p_promo_id",
      "p_promo_name",
      "p_end_date_sk",
      "p_discount_active"
    ]
  },
  {
    "TableCatalogId": "123456789111",
    "DatabaseName": "tpc",
    "TableName": "dl_tpc_promotion",
    "Name": "developer_promotion_allrows",
    "RowFilter": {
      "FilterExpression": "TRUE",
      "AllRowsWildcard": {}
    },
    "ColumnNames": [
      "p_channel_details",
      "p_start_date_sk",
      "p_promo_name"
    ]
  }
],
  "NextToken": "2MDA2MTgwNiwibmFub3MiOjE0MDAwMDAwMH19"
}
```

Para obtener más información, consulte [Filtrado de datos y seguridad a nivel de celda en Lake Formation en](#) la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte la Referencia [ListDataCellsFilter](#) de AWS CLI comandos.

## list-permissions

En el siguiente ejemplo de código, se muestra cómo usar `list-permissions`.

### AWS CLI

Ejemplo 1: Para recuperar la lista de los permisos principales del recurso

El siguiente `list-permissions` ejemplo devuelve una lista de los permisos principales de los recursos de la base de datos.

```
aws lakeformation list-permissions \  
--cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "CatalogId": "123456789111",  
  "ResourceType": "DATABASE",  
  "MaxResults": 2  
}
```

Salida:

```
{  
  "PrincipalResourcePermissions": [{  
    "Principal": {  
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-  
campaign-manager"  
    },  
    "Resource": {  
      "Database": {  
        "CatalogId": "123456789111",  
        "Name": "tpc"  
      }  
    }  
  }],  
}
```

```

    "Permissions": [
      "DESCRIBE"
    ],
    "PermissionsWithGrantOption": []
  ]],
  "NextToken":
  "E5S1JDSTZ1eUp6SWpvaU9UQTN0RE0zTXpFeE5Ua3pJbjE5TENKbGVIQnBjbUYwYVc5dUlqcDdJbk5sWTI5dVpITW1P
}

```

Para obtener más información, consulte [Gestión de los permisos de Lake Formation](#) en la Guía para desarrolladores de AWS Lake Formation.

Ejemplo 2: Para recuperar la lista de permisos principales de la tabla con filtros de datos

En el siguiente `list-permissions` ejemplo, se enumeran los permisos de la tabla con los filtros de datos relacionados concedidos al principal.

```

aws lakeformation list-permissions \
  --cli-input-json file://input.json

```

Contenidos de `input.json`:

```

{
  "CatalogId": "123456789111",
  "Resource": {
    "Table": {
      "CatalogId": "123456789111",
      "DatabaseName": "tpc",
      "Name": "dl_tpc_customer"
    }
  },
  "IncludeRelated": "TRUE",
  "MaxResults": 10
}

```

Salida:

```

{
  "PrincipalResourcePermissions": [{
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/
Admin"

```

```
    },
    "Resource": {
      "Table": {
        "CatalogId": "123456789111",
        "DatabaseName": "customer",
        "Name": "customer_invoice"
      }
    },
    "Permissions": [
      "ALL",
      "ALTER",
      "DELETE",
      "DESCRIBE",
      "DROP",
      "INSERT"
    ],
    "PermissionsWithGrantOption": [
      "ALL",
      "ALTER",
      "DELETE",
      "DESCRIBE",
      "DROP",
      "INSERT"
    ]
  },
  {
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/
Admin"
    },
    "Resource": {
      "TableWithColumns": {
        "CatalogId": "123456789111",
        "DatabaseName": "customer",
        "Name": "customer_invoice",
        "ColumnWildcard": {}
      }
    },
    "Permissions": [
      "SELECT"
    ],
    "PermissionsWithGrantOption": [
      "SELECT"
    ]
  }
]
```

```

    },
    {
      "Principal": {
Admin"
        "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/
      },
      "Resource": {
        "DataCellsFilter": {
          "TableCatalogId": "123456789111",
          "DatabaseName": "customer",
          "TableName": "customer_invoice",
          "Name": "dl_us_customer"
        }
      },
      "Permissions": [
        "DESCRIBE",
        "SELECT",
        "DROP"
      ],
      "PermissionsWithGrantOption": []
    }
  ],
  "NextToken": "VyeUFjY291bnRQZXJtaXNzaW9ucyI6ZmFsc2V9"
}

```

Para obtener más información, consulte [Gestión de los permisos de Lake Formation](#) en la Guía para desarrolladores de AWS Lake Formation.

Ejemplo 3: Para recuperar la lista de los permisos principales de las etiquetas LF

En el siguiente `list-permissions` ejemplo, se enumeran los permisos de las etiquetas LF concedidos al director.

```

aws lakeformation list-permissions \
  --cli-input-json file://input.json

```

Contenidos de `input.json`:

```

{
  "CatalogId": "123456789111",
  "Resource": {
    "LFTag": {

```



```

    "CatalogId": "123456789111",
    "TagKey": "category",
    "TagValues": [
        "private"
    ]
  },
  "MaxResults": 10
}

```

**Salida:**

```

{
  "PrincipalResourcePermissions": [{
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
admin"
    },
    "Resource": {
      "LFTag": {
        "CatalogId": "123456789111",
        "TagKey": "category",
        "TagValues": [
          "*"
        ]
      }
    },
    "Permissions": [
      "DESCRIBE"
    ],
    "PermissionsWithGrantOption": [
      "DESCRIBE"
    ]
  },
  {
    "Principal": {
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
admin"
    },
    "Resource": {
      "LFTag": {
        "CatalogId": "123456789111",
        "TagKey": "category",

```

```

        "TagValues": [
            "*"
        ]
    },
    "Permissions": [
        "ASSOCIATE"
    ],
    "PermissionsWithGrantOption": [
        "ASSOCIATE"
    ]
}
],
"NextToken": "EJwY21GMGFxOxVJanA3SW50cm1pc3Npb25zIjpmYWxzZX0="
}

```

Para obtener más información, consulte [Gestión de los permisos de Lake Formation](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [ListPermissions](#) la Referencia de AWS CLI comandos.

## list-resources

En el siguiente ejemplo de código, se muestra cómo usar `list-resources`.

### AWS CLI

Para enumerar los recursos gestionados por Lake Formation

En el siguiente `list-resources` ejemplo, se enumeran los recursos que coinciden con la condición que administra Lake Formation.

```

aws lakeformation list-resources \
  --cli-input-json file://input.json

```

Contenidos de `input.json`:

```

{
  "FilterConditionList": [{
    "Field": "ROLE_ARN",

```

```
    "ComparisonOperator": "CONTAINS",
    "StringValueList": [
      "123456789111"
    ]
  }],
  "MaxResults": 10
}
```

Salida:

```
{
  "ResourceInfoList": [{
    "ResourceArn": "arn:aws:s3:::lf-data-lake-123456789111",
    "RoleArn": "arn:aws:iam::123456789111:role/LF-GlueServiceRole",
    "LastModified": "2022-07-21T02:12:46.669000+00:00"
  },
  {
    "ResourceArn": "arn:aws:s3:::lf-emr-test-123456789111",
    "RoleArn": "arn:aws:iam::123456789111:role/EMRIFS3Role",
    "LastModified": "2022-07-29T16:22:03.211000+00:00"
  }
  ]
}
```

Para obtener más información, consulte [Gestión de los permisos de Lake Formation](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [ListResources](#) la Referencia de AWS CLI comandos.

## list-transactions

En el siguiente ejemplo de código, se muestra cómo usar `list-transactions`.

### AWS CLI

Para enumerar todos los detalles de las transacciones

El siguiente `list-transactions` ejemplo devuelve metadatos sobre las transacciones y su estado.

```
aws lakeformation list-transactions \
```

```
--cli-input-json file://input.json
```

Contenidos de input.json:

```
{
  "CatalogId": "123456789111",
  "StatusFilter": "ALL",
  "MaxResults": 3
}
```

Salida:

```
{
  "Transactions": [{
    "TransactionId": "1234569f08804cb790d950d4d0fe485e",
    "TransactionStatus": "committed",
    "TransactionStartTime": "2022-08-10T14:32:29.220000+00:00",
    "TransactionEndTime": "2022-08-10T14:32:33.751000+00:00"
  },
  {
    "TransactionId": "12345972ca8347b89825e33c5774aec4",
    "TransactionStatus": "committed",
    "TransactionStartTime": "2022-08-10T14:29:04.046000+00:00",
    "TransactionEndTime": "2022-08-10T14:29:09.681000+00:00"
  },
  {
    "TransactionId": "12345daf6cb047dbba8ad9b0414613b2",
    "TransactionStatus": "committed",
    "TransactionStartTime": "2022-08-10T13:56:51.261000+00:00",
    "TransactionEndTime": "2022-08-10T13:56:51.547000+00:00"
  }
  ],
  "NextToken": "77X1ebypsI7os+X21hHsZLGNC DK3nNGpwRdFpicS0HgcX1/
QMoniUAKcpR3kj3ts3PVdMA=="
}
```

Para obtener más información, consulte [Leer y escribir en el lago de datos dentro de las transacciones](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [ListTransactions](#) la Referencia de AWS CLI comandos.

## put-data-lake-settings

En el siguiente ejemplo de código, se muestra cómo usar `put-data-lake-settings`.

### AWS CLI

Para establecer la configuración de un AWS lago de datos gestionado por Lake Formation

El siguiente `put-data-lake-settings` ejemplo establece la lista de administradores de lagos de datos y otras configuraciones de lagos de datos.

```
aws lakeformation put-data-lake-settings \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "DataLakeSettings": {  
    "DataLakeAdmins": [{  
      "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-  
admin"  
    }  
  ],  
  "CreateDatabaseDefaultPermissions": [],  
  "CreateTableDefaultPermissions": [],  
  "TrustedResourceOwners": [],  
  "AllowExternalDataFiltering": true,  
  "ExternalDataFilteringAllowList": [{  
    "DataLakePrincipalIdentifier": "123456789111"  
  }],  
  "AuthorizedSessionTagValueList": ["Amazon EMR"]  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cambiar la configuración de seguridad predeterminada de su lago de datos en la Guía para desarrolladores de AWS Lake Formation](#).

- Para API obtener más información, consulte [PutDataLakeSettings](#) la Referencia de AWS CLI comandos.

## register-resource

En el siguiente ejemplo de código, se muestra cómo usar `register-resource`.

### AWS CLI

Ejemplo 1: Para registrar el almacenamiento de un lago de datos mediante Service Linked Role

En el siguiente `register-resource` ejemplo, se registra el recurso gestionado por Lake Formation mediante el rol vinculado a un servicio.

```
aws lakeformation register-resource \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "ResourceArn": "arn:aws:s3:::lf-emr-athena-result-123",  
  "UseServiceLinkedRole": true  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir una ubicación de Amazon S3 a su lago de datos](#) en la Guía para desarrolladores de AWS Lake Formation.

Ejemplo 2: Para registrar el almacenamiento de un lago de datos mediante una función personalizada

En el siguiente `register-resource` ejemplo, se registra el recurso como gestionado por Lake Formation mediante un rol personalizado.

```
aws lakeformation register-resource \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "ResourceArn": "arn:aws:s3:::lf-emr-athena-result-123",  
  "UseServiceLinkedRole": false,  
}
```

```
"RoleArn": "arn:aws:iam::123456789111:role/LF-GlueServiceRole"
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir una ubicación de Amazon S3 a su lago de datos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [RegisterResource](#) la Referencia de AWS CLI comandos.

## **remove-lf-tags-from-resource**

En el siguiente ejemplo de código, se muestra cómo usar `remove-lf-tags-from-resource`.

### AWS CLI

Para eliminar la etiqueta LF de un recurso

En el siguiente `remove-lf-tags-from-resource` ejemplo, se elimina la asociación de la etiqueta LF con el recurso de la tabla.

```
aws lakeformation remove-lf-tags-from-resource \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "CatalogId": "123456789111",
  "Resource": {
    "Table": {
      "CatalogId": "123456789111",
      "DatabaseName": "tpc",
      "Name": "dl_tpc_promotion"
    }
  },
  "LFTags": [{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }
]
```

```
    ]
  }
}
```

Salida:

```
{
  "Failures": []
}
```

Para obtener más información, consulte [Asignación de etiquetas LF a los recursos del catálogo de datos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte la Referencia de [RemoveLfTagsFromResource](#) comandos AWS CLI .

## revoke-permissions

En el siguiente ejemplo de código, se muestra cómo usar `revoke-permissions`.

### AWS CLI

Para revocar los permisos sobre los recursos del director

En el siguiente `revoke-permissions` ejemplo, se revoca el acceso principal a una tabla específica de una base de datos determinada.

```
aws lakeformation revoke-permissions \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "CatalogId": "123456789111",
  "Principal": {
    "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
  },
  "Resource": {
    "Table": {
      "CatalogId": "123456789111",
      "DatabaseName": "tpc",
```



```
        "Name": "dl_tpc_promotion"
      }
    },
    "Permissions": [
      "ALL"
    ],
    "PermissionsWithGrantOption": []
  }
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Concesión y revocación de permisos sobre los recursos del catálogo de datos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [RevokePermissions](#) la Referencia de AWS CLI comandos.

## search-databases-by-lf-tags

En el siguiente ejemplo de código, se muestra cómo usar `search-databases-by-lf-tags`.

### AWS CLI

Para buscar en los recursos de la base de datos por LFTags

En el siguiente `search-databases-by-lf-tags` ejemplo, se buscan recursos de bases de datos que coincidan con LFTag la expresión.

```
aws lakeformation search-databases-by-lf-tags \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "MaxResults": 1,
  "CatalogId": "123456789111",
  "Expression": [{
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }]
}
```

```
}
```

Salida:

```
{
  "DatabaseList": [{
    "Database": {
      "CatalogId": "123456789111",
      "Name": "tpc"
    },
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }
  ]
}]
}
```

Para obtener más información, consulte [Visualización de los recursos a los que se asigna una etiqueta LF en la Guía para desarrolladores de AWS Lake Formation](#).

- Para API obtener más información, consulte la Referencia [SearchDatabasesByLfTags](#) de AWS CLI comandos.

## search-tables-by-lf-tags

En el siguiente ejemplo de código, se muestra cómo usar `search-tables-by-lf-tags`.

### AWS CLI

Para buscar recursos en una tabla por LFTags

En el siguiente `search-tables-by-lf-tags` ejemplo, se busca en los recursos de la tabla que coincidan con LFTag la expresión.

```
aws lakeformation search-tables-by-lf-tags \
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{
  "MaxResults": 2,
  "CatalogId": "123456789111",
  "Expression": [{
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }]
}
```

**Salida:**

```
{
  "NextToken": "c2VhcmNoQWxsVGFnc0luVGFibGVzIjpmYWxzZX0=",
  "TableList": [{
    "Table": {
      "CatalogId": "123456789111",
      "DatabaseName": "tpc",
      "Name": "dl_tpc_item"
    },
    "LFTagOnDatabase": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }
  ]},
  "LFTagsOnTable": [{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }
  ],
  "LFTagsOnColumns": [{
    "Name": "i_item_desc",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }
  ]
}
```

```
    ]],
  },
  {
    "Name": "i_container",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_wholesale_cost",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_manufact_id",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_brand_id",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_formulation",
```

```
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_current_price",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_size",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_rec_start_date",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_manufact",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
```

```
        "developer"
      ]
    ]
  },
  {
    "Name": "i_item_sk",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_manager_id",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_item_id",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
  {
    "Name": "i_class_id",
    "LFTags": [{
      "CatalogId": "123456789111",
      "TagKey": "usergroup",
      "TagValues": [
        "developer"
      ]
    }]
  },
},
```

```
{
  "Name": "i_class",
  "LFTags": [{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }]
},
{
  "Name": "i_category",
  "LFTags": [{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }]
},
{
  "Name": "i_category_id",
  "LFTags": [{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }]
},
{
  "Name": "i_brand",
  "LFTags": [{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
      "developer"
    ]
  }]
},
{
  "Name": "i_units",
  "LFTags": [{
    "CatalogId": "123456789111",
```

```
        "TagKey": "usergroup",
        "TagValues": [
            "developer"
        ]
    }
},
{
    "Name": "i_rec_end_date",
    "LFTags": [{
        "CatalogId": "123456789111",
        "TagKey": "usergroup",
        "TagValues": [
            "developer"
        ]
    }]
},
{
    "Name": "i_color",
    "LFTags": [{
        "CatalogId": "123456789111",
        "TagKey": "usergroup",
        "TagValues": [
            "developer"
        ]
    }]
},
{
    "Name": "i_product_name",
    "LFTags": [{
        "CatalogId": "123456789111",
        "TagKey": "usergroup",
        "TagValues": [
            "developer"
        ]
    }]
}
]
}]
}
```

Para obtener más información, consulte [Visualización de los recursos a los que se asigna una etiqueta LF en la Guía para desarrolladores de AWS Lake Formation](#).



- Para API obtener más información, consulte la Referencia [SearchTablesByLfTags](#) de AWS CLI comandos.

## start-query-planning

En el siguiente ejemplo de código, se muestra cómo usar `start-query-planning`.

### AWS CLI

Para procesar la declaración de consulta

El siguiente `start-query-planning` ejemplo envía una solicitud para procesar una declaración de consulta.

```
aws lakeformation start-query-planning \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "QueryPlanningContext": {  
    "CatalogId": "012345678901",  
    "DatabaseName": "tpc"  
  },  
  "QueryString": "select * from dl_tpc_household_demographics_gov where  
hd_income_band_sk=9"  
}
```

Salida:

```
{  
  "QueryId": "772a273f-4a62-4cda-8d98-69615ee8be9b"  
}
```

Para obtener más información, consulte [Leer y escribir en el lago de datos dentro de las transacciones](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [StartQueryPlanning](#) la Referencia de AWS CLI comandos.

## start-transaction

En el siguiente ejemplo de código, se muestra cómo usar `start-transaction`.

### AWS CLI

Para iniciar una nueva transacción

En el siguiente `start-transaction` ejemplo, se inicia una nueva transacción y se devuelve su identificador de transacción.

```
aws lakeformation start-transaction \  
  --transaction-type = 'READ_AND_WRITE'
```

Salida:

```
{  
  "TransactionId": "b014d972ca8347b89825e33c5774aec4"  
}
```

Para obtener más información, consulte [Leer y escribir en el lago de datos dentro de las transacciones](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [StartTransaction](#) la Referencia de AWS CLI comandos.

## update-lf-tag

En el siguiente ejemplo de código, se muestra cómo usar `update-lf-tag`.

### AWS CLI

Para actualizar la definición de etiqueta LF

El siguiente `update-lf-tag` ejemplo actualiza la definición de etiqueta LF.

```
aws lakeformation update-lf-tag \  
  --catalog-id '123456789111' \  
  --tag-key 'usergroup' \  
  --tag-values-to-add ['admin']
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de etiquetas LF para el control de acceso a los metadatos](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte la Referencia [UpdateLfTag](#) de AWS CLI comandos.

## update-table-objects

En el siguiente ejemplo de código, se muestra cómo usar `update-table-objects`.

### AWS CLI

Para modificar los objetos de una tabla gobernada

El siguiente `update-table-objects` ejemplo agrega los objetos S3 proporcionados a la tabla gobernada especificada.

```
aws lakeformation update-table-objects \  
  --cli-input-json file://input.json
```

Contenidos de `input.json`:

```
{  
  "CatalogId": "012345678901",  
  "DatabaseName": "tpc",  
  "TableName": "dl_tpc_household_demographics_gov",  
  "TransactionId": "12347a9f75424b9b915f6ff201d2a190",  
  "WriteOperations": [{  
    "AddObject": {  
      "Uri": "s3://lf-data-lake-012345678901/target/  
dl_tpc_household_demographics_gov/run-unnamed-1-part-block-0-r-00000-snappy-  
ff26b17504414fe88b302cd795eabd00.parquet",  
      "ETag": "1234ab1fc50a316b149b4e1f21a73800",  
      "Size": 42200  
    }  
  ]  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Leer y escribir en el lago de datos dentro de las transacciones](#) en la Guía para desarrolladores de AWS Lake Formation.

- Para API obtener más información, consulte [UpdateTableObjects](#) la Referencia de AWS CLI comandos.

## Ejemplos de Lambda que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Lambda.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **add-layer-version-permission**

En el siguiente ejemplo de código, se muestra cómo usar `add-layer-version-permission`.

AWS CLI

Para añadir permisos a una versión de capa

En el siguiente `add-layer-version-permission` ejemplo, se concede permiso a la cuenta especificada para utilizar la versión 1 de la capa `my-layer`.

```
aws lambda add-layer-version-permission \  
  --layer-name my-layer \  
  --statement-id xaccount \  
  --action lambda:GetLayerVersion \  
  --principal 123456789012 \  
  --version-number 1
```

Salida:

```
{
  "RevisionId": "35d87451-f796-4a3f-a618-95a3671b0a0c",
  "Statement":
  {
    "Sid":"xaccount",
    "Effect":"Allow",
    "Principal":{
      "AWS":"arn:aws:iam::210987654321:root"
    },
    "Action":"lambda:GetLayerVersion",
    "Resource":"arn:aws:lambda:us-east-2:123456789012:layer:my-layer:1"
  }
}
```

Para obtener más información, consulte [Capas AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [AddLayerVersionPermission](#) de AWS CLI comandos.

## add-permission

En el siguiente ejemplo de código, se muestra cómo usar `add-permission`.

### AWS CLI

Para añadir permisos a una función Lambda existente

En el siguiente `add-permission` ejemplo, se concede al SNS servicio de Amazon permiso para invocar una función denominada `my-function`.

```
aws lambda add-permission \
  --function-name my-function \
  --action lambda:InvokeFunction \
  --statement-id sns \
  --principal sns.amazonaws.com
```

Salida:

```
{
  "Statement":
```

```
{
  "Sid": "sns",
  "Effect": "Allow",
  "Principal": {
    "Service": "sns.amazonaws.com"
  },
  "Action": "lambda:InvokeFunction",
  "Resource": "arn:aws:lambda:us-east-2:123456789012:function:my-function"
}
```

Para obtener más información, consulte [Uso de políticas basadas en recursos para Lambda AWS en la Guía para desarrolladores](#) de AWS Lambda.

- Para obtener API más información, consulte la Referencia de comandos [AddPermission](#).AWS CLI

## create-alias

En el siguiente ejemplo de código, se muestra cómo usar `create-alias`.

### AWS CLI

Creación de un alias para una función de Lambda

En el siguiente ejemplo de `create-alias`, se crea un alias llamado LIVE que apunta a la versión 1 de la función de Lambda `my-function`.

```
aws lambda create-alias \
  --function-name my-function \
  --description "alias for live version of function" \
  --function-version 1 \
  --name LIVE
```

Salida:

```
{
  "FunctionVersion": "1",
  "Name": "LIVE",
  "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:LIVE",
  "RevisionId": "873282ed-4cd3-4dc8-a069-d0c647e470c6",
```

```
"Description": "alias for live version of function"
}
```

Para obtener más información, consulte [Configuración de alias de funciones de AWS Lambda en la Guía para desarrolladores](#) de Lambda AWS .

- Para API obtener más información, consulte la Referencia de [CreateAlias](#) comandos AWS CLI .

## create-event-source-mapping

En el siguiente ejemplo de código, se muestra cómo usar `create-event-source-mapping`.

### AWS CLI

Para crear un mapeo entre una fuente de eventos y una función AWS Lambda

El siguiente `create-event-source-mapping` ejemplo crea un mapeo entre una SQS cola y la función `my-function` Lambda.

```
aws lambda create-event-source-mapping \
  --function-name my-function \
  --batch-size 5 \
  --event-source-arn arn:aws:sqs:us-west-2:123456789012:mySQSqueue
```

Salida:

```
{
  "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
  "StateTransitionReason": "USER_INITIATED",
  "LastModified": 1569284520.333,
  "BatchSize": 5,
  "State": "Creating",
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
  "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"
}
```

Para obtener más información, consulte [Mapeo de fuentes de eventos de AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [CreateEventSourceMapping](#) de AWS CLI comandos.

## create-function

En el siguiente ejemplo de código, se muestra cómo usar `create-function`.

### AWS CLI

Para crear una función de Lambda

El siguiente ejemplo de `create-function` crea una función de Lambda con el nombre `my-function`.

```
aws lambda create-function \  
  --function-name my-function \  
  --runtime nodejs18.x \  
  --zip-file fileb://my-function.zip \  
  --handler my-function.handler \  
  --role arn:aws:iam::123456789012:role/service-role/MyTestFunction-role-tges6bf4
```

Contenidos de `my-function.zip`:

```
This file is a deployment package that contains your function code and any dependencies.
```

Salida:

```
{  
  "TracingConfig": {  
    "Mode": "PassThrough"  
  },  
  "CodeSha256": "PFn4S+er27qk+UuZSTKEQfNKG/XNn7QJs90mJgq6oH8=",  
  "FunctionName": "my-function",  
  "CodeSize": 308,  
  "RevisionId": "873282ed-4cd3-4dc8-a069-d0c647e470c6",  
  "MemorySize": 128,  
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",  
  "Version": "$LATEST",  
  "Role": "arn:aws:iam::123456789012:role/service-role/MyTestFunction-role-zgur6bf4",  
  "Timeout": 3,  
  "LastModified": "2023-10-14T22:26:11.234+0000",  
  "Handler": "my-function.handler",  
  "Runtime": "nodejs18.x",  
  "Description": ""
```



```
}
```

Para obtener más información, consulte [Configuración las funciones de Lambda de AWS](#) en la Guía para desarrolladores de Lambda de AWS .

- Para API obtener más información, consulte [CreateFunction](#) la Referencia de AWS CLI comandos.

## delete-alias

En el siguiente ejemplo de código, se muestra cómo usar `delete-alias`.

### AWS CLI

Eliminación de un alias de función de Lambda

En el siguiente ejemplo de `delete-alias`, se elimina el alias nombrado `LIVE` de la función de Lambda `my-function`.

```
aws lambda delete-alias \  
  --function-name my-function \  
  --name LIVE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de alias de funciones de AWS Lambda en la Guía para desarrolladores](#) de Lambda AWS .

- Para API obtener más información, consulte la Referencia de [DeleteAlias](#) comandos AWS CLI .

## delete-event-source-mapping

En el siguiente ejemplo de código, se muestra cómo usar `delete-event-source-mapping`.

### AWS CLI

Para eliminar el mapeo entre un origen de eventos y una función AWS Lambda

En el siguiente `delete-event-source-mapping` ejemplo, se elimina la asignación entre una SQS cola y la función Lambda `my-function`.

```
aws lambda delete-event-source-mapping \  
  --function-name my-function \  
  --event-source-arn arn:aws:sqs:us-east-1:123456789012:my-queue
```

```
--uuid a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Salida:

```
{
  "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
  "StateTransitionReason": "USER_INITIATED",
  "LastModified": 1569285870.271,
  "BatchSize": 5,
  "State": "Deleting",
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
  "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"
}
```

Para obtener más información, consulte [Mapeo de fuentes de eventos de AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [DeleteEventSourceMapping](#) de AWS CLI comandos.

## **delete-function-concurrency**

En el siguiente ejemplo de código, se muestra cómo usar `delete-function-concurrency`.

AWS CLI

Eliminación del límite de ejecución simultánea reservado de una función

En el siguiente ejemplo de `delete-function-concurrency`, se elimina el límite de ejecución simultánea reservado de la función `my-function`.

```
aws lambda delete-function-concurrency \  
  --function-name my-function
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Reserva de simultaneidad para una función de Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte [DeleteFunctionConcurrency](#) la Referencia de AWS CLI comandos.

## delete-function-event-invoke-config

En el siguiente ejemplo de código, se muestra cómo usar `delete-function-event-invoke-config`.

### AWS CLI

Para eliminar una configuración de invocación asíncrona

En el siguiente `delete-function-event-invoke-config` ejemplo, se elimina la configuración de invocación asíncrona del alias de la función especificada. GREEN

```
aws lambda delete-function-event-invoke-config --function-name my-function:GREEN
```

- Para obtener API más información, consulte la Referencia de comandos.

[DeleteFunctionEventInvokeConfig](#) AWS CLI

## delete-function

En el siguiente ejemplo de código, se muestra cómo usar `delete-function`.

### AWS CLI

Ejemplo 1: eliminar una función de Lambda con el nombre de la función

En el siguiente ejemplo de `delete-function`, se elimina la función de Lambda denominada `my-function` especificando el nombre de la función.

```
aws lambda delete-function \  
  --function-name my-function
```

Este comando no genera ninguna salida.

Ejemplo 2: Para eliminar una función Lambda por función ARN

En el siguiente `delete-function` ejemplo, se elimina la función Lambda `my-function` nombrada especificando la función. ARN

```
aws lambda delete-function \  
  --function-arn arn:aws:lambda:us-east-1:123456789012:function:my-function
```

```
--function-name arn:aws:lambda:us-west-2:123456789012:function:my-function
```

Este comando no genera ninguna salida.

Ejemplo 3: Para eliminar una función Lambda mediante una función parcial ARN

En el siguiente `delete-function` ejemplo, se elimina la función Lambda `my-function` denominada especificando el parcial de la función. ARN

```
aws lambda delete-function \  
  --function-name 123456789012:function:my-function
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración las funciones de Lambda de AWS](#) en la Guía para desarrolladores de Lambda de AWS .

- Para API obtener más información, consulte la Referencia [DeleteFunction](#) de AWS CLI comandos.

## `delete-layer-version`

En el siguiente ejemplo de código, se muestra cómo usar `delete-layer-version`.

AWS CLI

Para eliminar una versión de una capa Lambda

En el siguiente `delete-layer-version` ejemplo, se elimina la versión 2 de la capa denominada. `my-layer`

```
aws lambda delete-layer-version \  
  --layer-name my-layer \  
  --version-number 2
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Capas AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [DeleteLayerVersion](#) de AWS CLI comandos.

## delete-provisioned-concurrency-config

En el siguiente ejemplo de código, se muestra cómo usar `delete-provisioned-concurrency-config`.

### AWS CLI

Eliminación de la configuración de simultaneidad aprovisionada

En el siguiente ejemplo de `delete-provisioned-concurrency-config`, se elimina la configuración de simultaneidad aprovisionada para el alias GREEN de la función especificada.

```
aws lambda delete-provisioned-concurrency-config \  
  --function-name my-function \  
  --qualifier GREEN
```

- Para API obtener más información, consulte [DeleteProvisionedConcurrencyConfig](#) la Referencia de AWS CLI comandos.

## get-account-settings

En el siguiente ejemplo de código, se muestra cómo usar `get-account-settings`.

### AWS CLI

Para recuperar los detalles de tu cuenta en una AWS región

En el siguiente ejemplo de `get-account-settings`, se muestran los límites de Lambda y la información de uso de su cuenta.

```
aws lambda get-account-settings
```

Salida:

```
{  
  "AccountLimit": {  
    "CodeSizeUnzipped": 262144000,  
    "UnreservedConcurrentExecutions": 1000,  
    "ConcurrentExecutions": 1000,  
    "CodeSizeZipped": 52428800,  
    "TotalCodeSize": 80530636800
```

```
  },
  "AccountUsage": {
    "FunctionCount": 4,
    "TotalCodeSize": 9426
  }
}
```

Para obtener más información, consulte [Límites de AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte [GetAccountSettings](#) la Referencia de AWS CLI comandos.

## get-alias

En el siguiente ejemplo de código, se muestra cómo usar `get-alias`.

### AWS CLI

Recuperación de detalles sobre el alias de una función

En el siguiente ejemplo de `get-alias`, se muestran detalles del alias nombrado LIVE de la función de Lambda `my-function`.

```
aws lambda get-alias \
  --function-name my-function \
  --name LIVE
```

Salida:

```
{
  "FunctionVersion": "3",
  "Name": "LIVE",
  "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:LIVE",
  "RevisionId": "594f41fb-b85f-4c20-95c7-6ca5f2a92c93",
  "Description": "alias for live version of function"
}
```

Para obtener más información, consulte [Configuración de alias de funciones de AWS Lambda en la Guía para desarrolladores](#) de Lambda AWS .

- Para API obtener más información, consulte la Referencia de [GetAlias](#) comandos AWS CLI .

## get-event-source-mapping

En el siguiente ejemplo de código, se muestra cómo usar `get-event-source-mapping`.

### AWS CLI

Para recuperar detalles sobre el mapeo de la fuente de un evento

En el siguiente `get-event-source-mapping` ejemplo, se muestran los detalles del mapeo entre una SQS cola y la función `my-function` Lambda.

```
aws lambda get-event-source-mapping \  
  --uuid "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
```

Salida:

```
{  
  "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
  "StateTransitionReason": "USER_INITIATED",  
  "LastModified": 1569284520.333,  
  "BatchSize": 5,  
  "State": "Enabled",  
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",  
  "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"  
}
```

Para obtener más información, consulte [Mapeo de fuentes de eventos de AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [GetEventSourceMapping](#) de AWS CLI comandos.

## get-function-concurrency

En el siguiente ejemplo de código, se muestra cómo usar `get-function-concurrency`.

### AWS CLI

Visualización de la configuración de simultaneidad reservada para una función

En el siguiente ejemplo de `get-function-concurrency`, se recupera la configuración de simultaneidad reservada para la función especificada.

```
aws lambda get-function-concurrency \  
  --function-name my-function
```

Salida:

```
{  
  "ReservedConcurrentExecutions": 250  
}
```

- Para API obtener más información, consulte [GetFunctionConcurrency](#) la Referencia de AWS CLI comandos.

## get-function-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-function-configuration`.

AWS CLI

Recuperación de la configuración específica de la versión de una función de Lambda

En el siguiente ejemplo de `get-function-configuration`, se muestran los ajustes de la versión 2 de la función `my-function`.

```
aws lambda get-function-configuration \  
  --function-name my-function:2
```

Salida:

```
{  
  "FunctionName": "my-function",  
  "LastModified": "2019-09-26T20:28:40.438+0000",  
  "RevisionId": "e52502d4-9320-4688-9cd6-152a6ab7490d",  
  "MemorySize": 256,  
  "Version": "2",  
  "Role": "arn:aws:iam::123456789012:role/service-role/my-function-role-uy3l9qq",  
  "Timeout": 3,  
  "Runtime": "nodejs10.x",  
  "TracingConfig": {  
    "Mode": "PassThrough"  
  },  
  "CodeSha256": "5tT2qgzYUHaqWR716pZ2dpkn/0J1FrzJm1KidWoaCgk=",
```



```
"Description": "",
  "VpcConfig": {
    "SubnetIds": [],
    "VpcId": "",
    "SecurityGroupIds": []
  },
  "CodeSize": 304,
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:2",
  "Handler": "index.handler"
}
```

Para obtener más información, consulte [Configuración las funciones de Lambda de AWS](#) en la Guía para desarrolladores de Lambda de AWS .

- Para API obtener más información, consulte [GetFunctionConfiguration](#) la Referencia de AWS CLI comandos.

## get-function-event-invoke-config

En el siguiente ejemplo de código, se muestra cómo usar `get-function-event-invoke-config`.

### AWS CLI

Para ver una configuración de invocación asíncrona

El siguiente `get-function-event-invoke-config` ejemplo recupera la configuración de invocación asíncrona para el alias de la función especificada. BLUE

```
aws lambda get-function-event-invoke-config \
  --function-name my-function:BLUE
```

Salida:

```
{
  "LastModified": 1577824396.653,
  "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:BLUE",
  "MaximumRetryAttempts": 0,
  "MaximumEventAgeInSeconds": 3600,
  "DestinationConfig": {
    "OnSuccess": {},
  }
}
```

```
    "OnFailure": {
      "Destination": "arn:aws:sqs:us-east-2:123456789012:failed-invocations"
    }
  }
}
```

- Para obtener API más información, consulte la Referencia de comandos.

[GetFunctionEventInvokeConfig](#) AWS CLI

## get-function

En el siguiente ejemplo de código, se muestra cómo usar `get-function`.

### AWS CLI

Cómo recuperar información sobre una función

En el siguiente ejemplo de `get-function` se muestra información sobre la función `my-function`.

```
aws lambda get-function \
  --function-name my-function
```

Salida:

```
{
  "Concurrency": {
    "ReservedConcurrentExecutions": 100
  },
  "Code": {
    "RepositoryType": "S3",
    "Location": "https://awslambda-us-west-2-tasks.s3.us-west-2.amazonaws.com/snapshots/123456789012/my-function..."
  },
  "Configuration": {
    "TracingConfig": {
      "Mode": "PassThrough"
    },
    "Version": "$LATEST",
    "CodeSha256": "5tT2qgzYUHoqwR616pZ2dpkn/0J1FrzJm1KidWaaCgk=",
    "FunctionName": "my-function",
    "VpcConfig": {
```

```

        "SubnetIds": [],
        "VpcId": "",
        "SecurityGroupIds": []
    },
    "MemorySize": 128,
    "RevisionId": "28f0fb31-5c5c-43d3-8955-03e76c5c1075",
    "CodeSize": 304,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
    "Handler": "index.handler",
    "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-role-uy3l9qq",
    "Timeout": 3,
    "LastModified": "2019-09-24T18:20:35.054+0000",
    "Runtime": "nodejs10.x",
    "Description": ""
}
}

```

Para obtener más información, consulte [Configuración las funciones de Lambda de AWS](#) en la Guía para desarrolladores de Lambda de AWS .

- Para API obtener más información, consulte [GetFunction](#) la Referencia de AWS CLI comandos.

## get-layer-version-by-arn

En el siguiente ejemplo de código, se muestra cómo usar `get-layer-version-by-arn`.

### AWS CLI

Para recuperar información sobre una versión de capa Lambda

En el siguiente `get-layer-version-by-arn` ejemplo, se muestra información sobre la versión de la capa con el nombre de recurso de Amazon (ARN) especificado.

```

aws lambda get-layer-version-by-arn \
  --arn "arn:aws:lambda:us-west-2:123456789012:layer:AWSLambda-Python311-SciPy1x:2"

```

Salida:

```

{
  "LayerVersionArn": "arn:aws:lambda:us-west-2:123456789012:layer:AWSLambda-Python311-SciPy1x:2",

```

```

    "Description": "AWS Lambda SciPy layer for Python 3.11 (scipy-1.1.0,
numpy-1.15.4) https://github.com/scipy/scipy/releases/tag/v1.1.0 https://
github.com/numpy/numpy/releases/tag/v1.15.4",
    "CreateDate": "2023-10-12T10:09:38.398+0000",
    "LayerArn": "arn:aws:lambda:us-west-2:123456789012:layer:AWSLambda-Python311-
SciPy1x",
    "Content": {
        "CodeSize": 41784542,
        "CodeSha256": "GGmv8ocUw4c1y0T8HL0Vx/f5V4RmSCGNjDIslY4VskM=",
        "Location": "https://awslambda-us-west-2-layers.s3.us-west-2.amazonaws.com/
snapshots/123456789012/..."
    },
    "Version": 2,
    "CompatibleRuntimes": [
        "python3.11"
    ],
    "LicenseInfo": "SciPy: https://github.com/scipy/scipy/blob/main/LICENSE.txt,
NumPy: https://github.com/numpy/numpy/blob/main/LICENSE.txt"
}

```

Para obtener más información, consulte [Capas AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [GetLayerVersionByArn](#) de AWS CLI comandos.

## get-layer-version-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-layer-version-policy`.

### AWS CLI

Para recuperar la política de permisos de una versión de capa Lambda

En el siguiente `get-layer-version-policy` ejemplo, se muestra información sobre la política sobre la versión 1 de la capa denominada `my-layer`.

```

aws lambda get-layer-version-policy \
  --layer-name my-layer \
  --version-number 1

```

Salida:

```
{
  "Policy": {
    "Version": "2012-10-17",
    "Id": "default",
    "Statement": [
      {
        "Sid": "xaccount",
        "Effect": "Allow",
        "Principal": {"AWS": "arn:aws:iam::123456789012:root"},
        "Action": "lambda:GetLayerVersion",
        "Resource": "arn:aws:lambda:us-west-2:123456789012:layer:my-layer:1"
      }
    ]
  },
  "RevisionId": "c68f21d2-cbf0-4026-90f6-1375ee465cd0"
}
```

Para obtener más información, consulte [Capas AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [GetLayerVersionPolicy](#) de AWS CLI comandos.

## get-layer-version

En el siguiente ejemplo de código, se muestra cómo usar `get-layer-version`.

### AWS CLI

Para recuperar información sobre una versión de capa Lambda

En el siguiente `get-layer-version` ejemplo, se muestra información sobre la versión 1 de la capa denominada `my-layer`.

```
aws lambda get-layer-version \
  --layer-name my-layer \
  --version-number 1
```

Salida:

```
{
```

```

"Content": {
  "Location": "https://awslambda-us-east-2-layers.s3.us-east-2.amazonaws.com/
snapshots/123456789012/my-layer-4aaa2fbb-ff77-4b0a-ad92-5b78a716a96a?
versionId=27iWyA73cCAYqyH...",
  "CodeSha256": "tv9jJ0+rPbXUUXuRKi7CwHzKtLDkDRJLB3cC3Z/ouXo=",
  "CodeSize": 169
},
"LayerArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-layer",
"LayerVersionArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-layer:1",
"Description": "My Python layer",
"CreateDate": "2018-11-14T23:03:52.894+0000",
"Version": 1,
"LicenseInfo": "MIT",
"CompatibleRuntimes": [
  "python3.10",
  "python3.11"
]
}

```

Para obtener más información, consulte [Capas AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [GetLayerVersion](#) de AWS CLI comandos.

## get-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-policy`.

### AWS CLI

Para recuperar la IAM política basada en recursos de una función, versión o alias

En el siguiente ejemplo de `get-policy` se muestra información de la política sobre la función de Lambda `my-function`.

```

aws lambda get-policy \
  --function-name my-function

```

Salida:

```

{
  "Policy": {

```

```

    "Version": "2012-10-17",
    "Id": "default",
    "Statement": [
      {
        "Sid": "iot-events",
        "Effect": "Allow",
        "Principal": {"Service": "iotevents.amazonaws.com"},
        "Action": "lambda:InvokeFunction",
        "Resource": "arn:aws:lambda:us-west-2:123456789012:function:my-
function"
      }
    ],
    "RevisionId": "93017fc9-59cb-41dc-901b-4845ce4bf668"
  }

```

Para obtener más información, consulte [Uso de políticas basadas en recursos para Lambda AWS en la Guía para desarrolladores](#) de AWS Lambda.

- Para obtener API más información, consulte la Referencia de comandos [GetPolicy](#).AWS CLI

## get-provisioned-concurrency-config

En el siguiente ejemplo de código, se muestra cómo usar `get-provisioned-concurrency-config`.

### AWS CLI

Visualización de una configuración de simultaneidad aprovisionada

En el siguiente ejemplo de `get-provisioned-concurrency-config` se muestran detalles de la configuración de simultaneidad aprovisionada para el alias BLUE de la función especificada.

```

aws lambda get-provisioned-concurrency-config \
  --function-name my-function \
  --qualifier BLUE

```

Salida:

```

{
  "RequestedProvisionedConcurrentExecutions": 100,
  "AvailableProvisionedConcurrentExecutions": 100,

```

```
"AllocatedProvisionedConcurrentExecutions": 100,  
"Status": "READY",  
"LastModified": "2019-12-31T20:28:49+0000"  
}
```

- Para API obtener más información, consulte [GetProvisionedConcurrencyConfig](#) la Referencia de AWS CLI comandos.

## invoke

En el siguiente ejemplo de código, se muestra cómo usar `invoke`.

### AWS CLI

Ejemplo 1: invocar una función de Lambda de forma sincrónica

En el siguiente ejemplo de `invoke`, se invoca la función `my-function` de forma sincrónica. La `cli-binary-format` opción es obligatoria si utiliza la AWS CLI versión 2. Para obtener más información, consulte las [opciones de línea de comandos globales AWS CLI compatibles](#) en la Guía del usuario de la interfaz de línea de AWS comandos.

```
aws lambda invoke \  
  --function-name my-function \  
  --cli-binary-format raw-in-base64-out \  
  --payload '{ "name": "Bob" }' \  
  response.json
```

Salida:

```
{  
  "ExecutedVersion": "$LATEST",  
  "StatusCode": 200  
}
```

Para obtener más información, consulte [Invocación sincrónica](#) en la Guía para desarrolladores de Lambda de AWS .

Ejemplo 2: invocar una función de Lambda de forma asíncrona

En el siguiente ejemplo de `invoke`, se invoca la función `my-function` de forma asíncrona. La `cli-binary-format` opción es obligatoria si utiliza la AWS CLI versión 2. Para obtener más



información, consulte las [opciones de línea de comandos globales AWS CLI compatibles](#) en la Guía del usuario de la interfaz de línea de AWS comandos.

```
aws lambda invoke \  
  --function-name my-function \  
  --invocation-type Event \  
  --cli-binary-format raw-in-base64-out \  
  --payload '{ "name": "Bob" }' \  
  response.json
```

Salida:

```
{  
  "StatusCode": 202  
}
```

Para obtener más información, consulte [Invocación asíncrona](#) en la Guía para desarrolladores de Lambda de AWS .

- Para API obtener más información, consulte [Invoke](#) in AWS CLI Command Reference.

## list-alias

En el siguiente ejemplo de código, se muestra cómo usar `list-alias`.

### AWS CLI

Para recuperar la lista de alias de una función Lambda

En el siguiente `list-alias` ejemplo, se muestra una lista de los alias de la función `my-function` Lambda.

```
aws lambda list-alias \  
  --function-name my-function
```

Salida:

```
{  
  "Aliases": [  
    {  
      "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-  
function:BETA",
```

```

    "RevisionId": "a410117f-ab16-494e-8035-7e204bb7933b",
    "FunctionVersion": "2",
    "Name": "BETA",
    "Description": "alias for beta version of function"
  },
  {
    "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function:LIVE",
    "RevisionId": "21d40116-f8b1-40ba-9360-3ea284da1bb5",
    "FunctionVersion": "1",
    "Name": "LIVE",
    "Description": "alias for live version of function"
  }
]
}

```

Para obtener más información, consulte [Configuración de alias de funciones de AWS Lambda en la Guía para desarrolladores](#) de Lambda AWS .

- Para API obtener más información, consulte la Referencia de [ListAliases](#) comandos AWS CLI .

## list-event-source-mappings

En el siguiente ejemplo de código, se muestra cómo usar `list-event-source-mappings`.

### AWS CLI

Para enumerar las asignaciones de fuentes de eventos de una función

En el siguiente `list-event-source-mappings` ejemplo, se muestra una lista de las asignaciones de fuentes de eventos para la función `Lambdamy-function`.

```
aws lambda list-event-source-mappings \
  --function-name my-function
```

Salida:

```

{
  "EventSourceMappings": [
    {
      "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
      "StateTransitionReason": "USER_INITIATED",
      "LastModified": 1569284520.333,

```

```
        "BatchSize": 5,
        "State": "Enabled",
        "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function",
        "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"
    }
]
}
```

Para obtener más información, consulte [Mapeo de fuentes de eventos de AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [ListEventSourceMappings](#) de AWS CLI comandos.

## list-function-event-invoke-configs

En el siguiente ejemplo de código, se muestra cómo usar `list-function-event-invoke-configs`.

### AWS CLI

Para ver una lista de configuraciones de invocación asíncrona

En el siguiente `list-function-event-invoke-configs` ejemplo, se enumeran las configuraciones de invocación asíncrona de la función especificada.

```
aws lambda list-function-event-invoke-configs \
  --function-name my-function
```

Salida:

```
{
  "FunctionEventInvokeConfigs": [
    {
      "LastModified": 1577824406.719,
      "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:GREEN",
      "MaximumRetryAttempts": 2,
      "MaximumEventAgeInSeconds": 1800
    },
    {
```

```

        "LastModified": 1577824396.653,
        "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:BLUE",
        "MaximumRetryAttempts": 0,
        "MaximumEventAgeInSeconds": 3600
    }
]
}

```

- Para obtener API más información, consulte la Referencia de comandos [ListFunctionEventInvokeConfigs](#).AWS CLI

## list-functions

En el siguiente ejemplo de código, se muestra cómo usar `list-functions`.

### AWS CLI

Cómo recuperar una lista de funciones de Lambda

En el siguiente ejemplo de `list-functions`, se muestra una lista de todas las funciones para el usuario actual:

```
aws lambda list-functions
```

Salida:

```

{
  "Functions": [
    {
      "TracingConfig": {
        "Mode": "PassThrough"
      },
      "Version": "$LATEST",
      "CodeSha256": "dBG9m8SGdmlEjw/JYXlhhvCrAv5TxvXsbL/RMr0fT/I=",
      "FunctionName": "helloworld",
      "MemorySize": 128,
      "RevisionId": "1718e831-badf-4253-9518-d0644210af7b",
      "CodeSize": 294,
      "FunctionArn": "arn:aws:lambda:us-
west-2:123456789012:function:helloworld",
      "Handler": "helloworld.handler",

```

```
    "Role": "arn:aws:iam::123456789012:role/service-role/MyTestFunction-
role-zgur6bf4",
    "Timeout": 3,
    "LastModified": "2023-09-23T18:32:33.857+0000",
    "Runtime": "nodejs18.x",
    "Description": ""
  },
  {
    "TracingConfig": {
      "Mode": "PassThrough"
    },
    "Version": "$LATEST",
    "CodeSha256": "sU0cJ2/h0ZevwV/1TxCuQqK3gDZP3i8gUoqUUVRmY6E=",
    "FunctionName": "my-function",
    "VpcConfig": {
      "SubnetIds": [],
      "VpcId": "",
      "SecurityGroupIds": []
    },
    "MemorySize": 256,
    "RevisionId": "93017fc9-59cb-41dc-901b-4845ce4bf668",
    "CodeSize": 266,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function",
    "Handler": "index.handler",
    "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-
role-uy3l9qqq",
    "Timeout": 3,
    "LastModified": "2023-10-01T16:47:28.490+0000",
    "Runtime": "nodejs18.x",
    "Description": ""
  },
  {
    "Layers": [
      {
        "CodeSize": 41784542,
        "Arn": "arn:aws:lambda:us-west-2:420165488524:layer:AWSLambda-
Python37-SciPy1x:2"
      },
      {
        "CodeSize": 4121,
        "Arn": "arn:aws:lambda:us-
west-2:123456789012:layer:pythonLayer:1"
      }
    ]
  }
}
```

```
    ],
    "TracingConfig": {
        "Mode": "PassThrough"
    },
    "Version": "$LATEST",
    "CodeSha256": "ZQukCqxtkqFgyF2cU41Avj99TKQ/hNihPtDtRcc08mI=",
    "FunctionName": "my-python-function",
    "VpcConfig": {
        "SubnetIds": [],
        "VpcId": "",
        "SecurityGroupIds": []
    },
    "MemorySize": 128,
    "RevisionId": "80b4eabc-acf7-4ea8-919a-e874c213707d",
    "CodeSize": 299,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
python-function",
    "Handler": "lambda_function.lambda_handler",
    "Role": "arn:aws:iam::123456789012:role/service-role/my-python-function-
role-z5g7dr6n",
    "Timeout": 3,
    "LastModified": "2023-10-01T19:40:41.643+0000",
    "Runtime": "python3.11",
    "Description": ""
    }
  ]
}
```

Para obtener más información, consulte [Configuración las funciones de Lambda de AWS](#) en la Guía para desarrolladores de Lambda de AWS .

- Para API obtener más información, consulte [ListFunctions](#) la Referencia de AWS CLI comandos.

## **list-layer-versions**

En el siguiente ejemplo de código, se muestra cómo usar `list-layer-versions`.

### AWS CLI

Para enumerar las versiones de una capa AWS Lambda

En el siguiente `list-layer-versions` ejemplo, se muestra información sobre las versiones de la capa denominada `my-layer`.

```
aws lambda list-layer-versions \  
  --layer-name my-layer
```

Salida:

```
{  
  "Layers": [  
    {  
      "LayerVersionArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-  
layer:2",  
      "Version": 2,  
      "Description": "My layer",  
      "CreateDate": "2023-11-15T00:37:46.592+0000",  
      "CompatibleRuntimes": [  
        "python3.10",  
        "python3.11"  
      ]  
    }  
  ]  
}
```

Para obtener más información, consulte [Capas AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [ListLayerVersions](#) de AWS CLI comandos.

## **list-layers**

En el siguiente ejemplo de código, se muestra cómo usar `list-layers`.

AWS CLI

Para ver una lista de las capas que son compatibles con el tiempo de ejecución de la función

El siguiente `list-layers` ejemplo muestra información sobre las capas que son compatibles con el motor de ejecución de Python 3.11.

```
aws lambda list-layers \  
  --compatible-runtimes python3.11
```

```
--compatible-runtime python3.11
```

Salida:

```
{
  "Layers": [
    {
      "LayerName": "my-layer",
      "LayerArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-layer",
      "LatestMatchingVersion": {
        "LayerVersionArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-layer:2",
        "Version": 2,
        "Description": "My layer",
        "CreateDate": "2023-11-15T00:37:46.592+0000",
        "CompatibleRuntimes": [
          "python3.10",
          "python3.11"
        ]
      }
    }
  ]
}
```

Para obtener más información, consulte [Capas AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [ListLayers](#) de AWS CLI comandos.

## list-provisioned-concurrency-configs

En el siguiente ejemplo de código, se muestra cómo usar `list-provisioned-concurrency-configs`.

AWS CLI

Obtención de una lista de configuraciones de simultaneidad aprovisionada

En el siguiente ejemplo `list-provisioned-concurrency-configs`, se enumeran las configuraciones de simultaneidad aprovisionadas para la función especificada.

```
aws lambda list-provisioned-concurrency-configs \
```



```
--function-name my-function
```

Salida:

```
{
  "ProvisionedConcurrencyConfigs": [
    {
      "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:GREEN",
      "RequestedProvisionedConcurrentExecutions": 100,
      "AvailableProvisionedConcurrentExecutions": 100,
      "AllocatedProvisionedConcurrentExecutions": 100,
      "Status": "READY",
      "LastModified": "2019-12-31T20:29:00+0000"
    },
    {
      "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:BLUE",
      "RequestedProvisionedConcurrentExecutions": 100,
      "AvailableProvisionedConcurrentExecutions": 100,
      "AllocatedProvisionedConcurrentExecutions": 100,
      "Status": "READY",
      "LastModified": "2019-12-31T20:28:49+0000"
    }
  ]
}
```

- Para API obtener más información, consulte [ListProvisionedConcurrencyConfigs](#) la Referencia de AWS CLI comandos.

## list-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-tags`.

AWS CLI

Recuperación de la lista de etiquetas para una función de Lambda

En el siguiente ejemplo de `list-tags`, se muestran las etiquetas asociadas a la función de Lambda `my-function`.

```
aws lambda list-tags \
```

```
--resource arn:aws:lambda:us-west-2:123456789012:function:my-function
```

Salida:

```
{
  "Tags": {
    "Category": "Web Tools",
    "Department": "Sales"
  }
}
```

Para obtener más información, consulte [Etiquetas de funciones de Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte [ListTags](#) la Referencia de AWS CLI comandos.

## list-versions-by-function

En el siguiente ejemplo de código, se muestra cómo usar `list-versions-by-function`.

AWS CLI

Recuperación de la lista de versiones de una función

En el siguiente ejemplo de `list-versions-by-function`, se muestra la lista de versiones de la función de Lambda `my-function`.

```
aws lambda list-versions-by-function \
  --function-name my-function
```

Salida:

```
{
  "Versions": [
    {
      "TracingConfig": {
        "Mode": "PassThrough"
      },
      "Version": "$LATEST",
      "CodeSha256": "sU0cJ2/h0ZevwV/1TxCuQqK3gDZP3i8gUoqUUVRmY6E=",
      "FunctionName": "my-function",
      "VpcConfig": {
```

```
        "SubnetIds": [],
        "VpcId": "",
        "SecurityGroupIds": []
    },
    "MemorySize": 256,
    "RevisionId": "93017fc9-59cb-41dc-901b-4845ce4bf668",
    "CodeSize": 266,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function:$LATEST",
    "Handler": "index.handler",
    "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-
role-uy3l9qyq",
    "Timeout": 3,
    "LastModified": "2019-10-01T16:47:28.490+0000",
    "Runtime": "nodejs10.x",
    "Description": ""
},
{
    "TracingConfig": {
        "Mode": "PassThrough"
    },
    "Version": "1",
    "CodeSha256": "5tT2qgzYUHoqwR616pZ2dpkn/0J1FrzJmlKidWaaCgk=",
    "FunctionName": "my-function",
    "VpcConfig": {
        "SubnetIds": [],
        "VpcId": "",
        "SecurityGroupIds": []
    },
    "MemorySize": 256,
    "RevisionId": "949c8914-012e-4795-998c-e467121951b1",
    "CodeSize": 304,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function:1",
    "Handler": "index.handler",
    "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-
role-uy3l9qyq",
    "Timeout": 3,
    "LastModified": "2019-09-26T20:28:40.438+0000",
    "Runtime": "nodejs10.x",
    "Description": "new version"
},
{
    "TracingConfig": {
```

```

        "Mode": "PassThrough"
    },
    "Version": "2",
    "CodeSha256": "sU0cJ2/h0ZevwV/1TxCuQqK3gDZP3i8gUoqUUVRmY6E=",
    "FunctionName": "my-function",
    "VpcConfig": {
        "SubnetIds": [],
        "VpcId": "",
        "SecurityGroupIds": []
    },
    "MemorySize": 256,
    "RevisionId": "cd669f21-0f3d-4e1c-9566-948837f2e2ea",
    "CodeSize": 266,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function:2",
    "Handler": "index.handler",
    "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-
role-uy3l9yq",
    "Timeout": 3,
    "LastModified": "2019-10-01T16:47:28.490+0000",
    "Runtime": "nodejs10.x",
    "Description": "newer version"
    }
]
}

```

Para obtener más información, consulte [Configuración de alias de funciones de AWS Lambda en la Guía para desarrolladores](#) de Lambda AWS .

- Para API obtener más información, consulte la Referencia de [ListVersionsByFunction](#) comandos AWS CLI .

## publish-layer-version

En el siguiente ejemplo de código, se muestra cómo usar `publish-layer-version`.

### AWS CLI

Para crear una versión de capa Lambda

El siguiente `publish-layer-version` ejemplo crea una nueva versión de capa de biblioteca de Python. El comando recupera el contenido de la capa, un archivo denominado `layer.zip` en el bucket de S3 especificado.

```
aws lambda publish-layer-version \  
  --layer-name my-layer \  
  --description "My Python layer" \  
  --license-info "MIT" \  
  --content S3Bucket=Lambda-layers-us-west-2-123456789012,S3Key=layer.zip \  
  --compatible-runtimes python3.10 python3.11
```

Salida:

```
{  
  "Content": {  
    "Location": "https://awslambda-us-west-2-layers.s3.us-west-2.amazonaws.com/  
snapshots/123456789012/my-layer-4aaa2fbb-ff77-4b0a-ad92-5b78a716a96a?  
versionId=27iWyA73cCAYqyH...",  
    "CodeSha256": "tv9jJ0+rPbXUUXuRKi7CwHzKtLDkDRJLB3cC3Z/ouXo=",  
    "CodeSize": 169  
  },  
  "LayerArn": "arn:aws:lambda:us-west-2:123456789012:layer:my-layer",  
  "LayerVersionArn": "arn:aws:lambda:us-west-2:123456789012:layer:my-layer:1",  
  "Description": "My Python layer",  
  "CreateDate": "2023-11-14T23:03:52.894+0000",  
  "Version": 1,  
  "LicenseInfo": "MIT",  
  "CompatibleRuntimes": [  
    "python3.10",  
    "python3.11"  
  ]  
}
```

Para obtener más información, consulte [Capas AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [PublishLayerVersion](#) de AWS CLI comandos.

## publish-version

En el siguiente ejemplo de código, se muestra cómo usar `publish-version`.

### AWS CLI

Publicación de una nueva versión de la función de Lambda

En el siguiente ejemplo de `publish-version`, se publica una nueva versión de la función de Lambda `my-function`.

```
aws lambda publish-version \  
  --function-name my-function
```

Salida:

```
{  
  "TracingConfig": {  
    "Mode": "PassThrough"  
  },  
  "CodeSha256": "dBG9m8SGdm1Ejw/JYX1hhvCrAv5TxvXsbl/RM1r0fT/I=",  
  "FunctionName": "my-function",  
  "CodeSize": 294,  
  "RevisionId": "f31d3d39-cc63-4520-97d4-43cd44c94c20",  
  "MemorySize": 128,  
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:3",  
  "Version": "2",  
  "Role": "arn:aws:iam::123456789012:role/service-role/MyTestFunction-role-zgur6bf4",  
  "Timeout": 3,  
  "LastModified": "2019-09-23T18:32:33.857+0000",  
  "Handler": "my-function.handler",  
  "Runtime": "nodejs10.x",  
  "Description": ""  
}
```

Para obtener más información, consulte [Configuración de alias de funciones de AWS Lambda en la Guía para desarrolladores](#) de Lambda AWS .

- Para API obtener más información, consulte la Referencia de [PublishVersion](#) comandos AWS CLI .

## **put-function-concurrency**

En el siguiente ejemplo de código, se muestra cómo usar `put-function-concurrency`.

AWS CLI

Configuración de un límite de simultaneidad reservado para una función

En el siguiente ejemplo de `put-function-concurrency`, se configuran 100 ejecuciones simultáneas reservadas para la función `my-function`.

```
aws lambda put-function-concurrency \  
  --function-name my-function \  
  --reserved-concurrent-executions 100
```

Salida:

```
{  
  "ReservedConcurrentExecutions": 100  
}
```

Para obtener más información, consulte [Reserva de simultaneidad para una función de Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte [PutFunctionConcurrency](#) la Referencia de AWS CLI comandos.

## `put-function-event-invoke-config`

En el siguiente ejemplo de código, se muestra cómo usar `put-function-event-invoke-config`.

AWS CLI

Para configurar la gestión de errores para la invocación asíncrona

En el siguiente `put-function-event-invoke-config` ejemplo, se establece una duración máxima del evento de una hora y se deshabilitan los reintentos para la función especificada.

```
aws lambda put-function-event-invoke-config \  
  --function-name my-function \  
  --maximum-event-age-in-seconds 3600 \  
  --maximum-retry-attempts 0
```

Salida:

```
{  
  "LastModified": 1573686021.479,
```

```
"FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-function:
$LATEST",
  "MaximumRetryAttempts": 0,
  "MaximumEventAgeInSeconds": 3600,
  "DestinationConfig": {
    "OnSuccess": {},
    "OnFailure": {}
  }
}
```

- Para API obtener más información, consulte la Referencia [PutFunctionEventInvokeConfig](#) de AWS CLI comandos.

## put-provisioned-concurrency-config

En el siguiente ejemplo de código, se muestra cómo usar `put-provisioned-concurrency-config`.

### AWS CLI

Asignación de simultaneidad aprovisionada

En el siguiente ejemplo de `put-provisioned-concurrency-config`, se asignan 100 simultaneidades aprovisionadas para el alias `BLUE` de la función especificada.

```
aws lambda put-provisioned-concurrency-config \
  --function-name my-function \
  --qualifier BLUE \
  --provisioned-concurrent-executions 100
```

Salida:

```
{
  "Requested ProvisionedConcurrentExecutions": 100,
  "Allocated ProvisionedConcurrentExecutions": 0,
  "Status": "IN_PROGRESS",
  "LastModified": "2019-11-21T19:32:12+0000"
}
```

- Para API obtener más información, consulte [PutProvisionedConcurrencyConfig](#) la Referencia de AWS CLI comandos.



## remove-layer-version-permission

En el siguiente ejemplo de código, se muestra cómo usar `remove-layer-version-permission`.

### AWS CLI

Para eliminar los permisos de las versiones de capa

En el siguiente `remove-layer-version-permission` ejemplo, se elimina el permiso de una cuenta para configurar una versión de capa.

```
aws lambda remove-layer-version-permission \  
  --layer-name my-layer \  
  --statement-id xaccount \  
  --version-number 1
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Capas AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [RemoveLayerVersionPermission](#) de AWS CLI comandos.

## remove-permission

En el siguiente ejemplo de código, se muestra cómo usar `remove-permission`.

### AWS CLI

Eliminación de permisos de una función de Lambda existente

En el siguiente ejemplo de `remove-permission`, se elimina el permiso para invocar una función denominada `my-function`.

```
aws lambda remove-permission \  
  --function-name my-function \  
  --statement-id sns
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de políticas basadas en recursos para Lambda AWS en la Guía para desarrolladores](#) de AWS Lambda.

- Para obtener API más información, consulte la Referencia de comandos [RemovePermission](#).AWS CLI

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

AWS CLI

Adición de etiquetas a una función de Lambda existente

En el siguiente ejemplo de `tag-resource`, se agrega una etiqueta con el nombre de clave `DEPARTMENT` y un valor de `Department A` a la función de Lambda especificada.

```
aws lambda tag-resource \  
  --resource arn:aws:lambda:us-west-2:123456789012:function:my-function \  
  --tags "DEPARTMENT=Department A"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetas de funciones de Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

AWS CLI

Eliminación de etiquetas de una función de Lambda existente

En el siguiente ejemplo de `untag-resource`, se elimina la etiqueta con el nombre de clave `DEPARTMENT` de la función de Lambda `my-function`.

```
aws lambda untag-resource \  
  --resource arn:aws:lambda:us-west-2:123456789012:function:my-function \  
  --tag-keys DEPARTMENT
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetas de funciones de Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-alias

En el siguiente ejemplo de código, se muestra cómo usar `update-alias`.

### AWS CLI

Actualización del alias de una función

En el siguiente ejemplo de `update-alias`, se actualiza el alias nombrado `LIVE` para que apunte a la versión 3 de la función de Lambda `my-function`.

```
aws lambda update-alias \  
  --function-name my-function \  
  --function-version 3 \  
  --name LIVE
```

Salida:

```
{  
  "FunctionVersion": "3",  
  "Name": "LIVE",  
  "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:LIVE",  
  "RevisionId": "594f41fb-b85f-4c20-95c7-6ca5f2a92c93",  
  "Description": "alias for live version of function"  
}
```

Para obtener más información, consulte [Configuración de alias de funciones de AWS Lambda en la Guía para desarrolladores](#) de Lambda AWS .

- Para API obtener más información, consulte la Referencia de [UpdateAlias](#) comandos AWS CLI .

## update-event-source-mapping

En el siguiente ejemplo de código, se muestra cómo usar `update-event-source-mapping`.

## AWS CLI

Para actualizar el mapeo entre un origen de eventos y una función AWS Lambda

En el siguiente `update-event-source-mapping` ejemplo, se actualiza el tamaño del lote a 8 en la asignación especificada.

```
aws lambda update-event-source-mapping \  
  --uuid "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE" \  
  --batch-size 8
```

Salida:

```
{  
  "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",  
  "StateTransitionReason": "USER_INITIATED",  
  "LastModified": 1569284520.333,  
  "BatchSize": 8,  
  "State": "Updating",  
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",  
  "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"  
}
```

Para obtener más información, consulte [Mapeo de fuentes de eventos de AWS Lambda](#) en la Guía para desarrolladores de AWS Lambda.

- Para API obtener más información, consulte la Referencia [UpdateEventSourceMapping](#) de AWS CLI comandos.

## update-function-code

En el siguiente ejemplo de código, se muestra cómo usar `update-function-code`.

### AWS CLI

Cómo actualizar el código de una función de Lambda

En el siguiente `update-function-code` ejemplo, se reemplaza el código de la versión no publicada (`$LATEST`) de la `my-function` función por el contenido del archivo zip especificado.

```
aws lambda update-function-code \  
  --function-name my-function --zip-file fileb://my-function.zip
```

```
--function-name my-function \  
--zip-file fileb://my-function.zip
```

Salida:

```
{  
  "FunctionName": "my-function",  
  "LastModified": "2019-09-26T20:28:40.438+0000",  
  "RevisionId": "e52502d4-9320-4688-9cd6-152a6ab7490d",  
  "MemorySize": 256,  
  "Version": "$LATEST",  
  "Role": "arn:aws:iam::123456789012:role/service-role/my-function-role-uy3l9qq",  
  "Timeout": 3,  
  "Runtime": "nodejs10.x",  
  "TracingConfig": {  
    "Mode": "PassThrough"  
  },  
  "CodeSha256": "5tT2qgzYUHaqwR716pZ2dpkn/0J1FrzJmlKidWoaCgk=",  
  "Description": "",  
  "VpcConfig": {  
    "SubnetIds": [],  
    "VpcId": "",  
    "SecurityGroupIds": []  
  },  
  "CodeSize": 304,  
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",  
  "Handler": "index.handler"  
}
```

Para obtener más información, consulte [Configuración las funciones de Lambda de AWS](#) en la Guía para desarrolladores de Lambda de AWS .

- Para API obtener más información, consulte [UpdateFunctionCode](#) la Referencia de AWS CLI comandos.

## update-function-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-function-configuration`.

### AWS CLI

Para modificar la configuración de una función

El siguiente `update-function-configuration` ejemplo modifica el tamaño de la memoria para que sea de 256 MB para la versión no publicada (`$LATEST`) de la `my-function` función.

```
aws lambda update-function-configuration \  
  --function-name my-function \  
  --memory-size 256
```

Salida:

```
{  
  "FunctionName": "my-function",  
  "LastModified": "2019-09-26T20:28:40.438+0000",  
  "RevisionId": "e52502d4-9320-4688-9cd6-152a6ab7490d",  
  "MemorySize": 256,  
  "Version": "$LATEST",  
  "Role": "arn:aws:iam::123456789012:role/service-role/my-function-role-uy319qq",  
  "Timeout": 3,  
  "Runtime": "nodejs10.x",  
  "TracingConfig": {  
    "Mode": "PassThrough"  
  },  
  "CodeSha256": "5tT2qgzYUHaqwR716pZ2dpkn/0J1FrzJm1KidWoaCgk=",  
  "Description": "",  
  "VpcConfig": {  
    "SubnetIds": [],  
    "VpcId": "",  
    "SecurityGroupIds": []  
  },  
  "CodeSize": 304,  
  "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",  
  "Handler": "index.handler"  
}
```

Para obtener más información, consulte [Configuración las funciones de Lambda de AWS](#) en la Guía para desarrolladores de Lambda de AWS .

- Para API obtener más información, consulte la Referencia [UpdateFunctionConfiguration](#) de AWS CLI comandos.

## update-function-event-invoke-config

En el siguiente ejemplo de código, se muestra cómo usar `update-function-event-invoke-config`.

### AWS CLI

Para actualizar una configuración de invocación asíncrona

El siguiente `update-function-event-invoke-config` ejemplo agrega un destino en caso de error a la configuración de invocación asíncrona existente para la función especificada.

```
aws lambda update-function-event-invoke-config \  
  --function-name my-function \  
  --destination-config '{"OnFailure":{"Destination": "arn:aws:sqs:us-  
east-2:123456789012:destination"}}'
```

Salida:

```
{  
  "LastModified": 1573687896.493,  
  "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-function:  
$LATEST",  
  "MaximumRetryAttempts": 0,  
  "MaximumEventAgeInSeconds": 3600,  
  "DestinationConfig": {  
    "OnSuccess": {},  
    "OnFailure": {  
      "Destination": "arn:aws:sqs:us-east-2:123456789012:destination"  
    }  
  }  
}
```

- Para obtener API más información, consulte la Referencia de comandos.

[UpdateFunctionEventInvokeConfig](#) AWS CLI

## Ejemplos de License Manager que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface mediante License Manager.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **create-license-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `create-license-configuration`.

AWS CLI

Ejemplo 1: Para crear una configuración de licencia

El siguiente `create-license-configuration` ejemplo crea una configuración de licencia con un límite estricto de 10 núcleos.

```
aws license-manager create-license-configuration --name my-license-configuration \  
  --license-counting-type Core \  
  --license-count 10 \  
  --license-count-hard-limit
```

Salida:

```
{  
  "LicenseConfigurationArn": "arn:aws:license-manager:us-  
west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba41EXAMPLE1111"  
}
```

Ejemplo 2: Para crear una configuración de licencia

El siguiente `create-license-configuration` ejemplo crea una configuración de licencia con un límite flexible de 100vCPUs. Utiliza una regla para habilitar la CPU optimización v.

```
aws license-manager create-license-configuration --name my-license-configuration
```



```
--license-counting-type vCPU \  
--license-count 100 \  
--license-rules "#honorVcpuOptimization=true"
```

Salida:

```
{  
  "LicenseConfigurationArn": "arn:aws:license-manager:us-  
west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba41EXAMPLE2222"  
}
```

- Para API obtener más información, consulte [CreateLicenseConfiguration](#) la Referencia de AWS CLI comandos.

## delete-license-configuration

En el siguiente ejemplo de código, se muestra cómo usar `delete-license-configuration`.

AWS CLI

Para eliminar una configuración de licencia

En el siguiente `delete-license-configuration` ejemplo, se elimina la configuración de licencia especificada.

```
aws license-manager delete-license-configuration \  
  --license-configuration-arn arn:aws:license-manager:us-  
west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteLicenseConfiguration](#) la Referencia de AWS CLI comandos.

## get-license-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-license-configuration`.

AWS CLI

Para obtener información sobre la configuración de la licencia

En el siguiente `get-license-configuration` ejemplo, se muestran los detalles de la configuración de licencia especificada.

```
aws license-manager get-license-configuration \  
  --license-configuration-arn arn:aws:license-manager:us-  
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE
```

Salida:

```
{  
  "LicenseConfigurationId": "lic-38b658717b87478aaa7c00883EXAMPLE",  
  "LicenseConfigurationArn": "arn:aws:license-manager:us-  
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE",  
  "Name": "my-license-configuration",  
  "LicenseCountingType": "vCPU",  
  "LicenseRules": [],  
  "LicenseCountHardLimit": false,  
  "ConsumedLicenses": 0,  
  "Status": "AVAILABLE",  
  "OwnerAccountId": "123456789012",  
  "ConsumedLicenseSummaryList": [  
    {  
      "ResourceType": "EC2_INSTANCE",  
      "ConsumedLicenses": 0  
    },  
    {  
      "ResourceType": "EC2_HOST",  
      "ConsumedLicenses": 0  
    },  
    {  
      "ResourceType": "SYSTEMS_MANAGER_MANAGED_INSTANCE",  
      "ConsumedLicenses": 0  
    }  
  ],  
  "ManagedResourceSummaryList": [  
    {  
      "ResourceType": "EC2_INSTANCE",  
      "AssociationCount": 0  
    },  
    {  
      "ResourceType": "EC2_HOST",  
      "AssociationCount": 0  
    }  
  ],  
}
```

```
{
  "ResourceType": "EC2_AMI",
  "AssociationCount": 2
},
{
  "ResourceType": "SYSTEMS_MANAGER_MANAGED_INSTANCE",
  "AssociationCount": 0
}
]
```

- Para API obtener más información, consulte [GetLicenseConfiguration](#) la Referencia de AWS CLI comandos.

## get-service-settings

En el siguiente ejemplo de código, se muestra cómo usar `get-service-settings`.

### AWS CLI

Para obtener la configuración de License Manager

El siguiente `get-service-settings` ejemplo muestra la configuración del servicio de License Manager en la región actual.

```
aws license-manager get-service-settings
```

A continuación, se muestra un ejemplo de resultado si la detección de recursos entre cuentas está deshabilitada.

```
{
  "OrganizationConfiguration": {
    "EnableIntegration": false
  },
  "EnableCrossAccountsDiscovery": false
}
```

A continuación, se muestra un ejemplo de resultado si la detección de recursos entre cuentas está habilitada.

```
{
```

```

    "S3BucketArn": "arn:aws:s3::aws-license-manager-service-c22d6279-35c4-47c4-bb",
    "OrganizationConfiguration": {
      "EnableIntegration": true
    },
    "EnableCrossAccountsDiscovery": true
  }

```

- Para API obtener más información, consulte [GetServiceSettings](#) la Referencia de AWS CLI comandos.

## list-associations-for-license-configuration

En el siguiente ejemplo de código, se muestra cómo usar `list-associations-for-license-configuration`.

### AWS CLI

Para obtener asociaciones para una configuración de licencia

En el siguiente `list-associations-for-license-configuration` ejemplo, se muestra información detallada sobre las asociaciones de la configuración de licencia especificada.

```

aws license-manager list-associations-for-license-configuration \
  --license-configuration-arn arn:aws:license-manager:us-west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE

```

Salida:

```

{
  "LicenseConfigurationAssociations": [
    {
      "ResourceArn": "arn:aws:ec2:us-west-2::image/ami-1234567890abcdef0",
      "ResourceType": "EC2_AMI",
      "ResourceOwnerId": "123456789012",
      "AssociationTime": 1568825118.617
    },
    {
      "ResourceArn": "arn:aws:ec2:us-west-2::image/ami-0abcdef1234567890",
      "ResourceType": "EC2_AMI",
      "ResourceOwnerId": "123456789012",
      "AssociationTime": 1568825118.946
    }
  ]
}

```

```
]
}
```

- Para API obtener más información, consulte [ListAssociationsForLicenseConfiguration](#) la Referencia de AWS CLI comandos.

## list-license-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-license-configurations`.

### AWS CLI

Ejemplo 1: Para enumerar todas las configuraciones de licencia

En el siguiente `list-license-configurations` ejemplo, se enumeran todas las configuraciones de licencia.

```
aws license-manager list-license-configurations
```

Salida:

```
{
  "LicenseConfigurations": [
    {
      "LicenseConfigurationId": "lic-6eb6586f508a786a2ba4f56c1EXAMPLE",
      "LicenseConfigurationArn": "arn:aws:license-manager:us-west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE",
      "Name": "my-license-configuration",
      "LicenseCountingType": "Core",
      "LicenseRules": [],
      "LicenseCount": 10,
      "LicenseCountHardLimit": true,
      "ConsumedLicenses": 0,
      "Status": "AVAILABLE",
      "OwnerAccountId": "123456789012",
      "ConsumedLicenseSummaryList": [
        {
          "ResourceType": "EC2_INSTANCE",
          "ConsumedLicenses": 0
        },
        {
          "ResourceType": "EC2_HOST",
```

```

        "ConsumedLicenses": 0
      },
      {
        "ResourceType": "SYSTEMS_MANAGER_MANAGED_INSTANCE",
        "ConsumedLicenses": 0
      }
    ],
    "ManagedResourceSummaryList": [
      {
        "ResourceType": "EC2_INSTANCE",
        "AssociationCount": 0
      },
      {
        "ResourceType": "EC2_HOST",
        "AssociationCount": 0
      },
      {
        "ResourceType": "EC2_AMI",
        "AssociationCount": 0
      },
      {
        "ResourceType": "SYSTEMS_MANAGER_MANAGED_INSTANCE",
        "AssociationCount": 0
      }
    ]
  },
  {
    ...
  }
]
}

```

Ejemplo 2: Para enumerar una configuración de licencia específica

El siguiente `list-license-configurations` ejemplo muestra solo la configuración de licencia especificada.

```

aws license-manager list-license-configurations \
  --license-configuration-arns arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE

```

- Para API obtener más información, consulte [ListLicenseConfigurations](#) la Referencia de AWS CLI comandos.

## list-license-specifications-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-license-specifications-for-resource`.

### AWS CLI

Para enumerar las configuraciones de licencia de un recurso

En el siguiente `list-license-specifications-for-resource` ejemplo, se enumeran las configuraciones de licencia asociadas a la Amazon Machine Image (AMI) especificada.

```
aws license-manager list-license-specifications-for-resource \  
  --resource-arn arn:aws:ec2:us-west-2:image/ami-1234567890abcdef0
```

Salida:

```
{  
  "LicenseConfigurationArn": "arn:aws:license-manager:us-  
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE"  
}
```

- Para API obtener más información, consulte [ListLicenseSpecificationsForResource](#) la Referencia de AWS CLI comandos.

## list-resource-inventory

En el siguiente ejemplo de código, se muestra cómo usar `list-resource-inventory`.

### AWS CLI

Para enumerar los recursos en el inventario de recursos

En el siguiente `list-resource-inventory` ejemplo, se enumeran los recursos gestionados mediante el inventario de Systems Manager.

```
aws license-manager list-resource-inventory
```

Salida:

```
{  
  "ResourceInventoryList": [  
    {  
      "ResourceName": "ExampleResource",  
      "ResourceType": "EXAMPLE_RESOURCE",  
      "Status": "ACTIVE",  
      "CreationTime": "2019-01-01T00:00:00Z",  
      "LastModifiedTime": "2019-01-01T00:00:00Z",  
      "Tags": [{"Key": "ExampleTag", "Value": "ExampleValue"}],  
      "LicenseConfigurationArn": "arn:aws:license-manager:us-west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE"  
    }  
  ]  
}
```

```

    {
      "Platform": "Red Hat Enterprise Linux Server",
      "ResourceType": "EC2Instance",
      "PlatformVersion": "7.4",
      "ResourceArn": "arn:aws:ec2:us-west-2:1234567890129:instance/
i-05d3cdfb05bd36376",
      "ResourceId": "i-05d3cdfb05bd36376",
      "ResourceOwningAccountId": "1234567890129"
    },
    {
      "Platform": "Amazon Linux",
      "ResourceType": "EC2Instance",
      "PlatformVersion": "2",
      "ResourceArn": "arn:aws:ec2:us-west-2:1234567890129:instance/
i-0b1d036cfd4594808",
      "ResourceId": "i-0b1d036cfd4594808",
      "ResourceOwningAccountId": "1234567890129"
    },
    {
      "Platform": "Microsoft Windows Server 2019 Datacenter",
      "ResourceType": "EC2Instance",
      "PlatformVersion": "10.0.17763",
      "ResourceArn": "arn:aws:ec2:us-west-2:1234567890129:instance/
i-0cdb3b54a2a8246ad",
      "ResourceId": "i-0cdb3b54a2a8246ad",
      "ResourceOwningAccountId": "1234567890129"
    }
  ]
}

```

- Para API obtener más información, consulte [ListResourceInventory](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de una configuración de licencia

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas de la configuración de licencia especificada.



```
aws license-manager list-tags-for-resource \  
  --resource-arn arn:aws:license-manager:us-west-2:123456789012:license-  
configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Salida:

```
{  
  "Tags": [  
    {  
      "Key": "project",  
      "Value": "lima"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## list-usage-for-license-configuration

En el siguiente ejemplo de código, se muestra cómo usar `list-usage-for-license-configuration`.

AWS CLI

Para enumerar las licencias en uso para una configuración de licencia

En el siguiente `list-usage-for-license-configuration` ejemplo, se muestra información sobre los recursos que utilizan licencias para la configuración de licencia especificada. Por ejemplo, si el tipo de licencia es vCPU, todas las instancias consumen una licencia por cada vCPU.

```
aws license-manager list-usage-for-license-configuration \  
  --license-configuration-arn arn:aws:license-manager:us-  
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE
```

Salida:

```
{  
  "LicenseConfigurationUsageList": [  
    {
```

```

        "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/
i-04a636d18e83cfacb",
        "ResourceType": "EC2_INSTANCE",
        "ResourceStatus": "running",
        "ResourceOwnerId": "123456789012",
        "AssociationTime": 1570892850.519,
        "ConsumedLicenses": 2
    }
]
}

```

- Para API obtener más información, consulte [ListUsageForLicenseConfiguration](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir una etiqueta, una configuración de licencia

En el siguiente `tag-resource` ejemplo, se agrega la etiqueta especificada (nombre y valor de la clave) a la configuración de licencia especificada.

```

aws license-manager tag-resource \
  --tags Key=project,Value=Lima \
  --resource-arn arn:aws:license-manager:us-west-2:123456789012:license-
configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE

```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar etiquetas de una configuración de licencia

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta especificada (nombre de clave y recurso) de la configuración de licencia especificada.

```
aws license-manager untag-resource \  
  --tag-keys project \  
  --resource-arn arn:aws:license-manager:us-west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-license-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-license-configuration`.

### AWS CLI

Para actualizar la configuración de una licencia

El siguiente `update-license-configuration` ejemplo actualiza la configuración de licencia especificada para eliminar el límite estricto.

```
aws license-manager update-license-configuration \  
  --no-license-count-hard-limit \  
  --license-configuration-arn arn:aws:license-manager:us-west-2:880185128111:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Este comando no genera ninguna salida.

El siguiente `update-license-configuration` ejemplo actualiza la configuración de licencia especificada para cambiar su estado a `DISABLED`.

```
aws license-manager update-license-configuration \  
  --license-configuration-status DISABLED \  
  --license-configuration-arn arn:aws:license-manager:us-west-2:880185128111:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UpdateLicenseConfiguration](#) la Referencia de AWS CLI comandos.

## update-license-specifications-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `update-license-specifications-for-resource`.

### AWS CLI

Para actualizar las configuraciones de licencia de un recurso

En el siguiente `update-license-specifications-for-resource` ejemplo, se reemplaza la configuración de licencia asociada a la Amazon Machine Image (AMI) especificada mediante la eliminación de una configuración de licencia y la adición de otra.

```
aws license-manager update-license-specifications-for-resource \  
  --resource-arn arn:aws:ec2:us-west-2:image/ami-1234567890abcdef0 \  
  --remove-license-specifications LicenseConfigurationArn=arn:aws:license-  
manager:us-west-2:123456789012:license-  
configuration:lic-38b658717b87478aaa7c00883EXAMPLE \  
  --add-license-specifications LicenseConfigurationArn=arn:aws:license-manager:us-  
west-2:123456789012:license-configuration:lic-42b6deb06e5399a980d555927EXAMPLE
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UpdateLicenseSpecificationsForResource](#) la Referencia de AWS CLI comandos.

## update-service-settings

En el siguiente ejemplo de código, se muestra cómo usar `update-service-settings`.

### AWS CLI

Para actualizar la configuración de License Manager

El siguiente `update-service-settings` ejemplo permite la detección de recursos entre cuentas para License Manager en la AWS región actual. El bucket de Amazon S3 es el recurso de sincronización de datos necesario para el inventario de Systems Manager.

```
aws license-manager update-service-settings \  

```

```
--organization-configuration EnableIntegration=true \  
--enable-cross-accounts-discovery \  
--s3-bucket-arn arn:aws:s3:::aws-license-manager-service-abcd1234EXAMPLE
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UpdateServiceSettings](#) la Referencia de AWS CLI comandos.

## Ejemplos de Lightsail usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Lightsail.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **allocate-static-ip**

En el siguiente ejemplo de código, se muestra cómo usar `allocate-static-ip`.

AWS CLI

Para crear una IP estática

En el siguiente `allocate-static-ip` ejemplo, se crea la IP estática especificada, que se puede adjuntar a una instancia.

```
aws lightsail allocate-static-ip \  
--static-ip-name StaticIp-1
```

Salida:

```
{
  "operations": [
    {
      "id": "b5d06d13-2f19-4683-889f-dEXAMPLEed79",
      "resourceName": "StaticIp-1",
      "resourceType": "StaticIp",
      "createdAt": 1571071325.076,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationType": "AllocateStaticIp",
      "status": "Succeeded",
      "statusChangedAt": 1571071325.274
    }
  ]
}
```

- Para API obtener más información, consulta [AllocateStaticIp](#) la Referencia de AWS CLI comandos.

## attach-disk

En el siguiente ejemplo de código, se muestra cómo usar attach-disk.

### AWS CLI

Para adjuntar un disco de almacenamiento en bloque a una instancia

En el siguiente attach-disk ejemplo, se conecta un disco Disk-1 a una instancia WordPress\_Multisite-1 con la ruta de disco de /dev/xvdf

```
aws lightsail attach-disk \
  --disk-name Disk-1 \
  --disk-path /dev/xvdf \
  --instance-name WordPress_Multisite-1
```

Salida:

```
{
  "operations": [
```

```

    {
      "id": "10a08267-19ce-43be-b913-6EXAMPLE7e80",
      "resourceName": "Disk-1",
      "resourceType": "Disk",
      "createdAt": 1571071465.472,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationDetails": "WordPress_Multisite-1",
      "operationType": "AttachDisk",
      "status": "Started",
      "statusChangedAt": 1571071465.472
    },
    {
      "id": "2912c477-5295-4539-88c9-bEXAMPLEd1f0",
      "resourceName": "WordPress_Multisite-1",
      "resourceType": "Instance",
      "createdAt": 1571071465.474,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationDetails": "Disk-1",
      "operationType": "AttachDisk",
      "status": "Started",
      "statusChangedAt": 1571071465.474
    }
  ]
}

```

- Para API obtener más información, consulte [AttachDisk](#) la Referencia de AWS CLI comandos.

## **attach-instances-to-load-balancer**

En el siguiente ejemplo de código, se muestra cómo usar `attach-instances-to-load-balancer`.

### AWS CLI

Para adjuntar instancias a un balanceador de cargas

En el siguiente `attach-instances-to-load-balancer` ejemplo MEAN-1, MEAN-2 se adjuntan las instancias y MEAN-3 al equilibrador de cargas. `LoadBalancer-1`

```
aws lightsail attach-instances-to-load-balancer \  
  --instance-names {"MEAN-1","MEAN-2","MEAN-3"} \  
  --load-balancer-name LoadBalancer-1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "8055d19d-abb2-40b9-b527-1EXAMPLE3c7b",  
      "resourceName": "LoadBalancer-1",  
      "resourceType": "LoadBalancer",  
      "createdAt": 1571071699.892,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "MEAN-2",  
      "operationType": "AttachInstancesToLoadBalancer",  
      "status": "Started",  
      "statusChangedAt": 1571071699.892  
    },  
    {  
      "id": "c35048eb-8538-456a-a118-0EXAMPLEfb73",  
      "resourceName": "MEAN-2",  
      "resourceType": "Instance",  
      "createdAt": 1571071699.887,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "LoadBalancer-1",  
      "operationType": "AttachInstancesToLoadBalancer",  
      "status": "Started",  
      "statusChangedAt": 1571071699.887  
    },  
    {  
      "id": "910d09e0-adc5-4372-bc2e-0EXAMPLEd891",
```



```
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancer",
    "createdAt": 1571071699.882,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "MEAN-3",
    "operationType": "AttachInstancesToLoadBalancer",
    "status": "Started",
    "statusChangedAt": 1571071699.882
  },
  {
    "id": "178b18ac-43e8-478c-9bed-1EXAMPLE4755",
    "resourceName": "MEAN-3",
    "resourceType": "Instance",
    "createdAt": 1571071699.901,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "LoadBalancer-1",
    "operationType": "AttachInstancesToLoadBalancer",
    "status": "Started",
    "statusChangedAt": 1571071699.901
  },
  {
    "id": "fb62536d-2a98-4190-a6fc-4EXAMPLE7470",
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancer",
    "createdAt": 1571071699.885,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "MEAN-1",
    "operationType": "AttachInstancesToLoadBalancer",
    "status": "Started",
    "statusChangedAt": 1571071699.885
  },
  {
```

```

    "id": "787dac0d-f98d-46c3-8571-3EXAMPLE5a85",
    "resourceName": "MEAN-1",
    "resourceType": "Instance",
    "createdAt": 1571071699.901,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "LoadBalancer-1",
    "operationType": "AttachInstancesToLoadBalancer",
    "status": "Started",
    "statusChangedAt": 1571071699.901
  }
]
}

```

- Para API obtener más información, consulta la Referencia [AttachInstancesToLoadBalancer](#) de AWS CLI comandos.

## attach-load-balancer-tls-certificate

En el siguiente ejemplo de código, se muestra cómo usar `attach-load-balancer-tls-certificate`.

### AWS CLI

Para adjuntar un TLS certificado a un balanceador de cargas

En el siguiente `attach-load-balancer-tls-certificate` ejemplo, se adjunta el TLS certificado del equilibrador de cargas `Certificate2` al equilibrador de cargas `LoadBalancer-1`

```

aws lightsail attach-load-balancer-tls-certificate \
  --certificate-name Certificate2 \
  --load-balancer-name LoadBalancer-1

```

Salida:

```

{
  "operations": [
    {

```

```

    "id": "cf1ad6e3-3cbb-4b8a-a7f2-3EXAMPLEa118",
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancer",
    "createdAt": 1571072255.416,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "Certificate2",
    "operationType": "AttachLoadBalancerTlsCertificate",
    "status": "Succeeded",
    "statusChangedAt": 1571072255.416
  },
  {
    "id": "dae1bcfb-d531-4c06-b4ea-bEXAMPLEc04e",
    "resourceName": "Certificate2",
    "resourceType": "LoadBalancerTlsCertificate",
    "createdAt": 1571072255.416,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "LoadBalancer-1",
    "operationType": "AttachLoadBalancerTlsCertificate",
    "status": "Succeeded",
    "statusChangedAt": 1571072255.416
  }
]
}

```

- Para API obtener más información, consulta la Referencia de [AttachLoadBalancerTlsCertificate](#) comandos AWS CLI .

## attach-static-ip

En el siguiente ejemplo de código, se muestra cómo usar `attach-static-ip`.

### AWS CLI

Para adjuntar una IP estática a una instancia

En el siguiente `attach-static-ip` ejemplo, se adjunta una IP estática `StaticIp-1` a la instancia `MEAN-1`.

```
aws lightsail attach-static-ip \  
  --static-ip-name StaticIp-1 \  
  --instance-name MEAN-1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "45e6fa13-4808-4b8d-9292-bEXAMPLE20b2",  
      "resourceName": "StaticIp-1",  
      "resourceType": "StaticIp",  
      "createdAt": 1571072569.375,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationDetails": "MEAN-1",  
      "operationType": "AttachStaticIp",  
      "status": "Succeeded",  
      "statusChangedAt": 1571072569.375  
    },  
    {  
      "id": "9ee09a17-863c-4e51-8a6d-3EXAMPLE5475",  
      "resourceName": "MEAN-1",  
      "resourceType": "Instance",  
      "createdAt": 1571072569.376,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationDetails": "StaticIp-1",  
      "operationType": "AttachStaticIp",  
      "status": "Succeeded",  
      "statusChangedAt": 1571072569.376  
    }  
  ]  
}
```

```
}
```

- Para API obtener más información, consulte [AttachStaticIp](#) la Referencia de AWS CLI comandos.

## close-instance-public-ports

En el siguiente ejemplo de código, se muestra cómo usar `close-instance-public-ports`.

### AWS CLI

Para cerrar los puertos del firewall de una instancia

En el siguiente `close-instance-public-ports` ejemplo, se cierra 22 el TCP puerto de una instancia `MEAN-2`.

```
aws lightsail close-instance-public-ports \  
  --instance-name MEAN-2 \  
  --port-info fromPort=22,protocol=TCP,toPort=22
```

Salida:

```
{  
  "operation": {  
    "id": "4f328636-1c96-4649-ae6d-1EXAMPLEf446",  
    "resourceName": "MEAN-2",  
    "resourceType": "Instance",  
    "createdAt": 1571072845.737,  
    "location": {  
      "availabilityZone": "us-west-2a",  
      "regionName": "us-west-2"  
    },  
    "isTerminal": true,  
    "operationDetails": "22/tcp",  
    "operationType": "CloseInstancePublicPorts",  
    "status": "Succeeded",  
    "statusChangedAt": 1571072845.737  
  }  
}
```

- Para API obtener más información, consulte [CloseInstancePublicPorts](#) la Referencia de AWS CLI comandos.

## copy-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `copy-snapshot`.

### AWS CLI

Ejemplo 1: Para copiar una instantánea dentro de la misma AWS región

El siguiente `copy-snapshot` ejemplo copia una instantánea de instancia `MEAN-1-1571075291` como una instantánea de instancia `MEAN-1-Copy` dentro de la misma AWS región `us-west-2`.

```
aws lightsail copy-snapshot \  
  --source-snapshot-name MEAN-1-1571075291 \  
  --target-snapshot-name MEAN-1-Copy \  
  --source-region us-west-2
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "ced16fc1-f401-4556-8d82-1EXAMPLEb982",  
      "resourceName": "MEAN-1-Copy",  
      "resourceType": "InstanceSnapshot",  
      "createdAt": 1571075581.498,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "us-west-2:MEAN-1-1571075291",  
      "operationType": "CopySnapshot",  
      "status": "Started",  
      "statusChangedAt": 1571075581.498  
    }  
  ]  
}
```

Para obtener más información, consulte [Copiar instantáneas de una AWS región a otra en Amazon Lightsail en](#) la Guía para desarrolladores de Lightsail.

Ejemplo 2: Para copiar una instantánea de una región a otra AWS

En el siguiente `copy-snapshot` ejemplo, se copia la instantánea de la instancia MEAN-1-1571075291 como instantánea MEAN-1-1571075291-Copy de AWS la instancia de una región `us-west-2` a `us-east-1`.

```
aws lightsail copy-snapshot \  
  --source-snapshot-name MEAN-1-1571075291 \  
  --target-snapshot-name MEAN-1-1571075291-Copy \  
  --source-region us-west-2 \  
  --region us-east-1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "91116b79-119c-4451-b44a-dEXAMPLEd97b",  
      "resourceName": "MEAN-1-1571075291-Copy",  
      "resourceType": "InstanceSnapshot",  
      "createdAt": 1571075695.069,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-east-1"  
      },  
      "isTerminal": false,  
      "operationDetails": "us-west-2:MEAN-1-1571075291",  
      "operationType": "CopySnapshot",  
      "status": "Started",  
      "statusChangedAt": 1571075695.069  
    }  
  ]  
}
```

Para obtener más información, consulte [Copiar instantáneas de una AWS región a otra en Amazon Lightsail en](#) la Guía para desarrolladores de Lightsail.

Ejemplo 3: Para copiar una instantánea automática dentro de la misma región AWS

El siguiente `copy-snapshot` ejemplo copia la instantánea automática 2019-10-14 de la instancia WordPress-1 como una instantánea manual WordPress-1-10142019 en la AWS región `us-west-2`.

```
aws lightsail copy-snapshot \  
  --source-snapshot-name 2019-10-14  
  --target-snapshot-name WordPress-1-10142019  
  --source-region us-west-2  
  --region us-west-2
```

```

--source-resource-name WordPress-1 \
--restore-date 2019-10-14 \
--target-snapshot-name WordPress-1-10142019 \
--source-region us-west-2

```

Salida:

```

{
  "operations": [
    {
      "id": "be3e6754-cd1d-48e6-ad9f-2EXAMPLE1805",
      "resourceName": "WordPress-1-10142019",
      "resourceType": "InstanceSnapshot",
      "createdAt": 1571082412.311,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationDetails": "us-west-2:WordPress-1",
      "operationType": "CopySnapshot",
      "status": "Started",
      "statusChangedAt": 1571082412.311
    }
  ]
}

```

Para obtener más información, consulte [Mantener instantáneas automáticas de instancias o discos en Amazon Lightsail](#) en la Guía para desarrolladores de Lightsail.

Ejemplo 4: Para copiar una instantánea automática de una región a otra AWS

El siguiente copy-snapshot ejemplo copia la instantánea automática 2019-10-14 de la instancia WordPress-1 como una instantánea manual WordPress-1-10142019 de la AWS región us-west-2 a us-east-1.

```

aws lightsail copy-snapshot \
--source-resource-name WordPress-1 \
--restore-date 2019-10-14 \
--target-snapshot-name WordPress-1-10142019 \
--source-region us-west-2 \
--region us-east-1

```



Salida:

```
{
  "operations": [
    {
      "id": "dfffa128b-0b07-476e-b390-bEXAMPLE3775",
      "resourceName": "WordPress-1-10142019",
      "resourceType": "InstanceSnapshot",
      "createdAt": 1571082493.422,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-east-1"
      },
      "isTerminal": false,
      "operationDetails": "us-west-2:WordPress-1",
      "operationType": "CopySnapshot",
      "status": "Started",
      "statusChangedAt": 1571082493.422
    }
  ]
}
```

Para obtener más información, consulte [Mantener instantáneas automáticas de instancias o discos en Amazon Lightsail en](#) la Guía para desarrolladores de Lightsail.

- Para obtener API más información, consulte la Referencia de comandos. [CopySnapshot](#)AWS CLI

## create-disk-from-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `create-disk-from-snapshot`.

AWS CLI

Para crear y crear un disco a partir de una instantánea de disco

En el siguiente `create-disk-from-snapshot` ejemplo, se crea un disco de almacenamiento en bloques denominado `Disk-2` a partir de la instantánea del disco de almacenamiento en bloque especificada. El disco se crea en la AWS región y la zona de disponibilidad especificadas, con 32 GB de espacio de almacenamiento.

```
aws lightsail create-disk-from-snapshot \
```

```
--disk-name Disk-2 \  
--disk-snapshot-name Disk-1-1566839161 \  
--availability-zone us-west-2a \  
--size-in-gb 32
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "d42b605d-5ef1-4b4a-8791-7a3e8b66b5e7",  
      "resourceName": "Disk-2",  
      "resourceType": "Disk",  
      "createdAt": 1569624941.471,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "CreateDiskFromSnapshot",  
      "status": "Started",  
      "statusChangedAt": 1569624941.791  
    }  
  ]  
}
```

Para obtener más información, consulte [Creación de un disco de almacenamiento en bloque a partir de una instantánea en Amazon Lightsail](#) en la Guía para desarrolladores de Lightsail.

- Para obtener API más información, consulte la Referencia de comandos [CreateDiskFromSnapshot.AWS CLI](#)

## create-disk-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `create-disk-snapshot`.

AWS CLI

Ejemplo 1: Para crear una instantánea de un disco

El siguiente `create-disk-snapshot` ejemplo crea una instantánea con el nombre `DiskSnapshot-1` del disco de almacenamiento en bloque especificado.

```
aws lightsail create-disk-snapshot \  
--disk-name Disk-1 \  
--disk-snapshot-name DiskSnapshot-1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "fa74c6d2-03a3-4f42-a7c7-792f124d534b",  
      "resourceName": "DiskSnapshot-1",  
      "resourceType": "DiskSnapshot",  
      "createdAt": 1569625129.739,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "Disk-1",  
      "operationType": "CreateDiskSnapshot",  
      "status": "Started",  
      "statusChangedAt": 1569625129.739  
    },  
    {  
      "id": "920a25df-185c-4528-87cd-7b85f5488c06",  
      "resourceName": "Disk-1",  
      "resourceType": "Disk",  
      "createdAt": 1569625129.739,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "DiskSnapshot-1",  
      "operationType": "CreateDiskSnapshot",  
      "status": "Started",  
      "statusChangedAt": 1569625129.739  
    }  
  ]  
}
```

Ejemplo 2: Para crear una instantánea del disco del sistema de una instancia

En el siguiente `create-disk-snapshot` ejemplo, se crea una instantánea del disco del sistema de la instancia especificada.

```
aws lightsail create-disk-snapshot \  
  --instance-name WordPress-1 \  
  --disk-snapshot-name SystemDiskSnapshot-1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "f508cf1c-6597-42a6-a4c3-4aebd75af0d9",  
      "resourceName": "SystemDiskSnapshot-1",  
      "resourceType": "DiskSnapshot",  
      "createdAt": 1569625294.685,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "WordPress-1",  
      "operationType": "CreateDiskSnapshot",  
      "status": "Started",  
      "statusChangedAt": 1569625294.685  
    },  
    {  
      "id": "0bb9f712-da3b-4d99-b508-3bf871d989e5",  
      "resourceName": "WordPress-1",  
      "resourceType": "Instance",  
      "createdAt": 1569625294.685,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "SystemDiskSnapshot-1",  
      "operationType": "CreateDiskSnapshot",  
      "status": "Started",  
      "statusChangedAt": 1569625294.685  
    }  
  ]  
}
```

```
}
```

Para obtener más información, consulte [Instantáneas en Amazon Lightsail y Creación de una instantánea del volumen raíz de una instancia en Amazon Lightsail en la Guía para desarrolladores de Lightsail](#).

- Para API obtener más información, consulte la Referencia de comandos. [CreateDiskSnapshot](#) AWS CLI

## create-disk

En el siguiente ejemplo de código, se muestra cómo usar `create-disk`.

### AWS CLI

Para crear un disco de almacenamiento en bloque

En el siguiente `create-disk` ejemplo, se crea un disco de almacenamiento `Disk-1` en bloque en la AWS región y zona de disponibilidad especificadas, con 32 GB de espacio de almacenamiento.

```
aws lightsail create-disk \  
  --disk-name Disk-1 \  
  --availability-zone us-west-2a \  
  --size-in-gb 32
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "1c85e2ec-86ba-4697-b936-77f4d3dc013a",  
      "resourceName": "Disk-1",  
      "resourceType": "Disk",  
      "createdAt": 1569449220.36,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "CreateDisk",  
      "status": "Started",
```

```

        "statusChangedAt": 1569449220.588
      }
    ]
  }

```

- Para API obtener más información, consulte [CreateDisk](#) la Referencia de AWS CLI comandos.

## create-domain-entry

En el siguiente ejemplo de código, se muestra cómo usar `create-domain-entry`.

### AWS CLI

Para crear una entrada de dominio (DNSregistro)

En el siguiente `create-domain-entry` ejemplo, se crea un DNS registro (A) para el vértice del dominio especificado que apunta a la dirección IP de una instancia.

Nota: Las API operaciones relacionadas con el dominio de Lightsail solo están disponibles en la región. `us-east-1` Si su CLI perfil está configurado para usar una región diferente, debe incluir el `--region us-east-1` parámetro o el comando fallará.

```

aws lightsail create-domain-entry \
  --region us-east-1 \
  --domain-name example.com \
  --domain-entry name=example.com,type=A,target=192.0.2.0

```

Salida:

```

{
  "operation": {
    "id": "5be4494d-56f4-41fc-8730-693dcd0ef9e2",
    "resourceName": "example.com",
    "resourceType": "Domain",
    "createdAt": 1569865296.519,
    "location": {
      "availabilityZone": "all",
      "regionName": "global"
    },
    "isTerminal": true,
    "operationType": "CreateDomainEntry",
    "status": "Succeeded",
  }
}

```

```
    "statusChangedAt": 1569865296.519
  }
}
```

Para obtener más información, consulte [Amazon Lightsail y Creación de DNS una zona para gestionar los registros de su dominio DNS en Amazon Lightsail en DNS la Guía para desarrolladores de Lightsail](#).

- Para obtener API más información, consulte la Referencia de comandos. [CreateDomainEntry](#) AWS CLI

## create-domain

En el siguiente ejemplo de código, se muestra cómo usar `create-domain`.

### AWS CLI

Para crear un dominio (DNSzona)

En el siguiente `create-domain` ejemplo, se crea una DNS zona para el dominio especificado.

Nota: Las API operaciones de Lightsail relacionadas con el dominio solo están disponibles en la región. `us-east-1` Si su CLI perfil está configurado para usar una región diferente, debe incluir el `--region us-east-1` parámetro o el comando fallará.

```
aws lightsail create-domain \
  --region us-east-1 \
  --domain-name example.com
```

Salida:

```
{
  "operation": {
    "id": "64e522c8-9ae1-4c05-9b65-3f237324dc34",
    "resourceName": "example.com",
    "resourceType": "Domain",
    "createdAt": 1569864291.92,
    "location": {
      "availabilityZone": "all",
      "regionName": "global"
    },
    "isTerminal": true,
  }
}
```

```
    "operationType": "CreateDomain",
    "status": "Succeeded",
    "statusChangedAt": 1569864292.109
  }
}
```

Para obtener más información, consulte [Amazon Lightsail y Creación de DNS una zona para gestionar los registros de su dominio DNS en Amazon Lightsail en DNS la Guía para desarrolladores de Lightsail](#).

- Para obtener API más información, consulte la Referencia de comandos. [CreateDomain](#)AWS CLI

## create-instance-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `create-instance-snapshot`.

### AWS CLI

Para crear una instantánea de una instancia

En el siguiente `create-instance-snapshot` ejemplo, se crea una instantánea de la instancia especificada.

```
aws lightsail create-instance-snapshot \
  --instance-name WordPress-1 \
  --instance-snapshot-name WordPress-Snapshot-1
```

Salida:

```
{
  "operations": [
    {
      "id": "4c3db559-9dd0-41e7-89c0-2cb88c19786f",
      "resourceName": "WordPress-Snapshot-1",
      "resourceType": "InstanceSnapshot",
      "createdAt": 1569866438.48,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
```



```

    "operationDetails": "WordPress-1",
    "operationType": "CreateInstanceSnapshot",
    "status": "Started",
    "statusChangedAt": 1569866438.48
  },
  {
    "id": "c04fdc45-2981-488c-88b5-d6d2fd759a6a",
    "resourceName": "WordPress-1",
    "resourceType": "Instance",
    "createdAt": 1569866438.48,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "WordPress-Snapshot-1",
    "operationType": "CreateInstanceSnapshot",
    "status": "Started",
    "statusChangedAt": 1569866438.48
  }
]
}

```

- Para API obtener más información, consulte [CreateInstanceSnapshot](#) la Referencia de AWS CLI comandos.

## create-instances-from-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `create-instances-from-snapshot`.

### AWS CLI

Para crear una instancia a partir de una instantánea

En el siguiente `create-instances-from-snapshot` ejemplo, se crea una instancia a partir de la instantánea de la instancia especificada, en la AWS región y zona de disponibilidad especificadas, con el USD paquete de 10\$.

Nota: El paquete que especifiques debe tener especificaciones iguales o superiores al paquete de la instancia de origen original utilizado para crear la instantánea.

```

aws lightsail create-instances-from-snapshot \
  --instance-snapshot-name WordPress-1-1569866208 \

```

```
--instance-names WordPress-2 \  
--availability-zone us-west-2a \  
--bundle-id medium_2_0
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "003f8271-b711-464d-b9b8-7f3806cb496e",  
      "resourceName": "WordPress-2",  
      "resourceType": "Instance",  
      "createdAt": 1569865914.908,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "CreateInstancesFromSnapshot",  
      "status": "Started",  
      "statusChangedAt": 1569865914.908  
    }  
  ]  
}
```

- Para API obtener más información, consulte [CreateInstancesFromSnapshot](#) la Referencia de AWS CLI comandos.

## create-instances

En el siguiente ejemplo de código, se muestra cómo usar `create-instances`.

### AWS CLI

Ejemplo 1: Para crear una sola instancia

En el siguiente `create-instances` ejemplo, se crea una instancia en la AWS región y la zona de disponibilidad especificadas mediante el WordPress blueprint y el paquete de 3,50\$USD.

```
aws lightsail create-instances \  
--instance-names Instance-1 \  
--availability-zone us-west-2a \  
--bundle-id medium_2_0
```

```
--blueprint-id wordpress_5_1_1_2 \  
--bundle-id nano_2_0
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "9a77158f-7be3-4d6d-8054-cf5ae2b720cc",  
      "resourceName": "Instance-1",  
      "resourceType": "Instance",  
      "createdAt": 1569447986.061,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "CreateInstance",  
      "status": "Started",  
      "statusChangedAt": 1569447986.061  
    }  
  ]  
}
```

Ejemplo 2: Para crear varias instancias a la vez

En el siguiente `create-instances` ejemplo, se crean tres instancias en la AWS región y la zona de disponibilidad especificadas mediante el WordPress blueprint y el paquete de 3,50\$USD.

```
aws lightsail create-instances \  
--instance-names {"Instance1","Instance2","Instance3"} \  
--availability-zone us-west-2a \  
--blueprint-id wordpress_5_1_1_2 \  
--bundle-id nano_2_0
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "5492f015-9d2e-48c6-8eea-b516840e6903",  
      "resourceName": "Instance1",  
      "resourceType": "Instance",  
      "createdAt": 1569447986.061,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "CreateInstance",  
      "status": "Started",  
      "statusChangedAt": 1569447986.061  
    }  
  ]  
}
```

```
    "resourceType": "Instance",
    "createdAt": 1569448780.054,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationType": "CreateInstance",
    "status": "Started",
    "statusChangedAt": 1569448780.054
  },
  {
    "id": "c58b5f46-2676-44c8-b95c-3ad375898515",
    "resourceName": "Instance2",
    "resourceType": "Instance",
    "createdAt": 1569448780.054,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationType": "CreateInstance",
    "status": "Started",
    "statusChangedAt": 1569448780.054
  },
  {
    "id": "a5ad8006-9bee-4499-9eb7-75e42e6f5882",
    "resourceName": "Instance3",
    "resourceType": "Instance",
    "createdAt": 1569448780.054,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationType": "CreateInstance",
    "status": "Started",
    "statusChangedAt": 1569448780.054
  }
]
}
```

- Para API obtener más información, consulte la Referencia de [CreateInstances](#) comandos AWS CLI .

## create-key-pair

En el siguiente ejemplo de código, se muestra cómo usar `create-key-pair`.

### AWS CLI

Crear un par de claves

En el siguiente `create-key-pair` ejemplo, se crea un key pair que puedes usar para autenticar y conectarte a una instancia.

```
aws lightsail create-key-pair \
  --key-pair-name MyPersonalKeyPair
```

El resultado proporciona el valor base64 de la clave privada que puede usar para autenticarse en las instancias que usan el par de claves creado. Nota: Copia y pega el valor base64 de la clave privada en una ubicación segura, ya que no podrás recuperarlo más adelante.

```
{
  "keyPair": {
    "name": "MyPersonalKeyPair",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:KeyPair/55025c71-198f-403b-b42f-a69433e724fb",
    "supportCode": "621291663362/MyPersonalKeyPair",
    "createdAt": 1569866556.567,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "KeyPair"
  },
  "publicKeyBase64": "ssh-rsa ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQCV0xUEWx96amPERH7K1bVT1tTF190mNk6o7m5YVHk9x10dMbDRbFvhtXvw4jz
+BHUgedGUXno6uF7agqxZN01kPLJBIVTW26SSYBJ0tE
+y804UyVsjrBUqCaMXDhmfXpWuLMPwuXhwcKh7e8hwoTfkiX0E6Q1
+KqF/MiA3w6DCjEqvvdI07SiEZJFsuGNfYDDN3w60Re15MUhmn30Jdn4y/
A7NWb3IxL4pPVE4rgFRKU8n1jp9kwRnLVMVB0WuGXk6n+H6M2f1 ",
  "privateKeyBase64": "-----BEGIN RSA PRIVATE KEY-----
EXAMPLETCCAfICCQD6m7oRw0uX0jANBgqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
```

```

\nVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6\nnb24xFDASBgNVBA5TC01BTSBD
\nBgkqhkiG9w0BCQEWEG5vb25lQGFTYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
\nMTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
\nVQQHEwdTZWF0dGx1MQ8wDQEXAMPEwZBbWF6b24xFDASBgNVBA5TC01BTSBDb25z
\nb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHhAdBgkqhkiG9w0BCQEWEG5vb25lQGFT
\nYXpvbi5jb20wZGZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMEXAMPLE4GmWIWJ
\n21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
\nrDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
\nIbb30hjZncvQAaREXAMPLEMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4\nnnUHVxYUntneD9+h8Mg9q6q
+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
\nFFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780EXAMPLELvJx79LjStb
\nNYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=\n-----END RSA PRIVATE KEY-----",
  "operation": {
    "id": "67f984db-9994-45fe-ad38-59bafcaf82ef",
    "resourceName": "MyPersonalKeyPair",
    "resourceType": "KeyPair",
    "createdAt": 1569866556.567,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationType": "CreateKeyPair",
    "status": "Succeeded",
    "statusChangedAt": 1569866556.704
  }
}

```

- Para API obtener más información, consulte [CreateKeyPair](#) la Referencia de AWS CLI comandos.

## create-load-balancer-tls-certificate

En el siguiente ejemplo de código, se muestra cómo usar `create-load-balancer-tls-certificate`.

### AWS CLI

Para crear un TLS certificado para un balanceador de cargas

En el siguiente `create-load-balancer-tls-certificate` ejemplo, se crea un TLS certificado que se adjunta al equilibrador de cargas especificado. El certificado creado se aplica a

los dominios especificados. Nota: Solo se pueden crear dos certificados para un balanceador de cargas.

```
aws lightsail create-load-balancer-tls-certificate \  
  --certificate-alternative-names abc.example.com \  
  --certificate-domain-name example.com \  
  --certificate-name MySecondCertificate \  
  --load-balancer-name MyFirstLoadBalancer
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "be663aed-cb46-41e2-9b23-e2f747245bd4",  
      "resourceName": "MySecondCertificate",  
      "resourceType": "LoadBalancerTlsCertificate",  
      "createdAt": 1569867364.971,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationDetails": "MyFirstLoadBalancer",  
      "operationType": "CreateLoadBalancerTlsCertificate",  
      "status": "Succeeded",  
      "statusChangedAt": 1569867365.219  
    },  
    {  
      "id": "f3dfa930-969e-41cc-ac7d-337178716f6d",  
      "resourceName": "MyFirstLoadBalancer",  
      "resourceType": "LoadBalancer",  
      "createdAt": 1569867364.971,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationDetails": "MySecondCertificate",  
      "operationType": "CreateLoadBalancerTlsCertificate",  
      "status": "Succeeded",  
      "statusChangedAt": 1569867365.219  
    }  
  ]  
}
```

```
]
}
```

- Para API obtener más información, consulte [CreateLoadBalancerTlsCertificate](#) la Referencia de AWS CLI comandos.

## create-load-balancer

En el siguiente ejemplo de código, se muestra cómo usar `create-load-balancer`.

### AWS CLI

Para crear un balanceador de cargas

En el siguiente `create-load-balancer` ejemplo, se crea un equilibrador de carga con un TLS certificado. El TLS certificado se aplica a los dominios especificados y enruta el tráfico a las instancias del puerto 80.

```
aws lightsail create-load-balancer \
  --certificate-alternative-names www.example.com test.example.com \
  --certificate-domain-name example.com \
  --certificate-name Certificate-1 \
  --instance-port 80 \
  --load-balancer-name LoadBalancer-1
```

Salida:

```
{
  "operations": [
    {
      "id": "cc7b920a-83d8-4762-a74e-9174fe1540be",
      "resourceName": "LoadBalancer-1",
      "resourceType": "LoadBalancer",
      "createdAt": 1569867169.406,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationType": "CreateLoadBalancer",
      "status": "Started",
```



```

    "statusChangedAt": 1569867169.406
  },
  {
    "id": "658ed43b-f729-42f3-a8e4-3f8024d3c98d",
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancerTlsCertificate",
    "createdAt": 1569867170.193,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "LoadBalancer-1",
    "operationType": "CreateLoadBalancerTlsCertificate",
    "status": "Succeeded",
    "statusChangedAt": 1569867170.54
  },
  {
    "id": "4757a342-5181-4870-b1e0-227eebc35ab5",
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancer",
    "createdAt": 1569867170.193,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "Certificate-1",
    "operationType": "CreateLoadBalancerTlsCertificate",
    "status": "Succeeded",
    "statusChangedAt": 1569867170.54
  }
]
}

```

Para obtener más información, consulte los balanceadores de [carga de Lightsail](#) en la Guía para desarrolladores de Lightsail.

- Para obtener API más información, consulte la Referencia de comandos.

[CreateLoadBalancer](#) AWS CLI

## create-relational-database-from-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `create-relational-database-from-snapshot`.

### AWS CLI

Para crear una base de datos gestionada a partir de una instantánea

El siguiente `create-relational-database-from-snapshot` ejemplo crea una base de datos administrada a partir de la instantánea especificada en la AWS región y zona de disponibilidad especificadas, utilizando el paquete de base de datos USD estándar de 15 dólares. Nota: El paquete que especifique debe tener especificaciones iguales o superiores al paquete de la base de datos de origen original utilizado para crear la instantánea.

```
aws lightsail create-relational-database-from-snapshot \  
  --relational-database-snapshot-name Database-Oregon-1-1566839359 \  
  --relational-database-name Database-1 \  
  --availability-zone us-west-2a \  
  --relational-database-bundle-id micro_1_0 \  
  --no-publicly-accessible
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "ad6d9193-9d5c-4ea1-97ae-8fe6de600b4c",  
      "resourceName": "Database-1",  
      "resourceType": "RelationalDatabase",  
      "createdAt": 1569867916.938,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "CreateRelationalDatabaseFromSnapshot",  
      "status": "Started",  
      "statusChangedAt": 1569867918.643  
    }  
  ]  
}
```

- Para API obtener más información, consulte [CreateRelationalDatabaseFromSnapshot](#) la Referencia de AWS CLI comandos.

## create-relational-database-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `create-relational-database-snapshot`.

### AWS CLI

Para crear una instantánea de una base de datos gestionada

En el siguiente `create-relational-database-snapshot` ejemplo, se crea una instantánea de la base de datos gestionada especificada.

```
aws lightsail create-relational-database-snapshot \  
  --relational-database-name Database1 \  
  --relational-database-snapshot-name RelationalDatabaseSnapshot1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "853667fb-ea91-4c02-8d20-8fc5fd43b9eb",  
      "resourceName": "RelationalDatabaseSnapshot1",  
      "resourceType": "RelationalDatabaseSnapshot",  
      "createdAt": 1569868074.645,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "Database1",  
      "operationType": "CreateRelationalDatabaseSnapshot",  
      "status": "Started",  
      "statusChangedAt": 1569868074.645  
    },  
    {  
      "id": "fbafa521-3cac-4be8-9773-1c143780b239",  
      "resourceName": "Database1",  
      "resourceType": "RelationalDatabase",
```

```
    "createdAt": 1569868074.645,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "RelationalDatabaseSnapshot1",
    "operationType": "CreateRelationalDatabaseSnapshot",
    "status": "Started",
    "statusChangedAt": 1569868074.645
  }
]
}
```

- Para API obtener más información, consulte [CreateRelationalDatabaseSnapshot](#) la Referencia de AWS CLI comandos.

## create-relational-database

En el siguiente ejemplo de código, se muestra cómo usar `create-relational-database`.

### AWS CLI

Para crear una base de datos gestionada

En el siguiente `create-relational-database` ejemplo, se crea una base de datos gestionada en la AWS región y zona de disponibilidad especificadas, mediante el motor de base de datos My SQL 5.6 (`mysql_5_6`) y el paquete de base de datos USD estándar de 15 dólares (`micro_1_0`). La base de datos gestionada viene rellena previamente con un nombre de usuario maestro y no es de acceso público.

```
aws lightsail create-relational-database \
  --relational-database-name Database-1 \
  --availability-zone us-west-2a \
  --relational-database-blueprint-id mysql_5_6 \
  --relational-database-bundle-id micro_1_0 \
  --master-database-name dbmaster \
  --master-username user \
  --no-publicly-accessible
```

Salida:

```
{
  "operations": [
    {
      "id": "b52bedee-73ed-4798-8d2a-9c12df89adcd",
      "resourceName": "Database-1",
      "resourceType": "RelationalDatabase",
      "createdAt": 1569450017.244,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationType": "CreateRelationalDatabase",
      "status": "Started",
      "statusChangedAt": 1569450018.637
    }
  ]
}
```

- Para API obtener más información, consulte [CreateRelationalDatabase](#) la Referencia de AWS CLI comandos.

## delete-auto-snapshot

En el siguiente ejemplo de código, se muestra cómo usar delete-auto-snapshot.

### AWS CLI

Para eliminar una instantánea automática

El siguiente delete-auto-snapshot ejemplo elimina la instantánea automática 2019-10-10 de la instanciaWordPress-1.

```
aws lightsail delete-auto-snapshot \
  --resource-name WordPress-1 \
  --date 2019-10-10
```

Salida:

```
{
  "operations": [
    {
```

```

    "id": "31c36e09-3d52-46d5-b6d8-7EXAMPLE534a",
    "resourceName": "WordPress-1",
    "resourceType": "Instance",
    "createdAt": 1571088141.501,
    "location": {
      "availabilityZone": "us-west-2",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "DeleteAutoSnapshot-2019-10-10",
    "operationType": "DeleteAutoSnapshot",
    "status": "Succeeded"
  }
]
}

```

Para obtener más información, consulte [Eliminar instantáneas automáticas de instancias o discos en Amazon Lightsail en](#) la Guía para desarrolladores de Lightsail.

- Para obtener API más información, consulte la Referencia de comandos. [DeleteAutoSnapshot](#) AWS CLI

## delete-disk-snapshot

En el siguiente ejemplo de código, se muestra cómo usar delete-disk-snapshot.

### AWS CLI

Para eliminar una instantánea de un disco de almacenamiento en bloque

En el siguiente delete-disk-snapshot ejemplo, se elimina la instantánea especificada de un disco de almacenamiento en bloque

```

aws lightsail delete-disk-snapshot \
  --disk-snapshot-name DiskSnapshot-1

```

Salida:

```

{
  "operations": [
    {
      "id": "d1e5766d-b81e-4595-ad5d-02afbcccfd5d",
      "resourceName": "DiskSnapshot-1",

```

```
    "resourceType": "DiskSnapshot",
    "createdAt": 1569873552.79,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationType": "DeleteDiskSnapshot",
    "status": "Succeeded",
    "statusChangedAt": 1569873552.79
  }
]
}
```

- Para API obtener más información, consulte [DeleteDiskSnapshot](#) la Referencia de AWS CLI comandos.

## delete-disk

En el siguiente ejemplo de código, se muestra cómo usar `delete-disk`.

### AWS CLI

Para eliminar un disco de almacenamiento en bloque

En el siguiente `delete-disk` ejemplo, se elimina el disco de almacenamiento en bloque especificado.

```
aws lightsail delete-disk \  
  --disk-name Disk-1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "6378c70f-4d75-4f7a-ab66-730fca0bb2fc",  
      "resourceName": "Disk-1",  
      "resourceType": "Disk",  
      "createdAt": 1569872887.864,  
      "location": {  
        "availabilityZone": "us-west-2a",
```

```

        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationType": "DeleteDisk",
      "status": "Succeeded",
      "statusChangedAt": 1569872887.864
    }
  ]
}

```

- Para API obtener más información, consulte [DeleteDisk](#) la Referencia de AWS CLI comandos.

## delete-domain-entry

En el siguiente ejemplo de código, se muestra cómo usar `delete-domain-entry`.

### AWS CLI

Para eliminar una entrada de dominio (DNSregistro)

En el siguiente `delete-domain-entry` ejemplo, se elimina la entrada de dominio especificada de un dominio existente.

Nota: Las API operaciones de Lightsail relacionadas con el dominio solo están disponibles en la región. `us-east-1` Si su CLI perfil está configurado para usar una región diferente, debe incluir el `--region us-east-1` parámetro o el comando fallará.

```

aws lightsail delete-domain-entry \
  --region us-east-1 \
  --domain-name example.com \
  --domain-entry name=123.example.com,target=192.0.2.0,type=A

```

Salida:

```

{
  "operation": {
    "id": "06eacd01-d785-420e-8daa-823150c7dca1",
    "resourceName": "example.com ",
    "resourceType": "Domain",
    "createdAt": 1569874157.005,
    "location": {
      "availabilityZone": "all",

```



```
        "regionName": "global"
    },
    "isTerminal": true,
    "operationType": "DeleteDomainEntry",
    "status": "Succeeded",
    "statusChangedAt": 1569874157.005
}
}
```

- Para API obtener más información, consulte [DeleteDomainEntry](#) la Referencia de AWS CLI comandos.

## delete-domain

En el siguiente ejemplo de código, se muestra cómo usar `delete-domain`.

### AWS CLI

Para eliminar un dominio (DNSzona)

El siguiente `delete-domain` ejemplo elimina el dominio especificado y todas las entradas del dominio (DNSregistros).

Nota: Las API operaciones relacionadas con el dominio de Lightsail solo están disponibles en la región. `us-east-1` Si su CLI perfil está configurado para usar una región diferente, debe incluir el `--region us-east-1` parámetro o el comando fallará.

```
aws lightsail delete-domain \
  --region us-east-1 \
  --domain-name example.com
```

Salida:

```
{
  "operation": {
    "id": "fcef5265-5af1-4a46-a3d7-90b5e18b9b32",
    "resourceName": "example.com",
    "resourceType": "Domain",
    "createdAt": 1569873788.13,
    "location": {
      "availabilityZone": "all",
      "regionName": "global"
    }
  }
}
```

```
    },
    "isTerminal": true,
    "operationType": "DeleteDomain",
    "status": "Succeeded",
    "statusChangedAt": 1569873788.13
  }
}
```

- Para API obtener más información, consulte [DeleteDomain](#) la Referencia de AWS CLI comandos.

## delete-instance-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `delete-instance-snapshot`.

AWS CLI

title

En el siguiente `delete-instance-snapshot` ejemplo, se elimina la instantánea especificada de una instancia.

```
aws lightsail delete-instance-snapshot \
  --instance-snapshot-name WordPress-1-Snapshot-1
```

Salida:

```
{
  "operations": [
    {
      "id": "14dad182-976a-46c6-bfd4-9480482bf0ea",
      "resourceName": "WordPress-1-Snapshot-1",
      "resourceType": "InstanceSnapshot",
      "createdAt": 1569874524.562,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationType": "DeleteInstanceSnapshot",
      "status": "Succeeded",
      "statusChangedAt": 1569874524.562
    }
  ]
}
```

```
    }  
  ]  
}
```

- Para API obtener más información, consulta [DeleteInstanceSnapshot](#) la Referencia de AWS CLI comandos.

## delete-instance

En el siguiente ejemplo de código, se muestra cómo usar `delete-instance`.

### AWS CLI

Para eliminar una instancia

En el siguiente `delete-instance` ejemplo, se elimina la instancia especificada.

```
aws lightsail delete-instance \  
  --instance-name WordPress-1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "d77345a3-8f80-4d2e-b47d-aaa622718df2",  
      "resourceName": "Disk-1",  
      "resourceType": "Disk",  
      "createdAt": 1569874357.469,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "WordPress-1",  
      "operationType": "DetachDisk",  
      "status": "Started",  
      "statusChangedAt": 1569874357.469  
    },  
    {  
      "id": "708fa606-2bfd-4e48-a2c1-0b856585b5b1",  
      "resourceName": "WordPress-1",
```

```

    "resourceType": "Instance",
    "createdAt": 1569874357.465,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "Disk-1",
    "operationType": "DetachDisk",
    "status": "Started",
    "statusChangedAt": 1569874357.465
  },
  {
    "id": "3187e823-8acb-405d-b098-fad5ceb17bec",
    "resourceName": "WordPress-1",
    "resourceType": "Instance",
    "createdAt": 1569874357.829,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationType": "DeleteInstance",
    "status": "Succeeded",
    "statusChangedAt": 1569874357.829
  }
]
}

```

- Para API obtener más información, consulte [DeleteInstance](#) la Referencia de AWS CLI comandos.

## delete-key-pair

En el siguiente ejemplo de código, se muestra cómo usar `delete-key-pair`.

### AWS CLI

Eliminar un par de claves

En el siguiente `delete-key-pair` ejemplo, se elimina el key pair especificado.

```
aws lightsail delete-key-pair \
```

```
--key-pair-name MyPersonalKeyPair
```

Salida:

```
{
  "operation": {
    "id": "81621463-df38-4810-b866-6e801a15abbf",
    "resourceName": "MyPersonalKeyPair",
    "resourceType": "KeyPair",
    "createdAt": 1569874626.466,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationType": "DeleteKeyPair",
    "status": "Succeeded",
    "statusChangedAt": 1569874626.685
  }
}
```

- Para API obtener más información, consulte [DeleteKeyPair](#) la Referencia de AWS CLI comandos.

## delete-known-host-keys

En el siguiente ejemplo de código, se muestra cómo usar `delete-known-host-keys`.

AWS CLI

Para eliminar claves de host conocidas de una instancia

En el siguiente `delete-known-host-keys` ejemplo, se elimina la clave de host conocida de la instancia especificada.

```
aws lightsail delete-known-host-keys \  
  --instance-name Instance-1
```

Salida:

```
{
  "operations": [
```

```
{
  "id": "c61afe9c-45a4-41e6-a97e-d212364da3f5",
  "resourceName": "Instance-1",
  "resourceType": "Instance",
  "createdAt": 1569874760.201,
  "location": {
    "availabilityZone": "us-west-2a",
    "regionName": "us-west-2"
  },
  "isTerminal": true,
  "operationType": "DeleteKnownHostKeys",
  "status": "Succeeded",
  "statusChangedAt": 1569874760.201
}
]
```

Para obtener más información, consulte [Solución de problemas de conexión con el RDP cliente SSH o el navegador Amazon Lightsail](#) en la Guía para desarrolladores de Lightsail.

- Para obtener API más información, consulte la Referencia de comandos. [DeleteKnownHostKeys](#) AWS CLI

## delete-load-balancer-tls-certificate

En el siguiente ejemplo de código, se muestra cómo usar `delete-load-balancer-tls-certificate`.

### AWS CLI

Para eliminar un TLS certificado de un balanceador de cargas

En el siguiente `delete-load-balancer-tls-certificate` ejemplo, se elimina el TLS certificado especificado del equilibrador de cargas especificado.

```
aws lightsail delete-load-balancer-tls-certificate \
  --load-balancer-name MyFirstLoadBalancer \
  --certificate-name MyFirstCertificate
```

Salida:

```
{
```

```

"operations": [
  {
    "id": "50bec274-e45e-4caa-8a69-b763ef636583",
    "resourceName": "MyFirstCertificate",
    "resourceType": "LoadBalancerTlsCertificate",
    "createdAt": 1569874989.48,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationType": "DeleteLoadBalancerTlsCertificate",
    "status": "Started",
    "statusChangedAt": 1569874989.48
  },
  {
    "id": "78c58cdc-a59a-4b27-8213-500638634a8f",
    "resourceName": "MyFirstLoadBalancer",
    "resourceType": "LoadBalancer",
    "createdAt": 1569874989.48,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationType": "DeleteLoadBalancerTlsCertificate",
    "status": "Started",
    "statusChangedAt": 1569874989.48
  }
]
}

```

- Para API obtener más información, consulte la Referencia de [DeleteLoadBalancerTlsCertificate](#) comandos AWS CLI .

## delete-load-balancer

En el siguiente ejemplo de código, se muestra cómo usar delete-load-balancer.

### AWS CLI

#### Eliminación de un equilibrador de carga

En el siguiente delete-load-balancer ejemplo, se elimina el balanceador de cargas especificado y todos los certificados asociados TLS.

```
aws lightsail delete-load-balancer \  
--load-balancer-name MyFirstLoadBalancer
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "a8c968c7-72a3-4680-a714-af8f03eea535",  
      "resourceName": "MyFirstLoadBalancer",  
      "resourceType": "LoadBalancer",  
      "createdAt": 1569875092.125,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationType": "DeleteLoadBalancer",  
      "status": "Succeeded",  
      "statusChangedAt": 1569875092.125  
    },  
    {  
      "id": "f91a29fc-8ce3-4e69-a227-ea70ca890bf5",  
      "resourceName": "MySecondCertificate",  
      "resourceType": "LoadBalancerTlsCertificate",  
      "createdAt": 1569875091.938,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationType": "DeleteLoadBalancerTlsCertificate",  
      "status": "Started",  
      "statusChangedAt": 1569875091.938  
    },  
    {  
      "id": "cf64c060-154b-4eb4-ba57-84e2e41563d6",  
      "resourceName": "MyFirstLoadBalancer",  
      "resourceType": "LoadBalancer",  
      "createdAt": 1569875091.94,
```



```

        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "isTerminal": false,
        "operationType": "DeleteLoadBalancerTlsCertificate",
        "status": "Started",
        "statusChangedAt": 1569875091.94
    }
]
}

```

Para obtener más información, consulta el título de la guía.

- Para API obtener más información, consulte [DeleteLoadBalancer](#) la Referencia de AWS CLI comandos.

## delete-relational-database-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `delete-relational-database-snapshot`.

### AWS CLI

Para eliminar una instantánea de una base de datos gestionada

En el siguiente `delete-relational-database-snapshot` ejemplo, se elimina la instantánea especificada de una base de datos gestionada.

```

aws lightsail delete-relational-database-snapshot \
  --relational-database-snapshot-name Database-Oregon-1-1566839359

```

Salida:

```

{
  "operations": [
    {
      "id": "b99acae8-735b-4823-922f-30af580e3729",
      "resourceName": "Database-Oregon-1-1566839359",
      "resourceType": "RelationalDatabaseSnapshot",
      "createdAt": 1569875293.58,
      "location": {

```

```
        "availabilityZone": "all",
        "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationType": "DeleteRelationalDatabaseSnapshot",
    "status": "Succeeded",
    "statusChangedAt": 1569875293.58
  }
]
}
```

- Para API obtener más información, consulte [DeleteRelationalDatabaseSnapshot](#) la Referencia de AWS CLI comandos.

## delete-relational-database

En el siguiente ejemplo de código, se muestra cómo usar `delete-relational-database`.

### AWS CLI

Para eliminar una base de datos gestionada

En el siguiente `delete-relational-database` ejemplo, se elimina la base de datos gestionada especificada.

```
aws lightsail delete-relational-database \
  --relational-database-name Database-1
```

Salida:

```
{
  "operations": [
    {
      "id": "3b0c41c1-053d-46f0-92a3-14f76141dc86",
      "resourceName": "Database-1",
      "resourceType": "RelationalDatabase",
      "createdAt": 1569875210.999,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
```

```

    "operationType": "DeleteRelationalDatabase",
    "status": "Started",
    "statusChangedAt": 1569875210.999
  },
  {
    "id": "01ddeae8-a87a-4a4b-a1f3-092c71bf9180",
    "resourceName": "Database-1",
    "resourceType": "RelationalDatabase",
    "createdAt": 1569875211.029,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "Database-1-FinalSnapshot-1569875210793",
    "operationType": "CreateRelationalDatabaseSnapshot",
    "status": "Started",
    "statusChangedAt": 1569875211.029
  },
  {
    "id": "74d73681-30e8-4532-974e-1f23cd3f9f73",
    "resourceName": "Database-1-FinalSnapshot-1569875210793",
    "resourceType": "RelationalDatabaseSnapshot",
    "createdAt": 1569875211.029,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationDetails": "Database-1",
    "operationType": "CreateRelationalDatabaseSnapshot",
    "status": "Started",
    "statusChangedAt": 1569875211.029
  }
]
}

```

- Para API obtener más información, consulte [DeleteRelationalDatabase](#) la Referencia de AWS CLI comandos.

## detach-static-ip

En el siguiente ejemplo de código, se muestra cómo usar detach-static-ip.

## AWS CLI

Para separar una IP estática de una instancia

En el siguiente `detach-static-ip` ejemplo, se separa la IP estática `StaticIp-1` de cualquier instancia conectada.

```
aws lightsail detach-static-ip \  
  --static-ip-name StaticIp-1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "2a43d8a3-9f2d-4fe7-bdd0-eEXAMPLE3cf3",  
      "resourceName": "StaticIp-1",  
      "resourceType": "StaticIp",  
      "createdAt": 1571088261.999,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationDetails": "MEAN-1",  
      "operationType": "DetachStaticIp",  
      "status": "Succeeded",  
      "statusChangedAt": 1571088261.999  
    },  
    {  
      "id": "41a7d40c-74e8-4d2e-a837-cEXAMPLEf747",  
      "resourceName": "MEAN-1",  
      "resourceType": "Instance",  
      "createdAt": 1571088262.022,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationDetails": "StaticIp-1",  
      "operationType": "DetachStaticIp",  
      "status": "Succeeded",  
      "statusChangedAt": 1571088262.022  
    }  
  ]  
}
```

```
    }  
  ]  
}
```

- Para API obtener más información, consulte [DetachStaticIp](#) la Referencia de AWS CLI comandos.

## get-active-names

En el siguiente ejemplo de código, se muestra cómo usar `get-active-names`.

### AWS CLI

Para obtener los nombres de los recursos activos

El siguiente `get-active-names` ejemplo devuelve los nombres de los recursos activos de la AWS región configurada.

```
aws lightsail get-active-names
```

Salida:

```
{  
  "activeNames": [  
    "WordPress-1",  
    "StaticIp-1",  
    "MEAN-1",  
    "Plesk_Hosting_Stack_on_Ubuntu-1"  
  ]  
}
```

- Para API obtener más información, consulte [GetActiveNames](#) la Referencia de AWS CLI comandos.

## get-auto-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `get-auto-snapshots`.

### AWS CLI

Para obtener las instantáneas automáticas disponibles para una instancia

En el siguiente `get-auto-snapshots` ejemplo, se devuelven las instantáneas automáticas disponibles, por ejemplo. `WordPress-1`

```
aws lightsail get-auto-snapshots \  
  --resource-name WordPress-1
```

Salida:

```
{  
  "resourceName": "WordPress-1",  
  "resourceType": "Instance",  
  "autoSnapshots": [  
    {  
      "date": "2019-10-14",  
      "createdAt": 1571033872.0,  
      "status": "Success",  
      "fromAttachedDisks": []  
    },  
    {  
      "date": "2019-10-13",  
      "createdAt": 1570947473.0,  
      "status": "Success",  
      "fromAttachedDisks": []  
    },  
    {  
      "date": "2019-10-12",  
      "createdAt": 1570861072.0,  
      "status": "Success",  
      "fromAttachedDisks": []  
    },  
    {  
      "date": "2019-10-11",  
      "createdAt": 1570774672.0,  
      "status": "Success",  
      "fromAttachedDisks": []  
    }  
  ]  
}
```

Para obtener más información, consulte [Mantener instantáneas automáticas de instancias o discos en Amazon Lightsail](#) en la Guía para desarrolladores de Lightsail.

- Para obtener API más información, consulte la Referencia de comandos.  
[GetAutoSnapshots](#)AWS CLI

## get-blueprints

En el siguiente ejemplo de código, se muestra cómo usar `get-blueprints`.

AWS CLI

Para obtener los planos de las nuevas instancias

En el siguiente `get-blueprints` ejemplo, se muestran detalles sobre todos los blueprints disponibles que se pueden usar para crear nuevas instancias en Amazon Lightsail.

```
aws lightsail get-blueprints
```

Salida:

```
{
  "blueprints": [
    {
      "blueprintId": "wordpress",
      "name": "WordPress",
      "group": "wordpress",
      "type": "app",
      "description": "Bitnami, the leaders in application packaging, and Automattic, the experts behind WordPress, have teamed up to offer this official WordPress image. This image is a pre-configured, ready-to-run image for running WordPress on Amazon Lightsail. WordPress is the world's most popular content management platform. Whether it's for an enterprise or small business website, or a personal or corporate blog, content authors can easily create content using its new Gutenberg editor, and developers can extend the base platform with additional features. Popular plugins like Jetpack, Akismet, All in One SEO Pack, WP Mail, Google Analytics for WordPress, and Amazon Polly are all pre-installed in this image. Let's Encrypt SSL certificates are supported through an auto-configuration script.",
      "isActive": true,
      "minPower": 0,
      "version": "5.2.2-3",
      "versionCode": "1",
      "productUrl": "https://aws.amazon.com/marketplace/pp/B00NN8Y43U",
```

```
    "licenseUrl": "https://d7umqicpi7263.cloudfront.net/eula/
product/7d426cb7-9522-4dd7-a56b-55dd8cc1c8d0/588fd495-6492-4610-b3e8-
d15ce864454c.txt",
    "platform": "LINUX_UNIX"
  },
  {
    "blueprintId": "lamp_7_1_28",
    "name": "LAMP (PHP 7)",
    "group": "lamp_7",
    "type": "app",
    "description": "LAMP with PHP 7.x certified by Bitnami greatly
simplifies the development and deployment of PHP applications. It includes the
latest versions of PHP 7.x, Apache and MySQL together with phpMyAdmin and popular
PHP frameworks Zend, Symfony, CodeIgniter, CakePHP, Smarty, and Laravel. Other pre-
configured components and PHP modules include FastCGI, ModSecurity, SQLite, Varnish,
ImageMagick, xDebug, Xcache, OpenLDAP, Memcache, OAuth, PEAR, PECL, APC, GD and
cURL. It is secure by default and supports multiple applications, each with its own
virtual host and project directory. Let's Encrypt SSL certificates are supported
through an auto-configuration script.",
    "isActive": true,
    "minPower": 0,
    "version": "7.1.28",
    "versionCode": "1",
    "productUrl": "https://aws.amazon.com/marketplace/pp/B072JNJZ5C",
    "licenseUrl": "https://d7umqicpi7263.cloudfront.net/eula/product/
cb6afd05-a3b2-4916-a3e6-bccd414f5f21/12ab56cc-6a8c-4977-9611-dcd770824aad.txt",
    "platform": "LINUX_UNIX"
  },
  {
    "blueprintId": "nodejs",
    "name": "Node.js",
    "group": "node",
    "type": "app",
    "description": "Node.js certified by Bitnami is a pre-configured, ready
to run image for Node.js on Amazon EC2. It includes the latest version of Node.js,
Apache, Python and Redis. The image supports multiple Node.js applications, each
with its own virtual host and project directory. It is configured for production
use and is secure by default, as all ports except HTTP, HTTPS and SSH ports are
closed. Let's Encrypt SSL certificates are supported through an auto-configuration
script. Developers benefit from instant access to a secure, update and consistent
Node.js environment without having to manually install and configure multiple
components and libraries.",
    "isActive": true,
    "minPower": 0,
```



```

        "version": "12.7.0",
        "versionCode": "1",
        "productUrl": "https://aws.amazon.com/marketplace/pp/B00NNZUAK0",
        "licenseUrl": "https://d7umqicpi7263.cloudfront.net/
eula/product/033793fe-951d-47d0-aa94-5fbd0afb3582/25f8fa66-c868-4d80-
adf8-4a2b602064ae.txt",
        "platform": "LINUX_UNIX"
    },
    ...
}
]
}

```

- Para API obtener más información, consulte la Referencia de [GetBlueprints](#) comandos AWS CLI .

## get-bundles

En el siguiente ejemplo de código, se muestra cómo usar `get-bundles`.

### AWS CLI

Para obtener los paquetes de nuevas instancias

En el siguiente `get-bundles` ejemplo, se muestran detalles sobre todos los paquetes disponibles que se pueden usar para crear nuevas instancias en Amazon Lightsail.

```
aws lightsail get-bundles
```

Salida:

```

{
  "bundles": [
    {
      "price": 3.5,
      "cpuCount": 1,
      "diskSizeInGb": 20,
      "bundleId": "nano_2_0",
      "instanceType": "nano",
      "isActive": true,
      "name": "Nano",
    }
  ]
}

```

```
    "power": 300,
    "ramSizeInGb": 0.5,
    "transferPerMonthInGb": 1024,
    "supportedPlatforms": [
      "LINUX_UNIX"
    ]
  },
  {
    "price": 5.0,
    "cpuCount": 1,
    "diskSizeInGb": 40,
    "bundleId": "micro_2_0",
    "instanceType": "micro",
    "isActive": true,
    "name": "Micro",
    "power": 500,
    "ramSizeInGb": 1.0,
    "transferPerMonthInGb": 2048,
    "supportedPlatforms": [
      "LINUX_UNIX"
    ]
  },
  {
    "price": 10.0,
    "cpuCount": 1,
    "diskSizeInGb": 60,
    "bundleId": "small_2_0",
    "instanceType": "small",
    "isActive": true,
    "name": "Small",
    "power": 1000,
    "ramSizeInGb": 2.0,
    "transferPerMonthInGb": 3072,
    "supportedPlatforms": [
      "LINUX_UNIX"
    ]
  },
  ...
}
]
```

- Para API obtener más información, consulte la Referencia de [GetBundles](#) comandos AWS CLI .

## get-cloud-formation-stack-records

En el siguiente ejemplo de código, se muestra cómo usar `get-cloud-formation-stack-records`.

### AWS CLI

Para obtener los registros de la CloudFormation pila y sus pilas asociadas

El siguiente `get-cloud-formation-stack-records` ejemplo muestra detalles sobre los registros de CloudFormation pila y sus pilas asociadas que se utilizan para crear EC2 recursos de Amazon a partir de instantáneas de Amazon Lightsail exportadas.

```
aws lightsail get-cloud-formation-stack-records
```

Salida:

```
{
  "cloudFormationStackRecords": [
    {
      "name": "CloudFormationStackRecord-588a4243-
e2d1-490d-8200-3a7513ecebdf",
      "arn": "arn:aws:lightsail:us-
west-2:111122223333:CloudFormationStackRecord/28d646ab-27bc-48d9-a422-1EXAMPLE6d37",
      "createdAt": 1565301666.586,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "CloudFormationStackRecord",
      "state": "Succeeded",
      "sourceInfo": [
        {
          "resourceType": "ExportSnapshotRecord",
          "name": "ExportSnapshotRecord-
e02f23d7-0453-4aa9-9c95-91aa01a141dd",
          "arn": "arn:aws:lightsail:us-
west-2:111122223333:ExportSnapshotRecord/f12b8792-f3ea-4d6f-b547-2EXAMPLE8796"
        }
      ],
      "destinationInfo": {
```

```

        "id": "arn:aws:cloudformation:us-west-2:111122223333:stack/
Lightsail-Stack-588a4243-e2d1-490d-8200-3EXAMPLEebdf/063203b0-
ba28-11e9-838b-0EXAMPLE8b00",
        "service": "Aws::CloudFormation::Stack"
    }
}
]
}

```

- Para obtener API más información, consulte la Referencia de comandos [GetCloudFormationStackRecords](#).AWS CLI

## get-disk-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `get-disk-snapshot`.

### AWS CLI

Para obtener información sobre una instantánea de disco

En el siguiente `get-disk-snapshot` ejemplo, se muestran detalles sobre la instantánea del disco `Disk-1-1566839161`.

```

aws lightsail get-disk-snapshot \
  --disk-snapshot-name Disk-1-1566839161

```

Salida:

```

{
  "diskSnapshot": {
    "name": "Disk-1-1566839161",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:DiskSnapshot/
e2d0fa53-8ee0-41a0-8e56-0EXAMPLE1051",
    "supportCode": "6EXAMPLE3362/snap-0EXAMPLE06100d09",
    "createdAt": 1566839163.749,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "DiskSnapshot",
    "tags": [],
  }
}

```

```
    "sizeInGb": 8,  
    "state": "completed",  
    "progress": "100%",  
    "fromDiskName": "Disk-1",  
    "fromDiskArn": "arn:aws:lightsail:us-west-2:111122223333:Disk/  
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",  
    "isFromAutoSnapshot": false  
  }  
}
```

Para obtener más información, consulte el título de la guía.

- Para API obtener más información, consulte [GetDiskSnapshot](#) la Referencia de AWS CLI comandos.

## get-disk-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `get-disk-snapshots`.

### AWS CLI

Para obtener información sobre todas las instantáneas de disco

En el siguiente `get-disk-snapshots` ejemplo, se muestran detalles sobre todas las instantáneas de disco de la región configurada AWS .

```
aws lightsail get-disk-snapshots
```

Salida:

```
{  
  "diskSnapshots": [  
    {  
      "name": "Disk-2-1571090588",  
      "arn": "arn:aws:lightsail:us-  
west-2:111122223333:DiskSnapshot/32e889a9-38d4-4687-9f21-eEXAMPLE7839",  
      "supportCode": "6EXAMPLE3362/snap-0EXAMPLE1ca192a4",  
      "createdAt": 1571090591.226,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
    },  
  ],  
}
```

```

    "resourceType": "DiskSnapshot",
    "tags": [],
    "sizeInGb": 8,
    "state": "completed",
    "progress": "100%",
    "fromDiskName": "Disk-2",
    "fromDiskArn": "arn:aws:lightsail:us-
west-2:111122223333:Disk/6a343ff8-6341-422d-86e2-bEXAMPLE16c2",
    "isFromAutoSnapshot": false
  },
  {
    "name": "Disk-1-1566839161",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:DiskSnapshot/
e2d0fa53-8ee0-41a0-8e56-0EXAMPLE1051",
    "supportCode": "6EXAMPLE3362/snap-0EXAMPLEe06100d09",
    "createdAt": 1566839163.749,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "DiskSnapshot",
    "tags": [],
    "sizeInGb": 8,
    "state": "completed",
    "progress": "100%",
    "fromDiskName": "Disk-1",
    "fromDiskArn": "arn:aws:lightsail:us-west-2:111122223333:Disk/
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",
    "isFromAutoSnapshot": false
  }
]
}

```

- Para API obtener más información, consulte [GetDiskSnapshots](#) la Referencia de AWS CLI comandos.

## get-disk

En el siguiente ejemplo de código, se muestra cómo usar `get-disk`.

### AWS CLI

Para obtener información sobre un disco de almacenamiento en bloque

En el siguiente `get-disk` ejemplo, se muestran detalles sobre el disco `Disk-1`.

```
aws lightsail get-disk \  
  --disk-name Disk-1
```

Salida:

```
{  
  "disk": {  
    "name": "Disk-1",  
    "arn": "arn:aws:lightsail:us-west-2:111122223333:Disk/  
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",  
    "supportCode": "6EXAMPLE3362/vol-0EXAMPLEf2f88b32f",  
    "createdAt": 1566585439.587,  
    "location": {  
      "availabilityZone": "us-west-2a",  
      "regionName": "us-west-2"  
    },  
    "resourceType": "Disk",  
    "tags": [],  
    "sizeInGb": 8,  
    "isSystemDisk": false,  
    "iops": 100,  
    "path": "/dev/xvdf",  
    "state": "in-use",  
    "attachedTo": "WordPress_Multisite-1",  
    "isAttached": true,  
    "attachmentState": "attached"  
  }  
}
```

Para obtener más información, consulte el título de la guía.

- Para API obtener más información, consulte [GetDisk](#) la Referencia de AWS CLI comandos.

## get-disks

En el siguiente ejemplo de código, se muestra cómo usar `get-disks`.

### AWS CLI

Para obtener información sobre todos los discos de almacenamiento en bloque

En el siguiente `get-disks` ejemplo, se muestran detalles sobre todos los discos de la AWS región configurada.

```
aws lightsail get-disks
```

Salida:

```
{
  "disks": [
    {
      "name": "Disk-2",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:Disk/6a343ff8-6341-422d-86e2-bEXAMPLE16c2",
      "supportCode": "6EXAMPLE3362/vol-0EXAMPLE929602087",
      "createdAt": 1571090461.634,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "resourceType": "Disk",
      "tags": [],
      "sizeInGb": 8,
      "isSystemDisk": false,
      "iops": 100,
      "state": "available",
      "isAttached": false,
      "attachmentState": "detached"
    },
    {
      "name": "Disk-1",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:Disk/c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",
      "supportCode": "6EXAMPLE3362/vol-0EXAMPLEf2f88b32f",
      "createdAt": 1566585439.587,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "resourceType": "Disk",
      "tags": [],
      "sizeInGb": 8,
      "isSystemDisk": false,
      "iops": 100,
```



```

        "path": "/dev/xvdf",
        "state": "in-use",
        "attachedTo": "WordPress_Multisite-1",
        "isAttached": true,
        "attachmentState": "attached"
    }
]
}

```

- Para API obtener más información, consulte [GetDisks](#) la Referencia de AWS CLI comandos.

## get-domain

En el siguiente ejemplo de código, se muestra cómo usar `get-domain`.

### AWS CLI

Para obtener información sobre un dominio

En el siguiente `get-domain` ejemplo, se muestran detalles sobre el dominio `example.com`.

Nota: Las API operaciones de Lightsail relacionadas con el dominio solo están disponibles en la región. `us-east-1` AWS Si su CLI perfil está configurado para usar una región diferente, debe incluir el parámetro `--region us-east-1` o el comando fallará.

```

aws lightsail get-domain \
  --domain-name example.com \
  --region us-east-1

```

Salida:

```

{
  "domain": {
    "name": "example.com",
    "arn":
"arn:aws:lightsail:global:111122223333:Domain/28cda903-3f15-44b2-9baf-3EXAMPLEb304",
    "supportCode": "6EXAMPLE3362//hostedzone/ZEXAMPLEONGSC1",
    "createdAt": 1570728588.6,
    "location": {
      "availabilityZone": "all",
      "regionName": "global"
    }
  },

```

```
"resourceType": "Domain",
"tags": [],
"domainEntries": [
  {
    "id": "-1682899164",
    "name": "example.com",
    "target": "192.0.2.0",
    "isAlias": false,
    "type": "A"
  },
  {
    "id": "1703104243",
    "name": "example.com",
    "target": "ns-137.awsdns-17.com",
    "isAlias": false,
    "type": "NS"
  },
  {
    "id": "-1038331153",
    "name": "example.com",
    "target": "ns-1710.awsdns-21.co.uk",
    "isAlias": false,
    "type": "NS"
  },
  {
    "id": "-2107289565",
    "name": "example.com",
    "target": "ns-692.awsdns-22.net",
    "isAlias": false,
    "type": "NS"
  },
  {
    "id": "1582095705",
    "name": "example.com",
    "target": "ns-1436.awsdns-51.org",
    "isAlias": false,
    "type": "NS"
  },
  {
    "id": "-1769796132",
    "name": "example.com",
    "target": "ns-1710.awsdns-21.co.uk. awsdns-hostmaster.amazon.com. 1
7200 900 1209600 86400",
    "isAlias": false,
```

```

        "type": "SOA"
      }
    ]
  }
}

```

- Para API obtener más información, consulte la Referencia de comandos. [GetDomain](#) AWS CLI

## get-domains

En el siguiente ejemplo de código, se muestra cómo usar `get-domains`.

### AWS CLI

Para obtener información sobre todos los dominios

En el siguiente `get-domains` ejemplo, se muestran detalles sobre todos los dominios de la AWS región configurada.

Nota: Las API operaciones de Lightsail relacionadas con los dominios solo están disponibles en la región. `us-east-1` AWS Si su CLI perfil está configurado para usar una región diferente, debe incluir el `--region us-east-1` parámetro o el comando fallará.

```

aws lightsail get-domains \
  --region us-east-1

```

Salida:

```

{
  "domains": [
    {
      "name": "example.com",
      "arn":
"arn:aws:lightsail:global:111122223333:Domain/28cda903-3f15-44b2-9baf-3EXAMPLEb304",
      "supportCode": "6EXAMPLE3362//hostedzone/ZEXAMPLEONGSC1",
      "createdAt": 1570728588.6,
      "location": {
        "availabilityZone": "all",
        "regionName": "global"
      },
      "resourceType": "Domain",
      "tags": [],

```

```
"domainEntries": [  
  {  
    "id": "-1682899164",  
    "name": "example.com",  
    "target": "192.0.2.0",  
    "isAlias": false,  
    "type": "A"  
  },  
  {  
    "id": "1703104243",  
    "name": "example.com",  
    "target": "ns-137.awsdns-17.com",  
    "isAlias": false,  
    "type": "NS"  
  },  
  {  
    "id": "-1038331153",  
    "name": "example.com",  
    "target": "ns-4567.awsdns-21.co.uk",  
    "isAlias": false,  
    "type": "NS"  
  },  
  {  
    "id": "-2107289565",  
    "name": "example.com",  
    "target": "ns-333.awsdns-22.net",  
    "isAlias": false,  
    "type": "NS"  
  },  
  {  
    "id": "1582095705",  
    "name": "example.com",  
    "target": "ns-1111.awsdns-51.org",  
    "isAlias": false,  
    "type": "NS"  
  },  
  {  
    "id": "-1769796132",  
    "name": "example.com",  
    "target": "ns-1234.awsdns-21.co.uk. awsdns-  
hostmaster.amazon.com. 1 7200 900 1209600 86400",  
    "isAlias": false,  
    "type": "SOA"  
  },  
]
```

```
        {
          "id": "1029454894",
          "name": "_dead6a124ede046a0319eb44a4eb3cbc.example.com",
          "target": "_be133b0a0899fb7b6bf79d9741d1a383.hkvuijqjoua.acm-
validations.aws",
          "isAlias": false,
          "type": "CNAME"
        }
      ]
    },
    {
      "name": "example.net",
      "arn": "arn:aws:lightsail:global:111122223333:Domain/9c9f0d70-
c92e-4753-86c2-6EXAMPLE029d",
      "supportCode": "6EXAMPLE3362//hostedzone/ZEXAMPLE5TPKMV",
      "createdAt": 1556661071.384,
      "location": {
        "availabilityZone": "all",
        "regionName": "global"
      },
      "resourceType": "Domain",
      "tags": [],
      "domainEntries": [
        {
          "id": "-766320943",
          "name": "example.net",
          "target": "192.0.2.2",
          "isAlias": false,
          "type": "A"
        },
        {
          "id": "-453913825",
          "name": "example.net",
          "target": "ns-123.awsdns-10.net",
          "isAlias": false,
          "type": "NS"
        },
        {
          "id": "1553601564",
          "name": "example.net",
          "target": "ns-4444.awsdns-47.co.uk",
          "isAlias": false,
          "type": "NS"
        }
      ],
    }
  ],
}
```

```

        {
            "id": "1653797661",
            "name": "example.net",
            "target": "ns-7890.awsdns-61.org",
            "isAlias": false,
            "type": "NS"
        },
        {
            "id": "706414698",
            "name": "example.net",
            "target": "ns-123.awsdns-44.com",
            "isAlias": false,
            "type": "NS"
        },
        {
            "id": "337271745",
            "name": "example.net",
            "target": "ns-4444.awsdns-47.co.uk. awsdns-
hostmaster.amazon.com. 1 7200 900 1209600 86400",
            "isAlias": false,
            "type": "SOA"
        },
        {
            "id": "-1785431096",
            "name": "www.example.net",
            "target": "192.0.2.2",
            "isAlias": false,
            "type": "A"
        }
    ]
},
{
    "name": "example.org",
    "arn": "arn:aws:lightsail:global:111122223333:Domain/
f0f13ba3-3df0-4fdc-8ebb-1EXAMPLEf26e",
    "supportCode": "6EXAMPLE3362//hostedzone/ZEXAMPLEAF038",
    "createdAt": 1556661199.106,
    "location": {
        "availabilityZone": "all",
        "regionName": "global"
    },
    "resourceType": "Domain",
    "tags": [],
    "domainEntries": [

```

```
{
  "id": "2065301345",
  "name": "example.org",
  "target": "192.0.2.4",
  "isAlias": false,
  "type": "A"
},
{
  "id": "-447198516",
  "name": "example.org",
  "target": "ns-123.awsdns-45.com",
  "isAlias": false,
  "type": "NS"
},
{
  "id": "136463022",
  "name": "example.org",
  "target": "ns-9999.awsdns-15.co.uk",
  "isAlias": false,
  "type": "NS"
},
{
  "id": "1395941679",
  "name": "example.org",
  "target": "ns-555.awsdns-01.net",
  "isAlias": false,
  "type": "NS"
},
{
  "id": "872052569",
  "name": "example.org",
  "target": "ns-6543.awsdns-38.org",
  "isAlias": false,
  "type": "NS"
},
{
  "id": "1001949377",
  "name": "example.org",
  "target": "ns-1234.awsdns-15.co.uk. awsdns-
hostmaster.amazon.com. 1 7200 900 1209600 86400",
  "isAlias": false,
  "type": "SOA"
},
{
```

```

        "id": "1046191192",
        "name": "www.example.org",
        "target": "192.0.2.4",
        "isAlias": false,
        "type": "A"
      }
    ]
  }
}

```

- Para API obtener más información, consulte [GetDomains](#) la Referencia de AWS CLI comandos.

## get-export-snapshot-record

En el siguiente ejemplo de código, se muestra cómo usar `get-export-snapshot-record`.

### AWS CLI

Para obtener los registros de las instantáneas exportadas a Amazon EC2

En el siguiente `get-export-snapshot-record` ejemplo, se muestran detalles sobre las instantáneas de disco o instancias de Amazon Lightsail exportadas a Amazon. EC2

```
aws lightsail get-export-snapshot-records
```

Salida:

```

{
  "exportSnapshotRecords": [
    {
      "name": "ExportSnapshotRecord-d2da10ce-0b3c-4ae1-ab3a-2EXAMPLEa586",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:ExportSnapshotRecord/076c7060-b0cc-4162-98f0-2EXAMPLEe28e",
      "createdAt": 1543534665.678,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "ExportSnapshotRecord",
      "state": "Succeeded",
      "sourceInfo": {

```



```

        "resourceType": "InstanceSnapshot",
        "createdAt": 1540339310.706,
        "name": "WordPress-512MB-Oregon-1-1540339219",
        "arn": "arn:aws:lightsail:us-
west-2:111122223333:InstanceSnapshot/5446f534-ed60-4c17-b4a5-bEXAMPLEf8b7",
        "fromResourceName": "WordPress-512MB-Oregon-1",
        "fromResourceArn": "arn:aws:lightsail:us-
west-2:111122223333:Instance/4b8f1f24-e4d1-4cf3-88ff-cEXAMPLEa397",
        "instanceSnapshotInfo": {
            "fromBundleId": "nano_2_0",
            "fromBlueprintId": "wordpress_4_9_8",
            "fromDiskInfo": [
                {
                    "path": "/dev/sda1",
                    "sizeInGb": 20,
                    "isSystemDisk": true
                }
            ]
        }
    },
    "destinationInfo": {
        "id": "ami-0EXAMPLEc0d65058e",
        "service": "Aws::EC2::Image"
    }
},
{
    "name": "ExportSnapshotRecord-1c94e884-40ff-4fe1-9302-0EXAMPLE14c2",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:ExportSnapshotRecord/
fb392ce8-6567-4013-9bfd-3EXAMPLE5b4c",
    "createdAt": 1543432110.2,
    "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
    },
    "resourceType": "ExportSnapshotRecord",
    "state": "Succeeded",
    "sourceInfo": {
        "resourceType": "InstanceSnapshot",
        "createdAt": 1540833603.545,
        "name": "LAMP_PHP_5-512MB-Oregon-1-1540833565",
        "arn": "arn:aws:lightsail:us-
west-2:111122223333:InstanceSnapshot/82334399-b5f2-49ec-8382-0EXAMPLEe45f",
        "fromResourceName": "LAMP_PHP_5-512MB-Oregon-1",

```

```

        "fromResourceArn": "arn:aws:lightsail:us-
west-2:111122223333:Instance/863b9f35-ab1e-4418-bdd2-1EXAMPLEbab2",
        "instanceSnapshotInfo": {
            "fromBundleId": "nano_2_0",
            "fromBlueprintId": "lamp_5_6_37_2",
            "fromDiskInfo": [
                {
                    "path": "/dev/sda1",
                    "sizeInGb": 20,
                    "isSystemDisk": true
                }
            ]
        },
        "destinationInfo": {
            "id": "ami-0EXAMPLE7c5ec84e2",
            "service": "Aws::EC2::Image"
        }
    }
]
}

```

- Para API obtener más información, consulte la Referencia de [GetExportSnapshotRecord](#) comandos AWS CLI .

## get-instance-access-details

En el siguiente ejemplo de código, se muestra cómo usar `get-instance-access-details`.

### AWS CLI

Para obtener la información clave del host de una instancia

En el siguiente `get-instance-access-details` ejemplo, se muestra la información de la clave del host, por ejemplo `WordPress_Multisite-1`.

```
aws lightsail get-instance-access-details \
  --instance-name WordPress_Multisite-1
```

Salida:

```
{
```

```

"accessDetails": {
  "certKey": "ssh-rsa-cert-v01@openssh.com
  AEXAMPLEaC1yc2EtY2VydC12MDFAb3B1bnNzaC5jb20AAAAGNf076Dt3ppmPd0fPxZVMmS491aEAYYH9cHqAJ3fNML8
vEXAMPLE2eBWJyQvn7o1/
i0+s966h5sx8qUD791PB7q5UESd5VZGFtytrykfQJnjiwqe7EV5agzvjb1Lj26Fb37EKda9HVfCOu8pWbvky7Tyn9w29
+xMfQM9xVz0rXZmqx8uJidJpRgLCMTviofwQJU/
K1EXAMPLEAAAAAAAAABAAAAL500MzMzMDU4MzA4ODg1MTY2NjM40np6UW1ndHk4UE1RSG9Stit0TG5QSEE9PQAAAAAsAAA
+LiB+ozNbUA0cdNL9Y67x7qPv/R7XhTc21+2A+8+GuVpK/Kz9dqDMKNAEXAMPLE+YYN
+tiXm7Y80gziK+7iDB7xUuQ4vghmn4+qgz9mKwYgWvVe2+0XLUv7cnWPB7iU1HQg
+E3LURkV4ZFw9pj7X2dFdNkFMxwWgI1ISWKimEXAMPLEEhJrf1Rqc/
QH6TpWCvPfcx8uvwVqdwTfke/SfA5BCzbGGI1UmIUadh8nHcb5FamQ1hK7kECy47K/x9FMn/
KwmM7pCwJbSLDM07n9bnbvck6m8ZoB2N2YLMG5dW7BerEXAMPLEEobqfdtyYJHHe1EyyEJs1fWNU3D5JIGlgzcPAV
+Z1bQyUCZXf0os1Sa+HE85f0/
FRq9SVSBSHrmbeb0fr1PhgMzgSmqLeyhlbr6wwWIDbREXAMPLEJZ49H7RdQxdKyYrZPwvRgcr0qI2EL0tAajnpQQ8UZo
Aqter0xN5PhFL0J490WTacwCGRAjLhibAx7K1t/1ZXWo6c+ijq8c111327EXAMPLE/
e89GC89KcmKCxfGQniDAUgF8UqofIbq3Z0UgiAAYCVXcLI4L68NhVXyoWuQXPBRQSEXAMPLEWm74tDL9tFN3c7tSe/
Oz0cTR+4sAAAIPIAAAAB3NzaC1yc2EAAAIAQnG/
L0DqiSnLrWhEox4aHqMgd0m0oLLAYx60QH9F0TM9EXAMPLE961rzSCMon7ZgsWnNl00wZQgDG
+rtJ4N0B7H0Vwns4ynUFbzNQ3qFGGeE31KwX1L41vV1iSy7sDk8aI0LmrKJi1LE1Qc118uboR1woX0YEXAMPLEaUCeX
+10+WEXAMPLEg6Y4U4ZvE2B3xyRdpvysb5TGFNTk5qPsIacnVkoL0GsZZXMPLGJnG40BpQLLtpj9sNMxAgZPCAUjhkqk
+nx0904NUZ2pTwbVSUaV1gm6pug9xbwN01Im21t34JeLlKTqxcJ6zzS8W0c0KKpAm5c4hWkseMbyutS2jav/4hiS
+BhrYgptzfwe5qRXEXAMPLEEHZQr3YfGzYoBJ/
lLK3NHhx0ihhsfAYwMei0BFZT1F/7CT3IH4iitEkIgodI06/
Mw6UDqMPozyQCK11EA6LFhYCOZG9drWcoRa74Lm4kY9TP028Za8gDMh1WpkXLq9Gixon50HP8aM/
sEXAMPLEEr2+fnkw+1Bto05L6+vKoPlXaGqZ/fBYEXAMPLEAMQHjnLM1JYNvtEEPhp+TNzXHzuixWf/
Ht04m0AVpXrzIDXaS102tXY=",
  "ipAddress": "192.0.2.0",
  "privateKey": "-----BEGIN RSA PRIVATE KEY-----
\nEXAMPLEEBAAKCAQEA+AD3qeU2toBy505v7wnRLVo/tngVickL5+6Jf4tPrPeuoebM
\nfKlA+/ZTwe6uVBENEVRhbcra8pH0CZ44sKnuxFeWoM7425S49uhW9+xCnWvR1Xw
\njrvKvm75Mu08p/cNvfWugrBuaPB65DspgxNn0fZWMVxpIpSq0SPWmSwQHV597d6C
\nrEXAMPLEo8hJmqz2KFQ09X7fB21BruGgr9aXiNPmWmovYKqWfmrFvR7odFmDecq
\n5EXAMPLE9dyU1ZsrWhGby77eYrVaF10GNGQ8qy1HGUIScquZ9NDIL49n4mXbfsTH
\n0EXAMPLE12ZqsfLiYnSaUYCwjE74qH8ECVPytQIDAQABaoIBAHeZV9Z58JHAjifz
\nCEXAMPLEEqC3do0VDgXS1kKI92qNo4z2VcUEho878paCuVVXVHcCGgSnGeyIh2tN
\nMEXAMPLESohR427BhH3YLA+3Z5SIVnejbTgYPfLC37B8khTaYqkqMvdZiFVZK5qn
\nIEXAMPLEM93oF9eSZCjcLKB/jGHsfb0eCDMP8BshHE2beuqzVMoK1Dx0nvoP3+Fp
\nAEXAMPLESq6pDpCo9YVUX8g1u3Ro9cPl2LXHDy+oVEY5KhbZQJ7VU1I72W0vppWW
\n0EXAMPLEkgY1q7p6qYtYcSgTEjz14gDiMfQ7SyHB3alkIoNONQ9ZPaWHyJvymeud
\noQTNuz0CgYEA/LFWNTEZrzdZdR1kJmyNRmAermU0B6utyNENChAlHGSHkB+11VSh
\nbEXAMPLEQo9ooUeW5Ux03YwacZLoDT1mwxw1PtC1+PNycZoLe1fE9UdARrdmGTob
\n8L7CPLSXp3xuR8VqSp2fnIc7hfiQs/NrPX9gm/E0rB0we0RKyDSzWScCgYEA+z/r
\nniob+nJZq0Ybn0SuP6oMULP4vnWniWj8MIhUJU53LwSAM8DeJd0NKDdkui0d52aAL
\nVgn7nLo88rVWKhJwVc4tu/rNgZLcR3bP4+kL6zand0KQnMLy0zNA2Ys26aa5udH1\nnqWl0WTt9WEm/

```

```

h10ndC1kn0MectrvsG17b38y5sMCgYEA54NiRGGz8oCPW6GN/FZA
\nKEXAMPLE5tw34GEH3Uxlcn3CejDaQmcz0ATwX4nIwRZDEqWyYZcS0btg1jhGiBD\nYEXAMPLEkc8Z71L/
agZEAaVCEog9FqfSqwB
+XTfoKh8qur74X1yCu9p6gof1q6k9\nEXAMPLEechJcNN0g4ETIfMkCgYBdV0RRhE4mqvWp0dzA7v66FdEz2YSkjAXKk
\naEXAMPLE8Z/8yBSmuBv1Qv03XA12my462uB92uzzGAuW
+1yBc2Kn1sXqYTy0y1z0\nngEXAMPLEBogjw4MqHKL1bPKMHyQU8/
q24PaYgzHPzy13w1H6pTYf1Xq1HdE2D6Vv\nYEXAMPLEegQC3i/
kVvhky/2XRwRVLC7J02Bg3QGTx38hpmDa5IuofKANjA+Wa3/zy\nbEXAMPLE6ytQgD9GN/YtBq+uh0
+2ZkvXPL+CWRi0ZRxpPwYDBBFU9Cw0AuWWG1L8\nwEXAMPLExM1cysRgcWB9RNgf3AuOpFd2i6XT/
riNsvvkpmJ+VooU8g==\n-----END RSA PRIVATE KEY-----\n",
    "protocol": "ssh",
    "instanceName": "WordPress_Multisite-1",
    "username": "bitnami",
    "hostKeys": [
        {
            "algorithm": "ssh-rsa",
            "publicKey":
                "AEXAMPLEaC1yc2EAAAADAQABAAQCoer9ieZTjQ3pXCHczuAYZFj1F7t
+uBkXuqeGMRex78pCvmS+DiEXAMPLEEuJ1Q8dcKhrQL4HpXbd9dosVCTaJnJwb4MQqsuSVFdHFzy3guP
+BKclWqtXJEXAMPLEsBGqZzlrIv6a9bTA0TCplZ8AD+hSRTaSXXqg6FT
+Qf16IktH0X1Ms7xIEXAMPLEmNtjCpzZiGXDHzytoMvUgwa8uHPp440g36EUu4VqQxoUHPJKoXvcQizyk3K8ym0hP0Tp
0t6y9HwvykEXAMPLEAfbKjbr42+u6+0Slkr4d339q2U1sTDytJhhs8HUel1wTfGRfp",
            "witnessedAt": 1570744377.699,
            "fingerprintSHA1": "SHA1:GEXAMPLEMoYgUg0ucadqU9Bt3Lk",
            "fingerprintSHA256": "SHA256:IEEXAMPLEcB5vgxnAUoJawbdZ
+MwELhIp6FUxuwq/LIU"
        },
        {
            "algorithm": "ssh-ed25519",
            "publicKey":
                "AEXAMPLEaC1lZDI1NTE5AAAAIC1gwGPDfGa0NxEXAMPLEJX3UNap781QxHQmn8nzlrUv",
            "witnessedAt": 1570744377.697,
            "fingerprintSHA1": "SHA1:VEXAMPLE5ReqSmTgv03sSUw9toU",
            "fingerprintSHA256": "SHA256:0EXAMPLEdE6tI95k3TJpG
+qhJbAoknB0yz9nAEaDt3A"
        },
        {
            "algorithm": "ecdsa-sha2-nistp256",
            "publicKey":
                "AEXAMPLEZHNhLXNoYTIItbmLzdHAYNTYAAAAIbmLzdHAYNTYAAABEXAMPLE9B4mZy8YSsZW7cixCDq5yHSAAxjJkDo5
+EnK1DCsYtUkxxEXAMPLE6V0WL2z63RTKa2AUPgd8irjxWI=",
            "witnessedAt": 1570744377.707,
            "fingerprintSHA1": "SHA1:UEEXAMPLE0YCFXsCf2G6tDg+7YG0",

```

```

        "fingerprintSHA256": "SHA256:wEXAMPLEQ9a/
iEXAMPLEhRufm6U9vFU4cpkMPHnBsNA"
    }
]
}
}

```

- Para API obtener más información, consulte [GetInstanceAccessDetails](#) la Referencia de AWS CLI comandos.

## get-instance-metric-data

En el siguiente ejemplo de código, se muestra cómo usar `get-instance-metric-data`.

### AWS CLI

Para obtener datos métricos de una instancia

El siguiente `get-instance-metric-data` ejemplo devuelve el porcentaje medio de `CPUUtilization` cada 7200 segundo (2 horas) entre 1571342400 y, 1571428800 por ejemplo `MEAN-1`.

Se recomienda utilizar un conversor de tiempo unix para identificar las horas de inicio y finalización.

```

aws lightsail get-instance-metric-data \
  --instance-name MEAN-1 \
  --metric-name CPUUtilization \
  --period 7200 \
  --start-time 1571342400 \
  --end-time 1571428800 \
  --unit Percent \
  --statistics Average

```

Salida:

```

{
  "metricName": "CPUUtilization",
  "metricData": [
    {
      "average": 0.26113718770120725,
      "timestamp": 1571342400.0,

```

```
    "unit": "Percent"
  },
  {
    "average": 0.26861268928111953,
    "timestamp": 1571392800.0,
    "unit": "Percent"
  },
  {
    "average": 0.28187475104748777,
    "timestamp": 1571378400.0,
    "unit": "Percent"
  },
  {
    "average": 0.2651936960458352,
    "timestamp": 1571421600.0,
    "unit": "Percent"
  },
  {
    "average": 0.2561856213712188,
    "timestamp": 1571371200.0,
    "unit": "Percent"
  },
  {
    "average": 0.3021383254607764,
    "timestamp": 1571356800.0,
    "unit": "Percent"
  },
  {
    "average": 0.2618381649223539,
    "timestamp": 1571407200.0,
    "unit": "Percent"
  },
  {
    "average": 0.26331929394825787,
    "timestamp": 1571400000.0,
    "unit": "Percent"
  },
  {
    "average": 0.2576348407007818,
    "timestamp": 1571385600.0,
    "unit": "Percent"
  },
  {
    "average": 0.2513008454658378,
```

```

        "timestamp": 1571364000.0,
        "unit": "Percent"
    },
    {
        "average": 0.26329974562758346,
        "timestamp": 1571414400.0,
        "unit": "Percent"
    },
    {
        "average": 0.2667092536656445,
        "timestamp": 1571349600.0,
        "unit": "Percent"
    }
]
}

```

- Para API obtener más información, consulte [GetInstanceMetricData](#) la Referencia de AWS CLI comandos.

## get-instance-port-states

En el siguiente ejemplo de código, se muestra cómo usar `get-instance-port-states`.

### AWS CLI

Para obtener información sobre el firewall de una instancia

El siguiente `get-instance-port-states` ejemplo devuelve los puertos del firewall configurados, por ejemplo `MEAN-1`.

```
aws lightsail get-instance-port-states \
  --instance-name MEAN-1
```

Salida:

```

{
  "portStates": [
    {
      "fromPort": 80,
      "toPort": 80,
      "protocol": "tcp",
      "state": "open"
    }
  ]
}

```

```

    },
    {
      "fromPort": 22,
      "toPort": 22,
      "protocol": "tcp",
      "state": "open"
    },
    {
      "fromPort": 443,
      "toPort": 443,
      "protocol": "tcp",
      "state": "open"
    }
  ]
}

```

- Para API obtener más información, consulte [GetInstancePortStates](#) la Referencia de AWS CLI comandos.

## get-instance-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `get-instance-snapshot`.

### AWS CLI

Para obtener información sobre una instantánea de instancia específica

En el siguiente `get-instance-snapshot` ejemplo, se muestran detalles sobre la instantánea de la instancia especificada.

```

aws lightsail get-instance-snapshot \
  --instance-snapshot-name MEAN-1-1571419854

```

Salida:

```

{
  "instanceSnapshot": {
    "name": "MEAN-1-1571419854",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:InstanceSnapshot/
ac54700c-48a8-40fd-b065-2EXAMPLEac8f",
    "supportCode": "6EXAMPLE3362/ami-0EXAMPLE67a73020d",
    "createdAt": 1571419891.927,

```



```
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "InstanceSnapshot",
    "tags": [],
    "state": "available",
    "fromAttachedDisks": [],
    "fromInstanceName": "MEAN-1",
    "fromInstanceArn": "arn:aws:lightsail:us-west-2:111122223333:Instance/
bd470fc5-a68b-44c5-8dbc-8EXAMPLEbada",
    "fromBlueprintId": "mean_4_0_9",
    "fromBundleId": "medium_2_0",
    "isFromAutoSnapshot": false,
    "sizeInGb": 80
  }
}
```

- Para API obtener más información, consulte [GetInstanceSnapshot](#) la Referencia de AWS CLI comandos.

## get-instance-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `get-instance-snapshots`.

### AWS CLI

Para obtener información sobre todas las instantáneas de su instancia

En el siguiente `get-instance-snapshots` ejemplo, se muestran detalles sobre todas las instantáneas de instancia de la región configurada AWS .

```
aws lightsail get-instance-snapshots
```

Salida:

```
{
  "instanceSnapshots": [
    {
      "name": "MEAN-1-1571421498",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:InstanceSnapshot/
a20e6ebe-b0ee-4ae4-a750-3EXAMPLEcb0c",
```

```
    "supportCode": "6EXAMPLE3362/ami-0EXAMPLEe33cabfa1",
    "createdAt": 1571421527.755,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "InstanceSnapshot",
    "tags": [
      {
        "key": "no_delete"
      }
    ],
    "state": "available",
    "fromAttachedDisks": [],
    "fromInstanceName": "MEAN-1",
    "fromInstanceArn": "arn:aws:lightsail:us-
west-2:111122223333:Instance/1761aa0a-6038-4f25-8b94-2EXAMPLE19fd",
    "fromBlueprintId": "wordpress_5_1_1_2",
    "fromBundleId": "micro_2_0",
    "isFromAutoSnapshot": false,
    "sizeInGb": 40
  },
  {
    "name": "MEAN-1-1571419854",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:InstanceSnapshot/
ac54700c-48a8-40fd-b065-2EXAMPLEac8f",
    "supportCode": "6EXAMPLE3362/ami-0EXAMPLE67a73020d",
    "createdAt": 1571419891.927,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "InstanceSnapshot",
    "tags": [],
    "state": "available",
    "fromAttachedDisks": [],
    "fromInstanceName": "MEAN-1",
    "fromInstanceArn": "arn:aws:lightsail:us-west-2:111122223333:Instance/
bd470fc5-a68b-44c5-8dbc-8EXAMPLEebada",
    "fromBlueprintId": "mean_4_0_9",
    "fromBundleId": "medium_2_0",
    "isFromAutoSnapshot": false,
    "sizeInGb": 80
  }
}
```

```
]
}
```

- Para API obtener más información, consulte [GetInstanceSnapshots](#) la Referencia de AWS CLI comandos.

## get-instance-state

En el siguiente ejemplo de código, se muestra cómo usar `get-instance-state`.

### AWS CLI

Para obtener información sobre el estado de una instancia

En el siguiente `get-instance-state` ejemplo, se devuelve el estado de la instancia especificada.

```
aws lightsail get-instance-state \
  --instance-name MEAN-1
```

Salida:

```
{
  "state": {
    "code": 16,
    "name": "running"
  }
}
```

- Para API obtener más información, consulte [GetInstanceState](#) la Referencia de AWS CLI comandos.

## get-instance

En el siguiente ejemplo de código, se muestra cómo usar `get-instance`.

### AWS CLI

Para obtener información sobre una instancia

En el siguiente `get-instance` ejemplo, se muestran detalles sobre la instancia `MEAN-1`.

```
aws lightsail get-instance \  
--instance-name MEAN-1
```

Salida:

```
{  
  "instance": {  
    "name": "MEAN-1",  
    "arn": "arn:aws:lightsail:us-west-2:111122223333:Instance/bd470fc5-  
a68b-44c5-8dbc-EXAMPLE4bada",  
    "supportCode": "6EXAMPLE3362/i-05EXAMPLE407c97d3",  
    "createdAt": 1570635023.124,  
    "location": {  
      "availabilityZone": "us-west-2a",  
      "regionName": "us-west-2"  
    },  
    "resourceType": "Instance",  
    "tags": [],  
    "blueprintId": "mean_4_0_9",  
    "blueprintName": "MEAN",  
    "bundleId": "medium_2_0",  
    "isStaticIp": false,  
    "privateIpAddress": "192.0.2.0",  
    "publicIpAddress": "192.0.2.0",  
    "hardware": {  
      "cpuCount": 2,  
      "disks": [  
        {  
          "createdAt": 1570635023.124,  
          "sizeInGb": 80,  
          "isSystemDisk": true,  
          "iops": 240,  
          "path": "/dev/sda1",  
          "attachedTo": "MEAN-1",  
          "attachmentState": "attached"  
        }  
      ],  
      "ramSizeInGb": 4.0  
    },  
    "networking": {  
      "monthlyTransfer": {  
        "gbPerMonthAllocated": 4096  
      }  
    }  
  }  
}
```

```
    "ports": [
      {
        "fromPort": 80,
        "toPort": 80,
        "protocol": "tcp",
        "accessFrom": "Anywhere (0.0.0.0/0)",
        "accessType": "public",
        "commonName": "",
        "accessDirection": "inbound"
      },
      {
        "fromPort": 22,
        "toPort": 22,
        "protocol": "tcp",
        "accessFrom": "Anywhere (0.0.0.0/0)",
        "accessType": "public",
        "commonName": "",
        "accessDirection": "inbound"
      },
      {
        "fromPort": 443,
        "toPort": 443,
        "protocol": "tcp",
        "accessFrom": "Anywhere (0.0.0.0/0)",
        "accessType": "public",
        "commonName": "",
        "accessDirection": "inbound"
      }
    ],
    "state": {
      "code": 16,
      "name": "running"
    },
    "username": "bitnami",
    "sshKeyName": "MyKey"
  }
}
```

- Para API obtener más información, consulte [GetInstance](#) la Referencia de AWS CLI comandos.

## get-instances

En el siguiente ejemplo de código, se muestra cómo usar `get-instances`.

### AWS CLI

Para obtener información sobre todas las instancias

En el siguiente `get-instances` ejemplo, se muestran detalles sobre todas las instancias de la AWS región configurada.

```
aws lightsail get-instances
```

Salida:

```
{
  "instances": [
    {
      "name": "Windows_Server_2016-1",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:Instance/0f44fbb9-8f55-4e47-a25e-EXAMPLE04763",
      "supportCode": "62EXAMPLE362/i-0bEXAMPLE71a686b9",
      "createdAt": 1571332358.665,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "resourceType": "Instance",
      "tags": [],
      "blueprintId": "windows_server_2016",
      "blueprintName": "Windows Server 2016",
      "bundleId": "small_win_2_0",
      "isStaticIp": false,
      "privateIpAddress": "192.0.2.0",
      "publicIpAddress": "192.0.2.0",
      "hardware": {
        "cpuCount": 1,
        "disks": [
          {
            "createdAt": 1571332358.665,
            "sizeInGb": 60,
            "isSystemDisk": true,
            "iops": 180,

```

```
        "path": "/dev/sda1",
        "attachedTo": "Windows_Server_2016-1",
        "attachmentState": "attached"
    },
    {
        "name": "my-disk-for-windows-server",
        "arn": "arn:aws:lightsail:us-
west-2:111122223333:Disk/4123a81c-484c-49ea-afea-5EXAMPLEda87",
        "supportCode": "6EXAMPLE3362/vol-0EXAMPLEb2b99ca3d",
        "createdAt": 1571355063.494,
        "location": {
            "availabilityZone": "us-west-2a",
            "regionName": "us-west-2"
        },
        "resourceType": "Disk",
        "tags": [],
        "sizeInGb": 128,
        "isSystemDisk": false,
        "iops": 384,
        "path": "/dev/xvdf",
        "state": "in-use",
        "attachedTo": "Windows_Server_2016-1",
        "isAttached": true,
        "attachmentState": "attached"
    }
],
"ramSizeInGb": 2.0
},
"networking": {
    "monthlyTransfer": {
        "gbPerMonthAllocated": 3072
    },
    "ports": [
        {
            "fromPort": 80,
            "toPort": 80,
            "protocol": "tcp",
            "accessFrom": "Anywhere (0.0.0.0/0)",
            "accessType": "public",
            "commonName": "",
            "accessDirection": "inbound"
        },
        {
            "fromPort": 22,
```

```

        "toPort": 22,
        "protocol": "tcp",
        "accessFrom": "Anywhere (0.0.0.0/0)",
        "accessType": "public",
        "commonName": "",
        "accessDirection": "inbound"
    },
    {
        "fromPort": 3389,
        "toPort": 3389,
        "protocol": "tcp",
        "accessFrom": "Anywhere (0.0.0.0/0)",
        "accessType": "public",
        "commonName": "",
        "accessDirection": "inbound"
    }
]
},
"state": {
    "code": 16,
    "name": "running"
},
"username": "Administrator",
"sshKeyName": "LightsailDefaultKeyPair"
},
{
    "name": "MEAN-1",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:Instance/bd470fc5-
a68b-44c5-8dbc-8EXAMPLEbada",
    "supportCode": "6EXAMPLE3362/i-0EXAMPLEa407c97d3",
    "createdAt": 1570635023.124,
    "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
    },
    "resourceType": "Instance",
    "tags": [],
    "blueprintId": "mean_4_0_9",
    "blueprintName": "MEAN",
    "bundleId": "medium_2_0",
    "isStaticIp": false,
    "privateIpAddress": "192.0.2.0",
    "publicIpAddress": "192.0.2.0",
    "hardware": {

```



```
    "cpuCount": 2,
    "disks": [
      {
        "name": "Disk-1",
        "arn": "arn:aws:lightsail:us-west-2:111122223333:Disk/
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",
        "supportCode": "6EXAMPLE3362/vol-0EXAMPLEf2f88b32f",
        "createdAt": 1566585439.587,
        "location": {
          "availabilityZone": "us-west-2a",
          "regionName": "us-west-2"
        },
        "resourceType": "Disk",
        "tags": [
          {
            "key": "test"
          }
        ],
        "sizeInGb": 8,
        "isSystemDisk": false,
        "iops": 100,
        "path": "/dev/xvdf",
        "state": "in-use",
        "attachedTo": "MEAN-1",
        "isAttached": true,
        "attachmentState": "attached"
      },
      {
        "createdAt": 1570635023.124,
        "sizeInGb": 80,
        "isSystemDisk": true,
        "iops": 240,
        "path": "/dev/sda1",
        "attachedTo": "MEAN-1",
        "attachmentState": "attached"
      }
    ],
    "ramSizeInGb": 4.0
  },
  "networking": {
    "monthlyTransfer": {
      "gbPerMonthAllocated": 4096
    },
    "ports": [
```

```
    {
      "fromPort": 80,
      "toPort": 80,
      "protocol": "tcp",
      "accessFrom": "Anywhere (0.0.0.0/0)",
      "accessType": "public",
      "commonName": "",
      "accessDirection": "inbound"
    },
    {
      "fromPort": 22,
      "toPort": 22,
      "protocol": "tcp",
      "accessFrom": "Anywhere (0.0.0.0/0)",
      "accessType": "public",
      "commonName": "",
      "accessDirection": "inbound"
    },
    {
      "fromPort": 443,
      "toPort": 443,
      "protocol": "tcp",
      "accessFrom": "Anywhere (0.0.0.0/0)",
      "accessType": "public",
      "commonName": "",
      "accessDirection": "inbound"
    }
  ]
},
"state": {
  "code": 16,
  "name": "running"
},
"username": "bitnami",
"sshKeyName": "MyTestKey"
}
]
```

- Para API obtener más información, consulte [GetInstances](#) la Referencia de AWS CLI comandos.

## get-key-pair

En el siguiente ejemplo de código, se muestra cómo usar `get-key-pair`.

### AWS CLI

Para obtener información sobre un key pair

En el siguiente `get-key-pair` ejemplo, se muestran detalles sobre el key pair especificado.

```
aws lightsail get-key-pair \  
  --key-pair-name MyKey1
```

Salida:

```
{  
  "keyPair": {  
    "name": "MyKey1",  
    "arn": "arn:aws:lightsail:us-west-2:111122223333:KeyPair/19a4efdf-3054-43d6-91fd-eEXAMPLE21bf",  
    "supportCode": "6EXAMPLE3362/MyKey1",  
    "createdAt": 1571255026.975,  
    "location": {  
      "availabilityZone": "all",  
      "regionName": "us-west-2"  
    },  
    "resourceType": "KeyPair",  
    "tags": [],  
    "fingerprint": "00:11:22:33:44:55:66:77:88:99:aa:bb:cc:dd:ee:ff:gg:hh:ii:jj"  
  }  
}
```

- Para API obtener más información, consulte [GetKeyPair](#) la Referencia de AWS CLI comandos.

## get-key-pairs

En el siguiente ejemplo de código, se muestra cómo usar `get-key-pairs`.

### AWS CLI

Para obtener información sobre todos los pares de claves

En el siguiente `get-key-pairs` ejemplo, se muestran detalles sobre todos los pares de claves de la AWS región configurada.

```
aws lightsail get-key-pairs
```

Salida:

```
{
  "keyPairs": [
    {
      "name": "MyKey1",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:KeyPair/19a4efdf-3054-43d6-91fd-eEXAMPLE21bf",
      "supportCode": "6EXAMPLE3362/MyKey1",
      "createdAt": 1571255026.975,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "KeyPair",
      "tags": [],
      "fingerprint":
      "00:11:22:33:44:55:66:77:88:99:aa:bb:cc:dd:ee:ff:gg:hh:ii:jj"
    }
  ]
}
```

- Para API obtener más información, consulte [GetKeyPairs](#) la Referencia de AWS CLI comandos.

## get-load-balancer-tls-certificates

En el siguiente ejemplo de código, se muestra cómo usar `get-load-balancer-tls-certificates`.

AWS CLI

Para obtener información sobre TLS los certificados de un balanceador de cargas

En el siguiente `get-load-balancer-tls-certificates` ejemplo, se muestran detalles sobre los TLS certificados del equilibrador de cargas especificado.

```
aws lightsail get-load-balancer-tls-certificates \
```

```
--load-balancer-name LoadBalancer-1
```

Salida:

```
{
  "tlsCertificates": [
    {
      "name": "example-com",
      "arn": "arn:aws:lightsail:us-
west-2:111122223333:LoadBalancerTlsCertificate/d7bf4643-6a02-4cd4-b3c4-
fEXAMPLE9b4d",
      "supportCode": "6EXAMPLE3362/arn:aws:acm:us-
west-2:333322221111:certificate/9af8e32c-a54e-4a67-8c63-cEXAMPLEb314",
      "createdAt": 1571678025.3,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "LoadBalancerTlsCertificate",
      "loadBalancerName": "LoadBalancer-1",
      "isAttached": false,
      "status": "ISSUED",
      "domainName": "example.com",
      "domainValidationRecords": [
        {
          "name": "_dEXAMPLE4ede046a0319eb44a4eb3cbc.example.com.",
          "type": "CNAME",
          "value": "_bEXAMPLE0899fb7b6bf79d9741d1a383.hkvuiqjoua.acm-
validations.aws.",
          "validationStatus": "SUCCESS",
          "domainName": "example.com"
        }
      ],
      "issuedAt": 1571678070.0,
      "issuer": "Amazon",
      "keyAlgorithm": "RSA-2048",
      "notAfter": 1605960000.0,
      "notBefore": 1571616000.0,
      "serial": "00:11:22:33:44:55:66:77:88:99:aa:bb:cc:dd:ee:ff",
      "signatureAlgorithm": "SHA256WITHRSA",
      "subject": "CN=example.com",
      "subjectAlternativeNames": [
        "example.com"
      ]
    }
  ]
}
```

```

    ]
  }
]
}

```

- Para API obtener más información, consulte [GetLoadBalancerTlsCertificates](#) la Referencia de AWS CLI comandos.

## get-load-balancer

En el siguiente ejemplo de código, se muestra cómo usar `get-load-balancer`.

### AWS CLI

Para obtener información sobre un balanceador de cargas

En el siguiente `get-load-balancer` ejemplo, se muestran detalles sobre el equilibrador de cargas especificado.

```

aws lightsail get-load-balancer \
  --load-balancer-name LoadBalancer-1

```

Salida:

```

{
  "loadBalancer": {
    "name": "LoadBalancer-1",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:LoadBalancer/40486b2b-1ad0-4152-83e4-cEXAMPLE6f4b",
    "supportCode": "6EXAMPLE3362/arn:aws:elasticloadbalancing:us-west-2:333322221111:loadbalancer/app/bEXAMPLE128cb59d86f946a9395dd304/1EXAMPLE8dd9d77e",
    "createdAt": 1571677906.723,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "LoadBalancer",
    "tags": [],
    "dnsName": "bEXAMPLE128cb59d86f946a9395dd304-1486911371.us-west-2.elb.amazonaws.com",
    "state": "active",
  }
}

```

```
    "protocol": "HTTP",
    "publicPorts": [
      80
    ],
    "healthCheckPath": "/",
    "instancePort": 80,
    "instanceHealthSummary": [
      {
        "instanceName": "MEAN-3",
        "instanceHealth": "healthy"
      },
      {
        "instanceName": "MEAN-1",
        "instanceHealth": "healthy"
      },
      {
        "instanceName": "MEAN-2",
        "instanceHealth": "healthy"
      }
    ],
    "tlsCertificateSummaries": [
      {
        "name": "example-com",
        "isAttached": false
      }
    ],
    "configurationOptions": {
      "SessionStickinessEnabled": "false",
      "SessionStickiness_LB_CookieDurationSeconds": "86400"
    }
  }
}
```

- Para API obtener más información, consulte [GetLoadBalancer](#) la Referencia de AWS CLI comandos.

## get-load-balancers

En el siguiente ejemplo de código, se muestra cómo usar `get-load-balancers`.

### AWS CLI

Para obtener información sobre todos los balanceadores de carga

En el siguiente `get-load-balancers` ejemplo, se muestran detalles sobre todos los equilibradores de carga de la región configurada AWS .

```
aws lightsail get-load-balancers
```

Salida:

```
{
  "loadBalancers": [
    {
      "name": "LoadBalancer-1",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:LoadBalancer/40486b2b-1ad0-4152-83e4-cEXAMPLE6f4b",
      "supportCode": "6EXAMPLE3362/arn:aws:elasticloadbalancing:us-west-2:333322221111:loadbalancer/app/bEXAMPLE128cb59d86f946a9395dd304/1EXAMPLE8dd9d77e",
      "createdAt": 1571677906.723,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "LoadBalancer",
      "tags": [],
      "dnsName": "bEXAMPLE128cb59d86f946a9395dd304-1486911371.us-west-2.elb.amazonaws.com",
      "state": "active",
      "protocol": "HTTP",
      "publicPorts": [
        80
      ],
      "healthCheckPath": "/",
      "instancePort": 80,
      "instanceHealthSummary": [
        {
          "instanceName": "MEAN-3",
          "instanceHealth": "healthy"
        },
        {
          "instanceName": "MEAN-1",
          "instanceHealth": "healthy"
        },
        {
          "instanceName": "MEAN-2",
```



```

        "instanceHealth": "healthy"
      }
    ],
    "tlsCertificateSummaries": [
      {
        "name": "example-com",
        "isAttached": false
      }
    ],
    "configurationOptions": {
      "SessionStickinessEnabled": "false",
      "SessionStickiness_LB_CookieDurationSeconds": "86400"
    }
  }
]
}

```

- Para API obtener más información, consulte [GetLoadBalancers](#) la Referencia de AWS CLI comandos.

## get-operation

En el siguiente ejemplo de código, se muestra cómo usar `get-operation`.

### AWS CLI

Para obtener información sobre una sola operación

En el siguiente `get-operation` ejemplo, se muestran detalles sobre la operación especificada.

```

aws lightsail get-operation \
  --operation-id e5700e8a-daf2-4b49-bc01-3EXAMPLE910a

```

Salida:

```

{
  "operation": {
    "id": "e5700e8a-daf2-4b49-bc01-3EXAMPLE910a",
    "resourceName": "Instance-1",
    "resourceType": "Instance",
    "createdAt": 1571679872.404,

```

```
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationType": "CreateInstance",
    "status": "Succeeded",
    "statusChangedAt": 1571679890.304
  }
}
```

- Para API obtener más información, consulte [GetOperation](#) la Referencia de AWS CLI comandos.

## get-operations-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `get-operations-for-resource`.

### AWS CLI

Para obtener todas las operaciones de un recurso

En el siguiente `get-operations-for-resource` ejemplo, se muestran detalles sobre todas las operaciones del recurso especificado.

```
aws lightsail get-operations-for-resource \
  --resource-name LoadBalancer-1
```

Salida:

```
{
  "operations": [
    {
      "id": "e2973046-43f8-4252-a4b4-9EXAMPLE69ce",
      "resourceName": "LoadBalancer-1",
      "resourceType": "LoadBalancer",
      "createdAt": 1571678786.071,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      }
    },
  ],
}
```

```

    "isTerminal": true,
    "operationDetails": "MEAN-1",
    "operationType": "DetachInstancesFromLoadBalancer",
    "status": "Succeeded",
    "statusChangedAt": 1571679087.57
  },
  {
    "id": "2d742a18-0e7f-48c8-9705-3EXAMPLEf98a",
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancer",
    "createdAt": 1571678782.784,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "MEAN-1",
    "operationType": "AttachInstancesToLoadBalancer",
    "status": "Succeeded",
    "statusChangedAt": 1571678798.465
  },
  {
    "id": "6c700fcc-4246-40ab-952b-1EXAMPLEdac2",
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancer",
    "createdAt": 1571678775.297,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "MEAN-3",
    "operationType": "AttachInstancesToLoadBalancer",
    "status": "Succeeded",
    "statusChangedAt": 1571678842.806
  },
  ...
}
]
}

```

- Para API obtener más información, consulte [GetOperationsForResource](#) la Referencia de AWS CLI comandos.

## get-operations

En el siguiente ejemplo de código, se muestra cómo usar `get-operations`.

### AWS CLI

Para obtener información sobre todas las operaciones

En el siguiente `get-operations` ejemplo, se muestran detalles sobre todas las operaciones de la AWS región configurada.

```
aws lightsail get-operations
```

Salida:

```
{
  "operations": [
    {
      "id": "e5700e8a-daf2-4b49-bc01-3EXAMPLE910a",
      "resourceName": "Instance-1",
      "resourceType": "Instance",
      "createdAt": 1571679872.404,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationType": "CreateInstance",
      "status": "Succeeded",
      "statusChangedAt": 1571679890.304
    },
    {
      "id": "701a3339-930e-4914-a9f9-7EXAMPLE68d7",
      "resourceName": "WordPress-1",
      "resourceType": "Instance",
      "createdAt": 1571678786.072,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "isTerminal": true,
      "operationDetails": "LoadBalancer-1",
      "operationType": "DetachInstancesFromLoadBalancer",

```

```

    "status": "Succeeded",
    "statusChangedAt": 1571679086.399
  },
  {
    "id": "e2973046-43f8-4252-a4b4-9EXAMPLE69ce",
    "resourceName": "LoadBalancer-1",
    "resourceType": "LoadBalancer",
    "createdAt": 1571678786.071,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "WordPress-1",
    "operationType": "DetachInstancesFromLoadBalancer",
    "status": "Succeeded",
    "statusChangedAt": 1571679087.57
  },
  ...
}
]
}

```

- Para API obtener más información, consulte [GetOperations](#) la Referencia de AWS CLI comandos.

## get-regions

En el siguiente ejemplo de código, se muestra cómo usar `get-regions`.

### AWS CLI

Para obtener todas las AWS regiones de Amazon Lightsail

En el siguiente `get-regions` ejemplo, se muestran detalles sobre todas las AWS regiones de Amazon Lightsail.

```
aws lightsail get-regions
```

Salida:

```
{
```

```
"regions": [
  {
    "continentCode": "NA",
    "description": "This region is recommended to serve users in the eastern
United States",
    "displayName": "Virginia",
    "name": "us-east-1",
    "availabilityZones": [],
    "relationalDatabaseAvailabilityZones": []
  },
  {
    "continentCode": "NA",
    "description": "This region is recommended to serve users in the eastern
United States",
    "displayName": "Ohio",
    "name": "us-east-2",
    "availabilityZones": [],
    "relationalDatabaseAvailabilityZones": []
  },
  {
    "continentCode": "NA",
    "description": "This region is recommended to serve users in the
northwestern United States, Alaska, and western Canada",
    "displayName": "Oregon",
    "name": "us-west-2",
    "availabilityZones": [],
    "relationalDatabaseAvailabilityZones": []
  },
  ...
]
}
```

- Para API obtener más información, consulte la Referencia [GetRegions](#) de AWS CLI comandos.

## get-relational-database-blueprints

En el siguiente ejemplo de código, se muestra cómo usar `get-relational-database-blueprints`.

### AWS CLI

Para obtener los planos de las nuevas bases de datos relacionales

El siguiente `get-relational-database-blueprints` ejemplo muestra detalles sobre todos los esquemas de bases de datos relacionales disponibles que se pueden usar para crear nuevas bases de datos relacionales en Amazon Lightsail.

```
aws lightsail get-relational-database-blueprints
```

Salida:

```
{
  "blueprints": [
    {
      "blueprintId": "mysql_5_6",
      "engine": "mysql",
      "engineVersion": "5.6.44",
      "engineDescription": "MySQL Community Edition",
      "engineVersionDescription": "MySQL 5.6.44",
      "isEngineDefault": false
    },
    {
      "blueprintId": "mysql_5_7",
      "engine": "mysql",
      "engineVersion": "5.7.26",
      "engineDescription": "MySQL Community Edition",
      "engineVersionDescription": "MySQL 5.7.26",
      "isEngineDefault": true
    },
    {
      "blueprintId": "mysql_8_0",
      "engine": "mysql",
      "engineVersion": "8.0.16",
      "engineDescription": "MySQL Community Edition",
      "engineVersionDescription": "MySQL 8.0.16",
      "isEngineDefault": false
    },
    {
      "blueprintId": "postgres_9_6",
      "engine": "postgres",
      "engineVersion": "9.6.15",
      "engineDescription": "PostgreSQL",
      "engineVersionDescription": "PostgreSQL 9.6.15-R1",
      "isEngineDefault": false
    },
    {
```

```

        "blueprintId": "postgres_10",
        "engine": "postgres",
        "engineVersion": "10.10",
        "engineDescription": "PostgreSQL",
        "engineVersionDescription": "PostgreSQL 10.10-R1",
        "isEngineDefault": false
    },
    {
        "blueprintId": "postgres_11",
        "engine": "postgres",
        "engineVersion": "11.5",
        "engineDescription": "PostgreSQL",
        "engineVersionDescription": "PostgreSQL 11.5-R1",
        "isEngineDefault": true
    }
]
}

```

- Para obtener API más información, consulte [Command Reference](#).  
[GetRelationalDatabaseBlueprints](#) AWS CLI

## get-relational-database-bundles

En el siguiente ejemplo de código, se muestra cómo usar `get-relational-database-bundles`.

### AWS CLI

Para obtener los paquetes de las nuevas bases de datos relacionales

El siguiente `get-relational-database-bundles` ejemplo muestra detalles sobre todos los paquetes de bases de datos relacionales disponibles que se pueden usar para crear nuevas bases de datos relacionales en Amazon Lightsail. Tenga en cuenta que la respuesta no incluye los paquetes inactivos porque el `--include-inactive` indicador no se especifica en el comando. No puede utilizar paquetes inactivos para crear nuevas bases de datos relacionales.

```
aws lightsail get-relational-database-bundles
```

Salida:

```

{
  "bundles": [
    {

```



```
    "bundleId": "micro_2_0",
    "name": "Micro",
    "price": 15.0,
    "ramSizeInGb": 1.0,
    "diskSizeInGb": 40,
    "transferPerMonthInGb": 100,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "micro_ha_2_0",
    "name": "Micro with High Availability",
    "price": 30.0,
    "ramSizeInGb": 1.0,
    "diskSizeInGb": 40,
    "transferPerMonthInGb": 100,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "small_2_0",
    "name": "Small",
    "price": 30.0,
    "ramSizeInGb": 2.0,
    "diskSizeInGb": 80,
    "transferPerMonthInGb": 100,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "small_ha_2_0",
    "name": "Small with High Availability",
    "price": 60.0,
    "ramSizeInGb": 2.0,
    "diskSizeInGb": 80,
    "transferPerMonthInGb": 100,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
```

```
    "bundleId": "medium_2_0",
    "name": "Medium",
    "price": 60.0,
    "ramSizeInGb": 4.0,
    "diskSizeInGb": 120,
    "transferPerMonthInGb": 100,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "medium_ha_2_0",
    "name": "Medium with High Availability",
    "price": 120.0,
    "ramSizeInGb": 4.0,
    "diskSizeInGb": 120,
    "transferPerMonthInGb": 100,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "large_2_0",
    "name": "Large",
    "price": 115.0,
    "ramSizeInGb": 8.0,
    "diskSizeInGb": 240,
    "transferPerMonthInGb": 200,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  },
  {
    "bundleId": "large_ha_2_0",
    "name": "Large with High Availability",
    "price": 230.0,
    "ramSizeInGb": 8.0,
    "diskSizeInGb": 240,
    "transferPerMonthInGb": 200,
    "cpuCount": 2,
    "isEncrypted": true,
    "isActive": true
  }
]
```

```
}
```

Para obtener más información, consulte [Creación de una base de datos en Amazon Lightsail](#) en la Guía para desarrolladores de Amazon Lightsail.

- Para obtener API más información, consulte la Referencia de comandos [GetRelationalDatabaseBundles](#).AWS CLI

## get-relational-database-events

En el siguiente ejemplo de código, se muestra cómo usar `get-relational-database-events`.

### AWS CLI

Para obtener los eventos de una base de datos relacional

El siguiente `get-relational-database-events` ejemplo muestra detalles sobre los eventos de las últimas 17 horas (1020 minutos) de la base de datos relacional especificada.

```
aws lightsail get-relational-database-events \
  --relational-database-name Database-1 \
  --duration-in-minutes 1020
```

Salida:

```
{
  "relationalDatabaseEvents": [
    {
      "resource": "Database-1",
      "createdAt": 1571654146.553,
      "message": "Backing up Relational Database",
      "eventCategories": [
        "backup"
      ]
    },
    {
      "resource": "Database-1",
      "createdAt": 1571654249.98,
      "message": "Finished Relational Database backup",
      "eventCategories": [
        "backup"
      ]
    }
  ]
}
```

```
]
}
```

- Para API obtener más información, consulte la Referencia [GetRelationalDatabaseEvents](#) de AWS CLI comandos.

## get-relational-database-log-events

En el siguiente ejemplo de código, se muestra cómo usar `get-relational-database-log-events`.

### AWS CLI

Para obtener el registro de eventos de una base de datos relacional

El siguiente `get-relational-database-log-events` ejemplo muestra detalles sobre el registro especificado entre 1570733176 y 1571597176 para la base de datos relacional. Database1 La información devuelta está configurada para empezar desde head.

Le recomendamos que utilice un conversor de hora de Unix para identificar las horas de inicio y finalización.

```
aws lightsail get-relational-database-log-events \
  --relational-database-name Database1 \
  --log-stream-name error \
  --start-from-head \
  --start-time 1570733176 \
  --end-time 1571597176
```

Salida:

```
{
  "resourceLogEvents": [
    {
      "createdAt": 1570820267.0,
      "message": "2019-10-11 18:57:47 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Name or service not known"
    },
    {
      "createdAt": 1570860974.0,
      "message": "2019-10-12 06:16:14 20969 [Warning] IP address '8192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    }
  ]
}
```

```
    },
    {
      "createdAt": 1570860977.0,
      "message": "2019-10-12 06:16:17 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    {
      "createdAt": 1570860979.0,
      "message": "2019-10-12 06:16:19 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    {
      "createdAt": 1570860981.0,
      "message": "2019-10-12 06:16:21 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    {
      "createdAt": 1570860982.0,
      "message": "2019-10-12 06:16:22 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    {
      "createdAt": 1570860984.0,
      "message": "2019-10-12 06:16:24 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    {
      "createdAt": 1570860986.0,
      "message": "2019-10-12 06:16:26 20969 [Warning] IP address '192.0.2.0'
could not be resolved: Temporary failure in name resolution"
    },
    ...
  ]
  "nextBackwardToken":
  "eEXAMPLEZXJUZXh0IjoIzRwb3F3cUpRS1Q5NndMYThxelRUZlFhR3J6c2dKWEEvM2kvajZMZzVVVWpqrDN0YjFXTj
  "nextForwardToken":
  "eEXAMPLEZXJUZXh0IjoIT09Lb0Z6ZFRJbHhaNEQ5N2tPbkkwRmwwNUxPZjFTbFFwUk1Qbz1SaWgvMWVXbEk4aG56VH
}
```

- Para API obtener más información, consulte [GetRelationalDatabaseLogEvents](#) la Referencia de AWS CLI comandos.

## get-relational-database-log-streams

En el siguiente ejemplo de código, se muestra cómo usar `get-relational-database-log-streams`.

### AWS CLI

Para obtener los flujos de registro de una base de datos relacional

El siguiente `get-relational-database-log-streams` ejemplo devuelve todos los flujos de registro disponibles para la base de datos relacional especificada.

```
aws lightsail get-relational-database-log-streams \
--relational-database-name Database1
```

Salida:

```
{
  "logStreams": [
    "audit",
    "error",
    "general",
    "slowquery"
  ]
}
```

- Para API obtener más información, consulte [GetRelationalDatabaseLogStreams](#) la Referencia de AWS CLI comandos.

## get-relational-database-master-user-password

En el siguiente ejemplo de código, se muestra cómo usar `get-relational-database-master-user-password`.

### AWS CLI

Para obtener la contraseña de usuario maestro de una base de datos relacional

El siguiente `get-relational-database-master-user-password` ejemplo devuelve información sobre la contraseña del usuario maestro de la base de datos relacional especificada.

```
aws lightsail get-relational-database-master-user-password \
```

```
--relational-database-name Database-1
```

Salida:

```
{
  "masterUserPassword": "VEXAMPLEec.9qvX,_t<)Wkf)kwboM,>2",
  "createdAt": 1571259453.959
}
```

- Para API obtener más información, consulte [GetRelationalDatabaseMasterUserPassword](#) la Referencia de AWS CLI comandos.

## get-relational-database-metric-data

En el siguiente ejemplo de código, se muestra cómo usar `get-relational-database-metric-data`.

AWS CLI

Para obtener datos métricos para una base de datos relacional

El siguiente `get-relational-database-metric-data` ejemplo devuelve la suma del recuento de la métrica `DatabaseConnections` durante un período de 24 horas (86400segundos) entre 1570733176 y 1571597176 para una base de datos relacional. `Database1`

Se recomienda utilizar un conversor de tiempo de Unix para identificar las horas de inicio y finalización.

```
aws lightsail get-relational-database-metric-data \
  --relational-database-name Database1 \
  --metric-name DatabaseConnections \
  --period 86400 \
  --start-time 1570733176 \
  --end-time 1571597176 \
  --unit Count \
  --statistics Sum
```

Salida:

```
{
```

```
"metricName": "DatabaseConnections",
"metricData": [
  {
    "sum": 1.0,
    "timestamp": 1571510760.0,
    "unit": "Count"
  },
  {
    "sum": 1.0,
    "timestamp": 1570733160.0,
    "unit": "Count"
  },
  {
    "sum": 1.0,
    "timestamp": 1570992360.0,
    "unit": "Count"
  },
  {
    "sum": 0.0,
    "timestamp": 1571251560.0,
    "unit": "Count"
  },
  {
    "sum": 721.0,
    "timestamp": 1570819560.0,
    "unit": "Count"
  },
  {
    "sum": 1.0,
    "timestamp": 1571078760.0,
    "unit": "Count"
  },
  {
    "sum": 2.0,
    "timestamp": 1571337960.0,
    "unit": "Count"
  },
  {
    "sum": 684.0,
    "timestamp": 1570905960.0,
    "unit": "Count"
  },
  {
    "sum": 0.0,
```



```

        "timestamp": 1571165160.0,
        "unit": "Count"
    },
    {
        "sum": 1.0,
        "timestamp": 1571424360.0,
        "unit": "Count"
    }
]
}

```

- Para API obtener más información, consulte [GetRelationalDatabaseMetricData](#) la Referencia de AWS CLI comandos.

## get-relational-database-parameters

En el siguiente ejemplo de código, se muestra cómo usar `get-relational-database-parameters`.

### AWS CLI

Para obtener los parámetros de una base de datos relacional

El siguiente `get-relational-database-parameters` ejemplo devuelve información sobre todos los parámetros disponibles para la base de datos relacional especificada.

```

aws lightsail get-relational-database-parameters \
  --relational-database-name Database-1

```

Salida:

```

{
  "parameters": [
    {
      "allowedValues": "0,1",
      "applyMethod": "pending-reboot",
      "applyType": "dynamic",
      "dataType": "boolean",
      "description": "Automatically set all granted roles as active after the user has authenticated successfully.",
      "isModifiable": true,

```

```

        "parameterName": "activate_all_roles_on_login",
        "parameterValue": "0"
    },
    {
        "allowedValues": "0,1",
        "applyMethod": "pending-reboot",
        "applyType": "static",
        "dataType": "boolean",
        "description": "Controls whether user-defined functions that have only
an xxx symbol for the main function can be loaded",
        "isModifiable": false,
        "parameterName": "allow-suspicious-udfs"
    },
    {
        "allowedValues": "0,1",
        "applyMethod": "pending-reboot",
        "applyType": "dynamic",
        "dataType": "boolean",
        "description": "Sets the autocommit mode",
        "isModifiable": true,
        "parameterName": "autocommit"
    },
    {
        "allowedValues": "0,1",
        "applyMethod": "pending-reboot",
        "applyType": "static",
        "dataType": "boolean",
        "description": "Controls whether the server autogenerates SSL key and
certificate files in the data directory, if they do not already exist.",
        "isModifiable": false,
        "parameterName": "auto_generate_certs"
    },
    ...
}
]
}

```

Para obtener más información, consulte [Actualización de los parámetros de la base de datos en Amazon Lightsail](#) en la Guía para desarrolladores de Lightsail.

- Para obtener API más información, consulte la Referencia de comandos [GetRelationalDatabaseParameters](#).AWS CLI

## get-relational-database-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `get-relational-database-snapshot`.

### AWS CLI

Para obtener información sobre una instantánea de una base de datos relacional

En el siguiente `get-relational-database-snapshot` ejemplo, se muestran detalles sobre la instantánea de la base de datos relacional especificada.

```
aws lightsail get-relational-database-snapshot \
  --relational-database-snapshot-name Database-1-1571350042
```

Salida:

```
{
  "relationalDatabaseSnapshot": {
    "name": "Database-1-1571350042",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabaseSnapshot/0389bbad-4b85-4c3d-9EXAMPLEaee3643d2",
    "supportCode": "6EXAMPLE3362/1s-8EXAMPLE2ba7ad041451946fafc2ad19cfbd9eb2",
    "createdAt": 1571350046.238,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
    "resourceType": "RelationalDatabaseSnapshot",
    "tags": [],
    "engine": "mysql",
    "engineVersion": "8.0.16",
    "sizeInGb": 40,
    "state": "available",
    "fromRelationalDatabaseName": "Database-1",
    "fromRelationalDatabaseArn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabase/7ea932b1-b85a-4bd5-9b3e-bEXAMPLE8cc4",
    "fromRelationalDatabaseBundleId": "micro_1_0",
    "fromRelationalDatabaseBlueprintId": "mysql_8_0"
  }
}
```

- Para API obtener más información, consulte [GetRelationalDatabaseSnapshot](#) la Referencia de AWS CLI comandos.

## get-relational-database-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `get-relational-database-snapshots`.

### AWS CLI

Para obtener información sobre todas las instantáneas de bases de datos relacionales

El siguiente `get-relational-database-snapshots` ejemplo muestra detalles sobre todas las instantáneas de bases de datos relacionales de la región configurada. AWS

```
aws lightsail get-relational-database-snapshots
```

Salida:

```
{
  "relationalDatabaseSnapshots": [
    {
      "name": "Database-1-1571350042",
      "arn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabaseSnapshot/0389bbad-4b85-4c3d-9861-6EXAMPLE43d2",
      "supportCode": "6EXAMPLE3362/1s-8EXAMPLE2ba7ad041451946fafc2ad19cfbd9eb2",
      "createdAt": 1571350046.238,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "RelationalDatabaseSnapshot",
      "tags": [],
      "engine": "mysql",
      "engineVersion": "8.0.16",
      "sizeInGb": 40,
      "state": "available",
      "fromRelationalDatabaseName": "Database-1",
      "fromRelationalDatabaseArn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabase/7ea932b1-b85a-4bd5-9b3e-bEXAMPLE8cc4",
      "fromRelationalDatabaseBundleId": "micro_1_0",
    }
  ]
}
```

```

        "fromRelationalDatabaseBlueprintId": "mysql_8_0"
    },
    {
        "name": "Database1-Console",
        "arn": "arn:aws:lightsail:us-
west-2:111122223333:RelationalDatabaseSnapshot/8b94136e-06ec-4b1a-
a3fb-5EXAMPLEe1e9",
        "supportCode": "6EXAMPLE3362/
1s-9EXAMPLE14b000d34c8d1c432734e137612d5b5c",
        "createdAt": 1571249981.025,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "resourceType": "RelationalDatabaseSnapshot",
        "tags": [
            {
                "key": "test"
            }
        ],
        "engine": "mysql",
        "engineVersion": "5.6.44",
        "sizeInGb": 40,
        "state": "available",
        "fromRelationalDatabaseName": "Database1",
        "fromRelationalDatabaseArn": "arn:aws:lightsail:us-
west-2:111122223333:RelationalDatabase/a6161cb7-4535-4f16-9dcf-8EXAMPLE3d4e",
        "fromRelationalDatabaseBundleId": "micro_1_0",
        "fromRelationalDatabaseBlueprintId": "mysql_5_6"
    }
]
}

```

- Para API obtener más información, consulte la Referencia [GetRelationalDatabaseSnapshots](#) de AWS CLI comandos.

## get-relational-database

En el siguiente ejemplo de código, se muestra cómo usar `get-relational-database`.

### AWS CLI

Para obtener información sobre una base de datos relacional

En el siguiente `get-relational-database` ejemplo, se muestran detalles sobre la base de datos relacional especificada.

```
aws lightsail get-relational-database \  
--relational-database-name Database-1
```

Salida:

```
{  
  "relationalDatabase": {  
    "name": "Database-1",  
    "arn": "arn:aws:lightsail:us-  
west-2:111122223333:RelationalDatabase/7ea932b1-b85a-4bd5-9b3e-bEXAMPLE8cc4",  
    "supportCode": "6EXAMPLE3362/1s-9EXAMPLE8ad863723b62cc8901a8aa6e794ae0d2",  
    "createdAt": 1571259453.795,  
    "location": {  
      "availabilityZone": "us-west-2a",  
      "regionName": "us-west-2"  
    },  
    "resourceType": "RelationalDatabase",  
    "tags": [],  
    "relationalDatabaseBlueprintId": "mysql_8_0",  
    "relationalDatabaseBundleId": "micro_1_0",  
    "masterDatabaseName": "dbmaster",  
    "hardware": {  
      "cpuCount": 1,  
      "diskSizeInGb": 40,  
      "ramSizeInGb": 1.0  
    },  
    "state": "available",  
    "backupRetentionEnabled": false,  
    "pendingModifiedValues": {},  
    "engine": "mysql",  
    "engineVersion": "8.0.16",  
    "masterUsername": "dbmasteruser",  
    "parameterApplyStatus": "in-sync",  
    "preferredBackupWindow": "10:01-10:31",  
    "preferredMaintenanceWindow": "sat:11:14-sat:11:44",  
    "publiclyAccessible": true,  
    "masterEndpoint": {  
      "port": 3306,  
      "address": "1s-9EXAMPLE8ad863723b62ccEXAMPLEa6e794ae0d2.czowadgeezqi.us-  
west-2.rds.amazonaws.com"
```

```
    },  
    "pendingMaintenanceActions": []  
  }  
}
```

- Para API obtener más información, consulte [GetRelationalDatabase](#) la Referencia de AWS CLI comandos.

## get-relational-databases

En el siguiente ejemplo de código, se muestra cómo usar `get-relational-databases`.

### AWS CLI

Para obtener información sobre todas las bases de datos relacionales

En el siguiente `get-relational-databases` ejemplo, se muestran detalles sobre todas las bases de datos relacionales de la región configurada AWS .

```
aws lightsail get-relational-databases
```

Salida:

```
{  
  "relationalDatabases": [  
    {  
      "name": "MySQL",  
      "arn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabase/8529020c-3ab9-4d51-92af-5EXAMPLE8979",  
      "supportCode": "6EXAMPLE3362/1s-3EXAMPLEa995d8c3b06b4501356e5f2f28e1aeba",  
      "createdAt": 1554306019.155,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "resourceType": "RelationalDatabase",  
      "tags": [],  
      "relationalDatabaseBlueprintId": "mysql_8_0",  
      "relationalDatabaseBundleId": "micro_1_0",  
      "masterDatabaseName": "dbmaster",  
      "hardware": {
```

```

        "cpuCount": 1,
        "diskSizeInGb": 40,
        "ramSizeInGb": 1.0
    },
    "state": "available",
    "backupRetentionEnabled": true,
    "pendingModifiedValues": {},
    "engine": "mysql",
    "engineVersion": "8.0.15",
    "latestRestorableTime": 1571686200.0,
    "masterUsername": "dbmasteruser",
    "parameterApplyStatus": "in-sync",
    "preferredBackupWindow": "07:51-08:21",
    "preferredMaintenanceWindow": "tue:12:18-tue:12:48",
    "publiclyAccessible": true,
    "masterEndpoint": {
        "port": 3306,
        "address":
"ls-3EXAMPLEa995d8c3b06b4501356e5f2fEXAMPLEa.czowadgeezqi.us-
west-2.rds.amazonaws.com"
    },
    "pendingMaintenanceActions": []
},
{
    "name": "Postgres",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabase/
e9780b6b-d0ab-4af2-85f1-1EXAMPLEac68",
    "supportCode": "6EXAMPLE3362/
ls-3EXAMPLEb4ffffb5cec056220c734713e14bd5fcd",
    "createdAt": 1554306000.814,
    "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
    },
    "resourceType": "RelationalDatabase",
    "tags": [],
    "relationalDatabaseBlueprintId": "postgres_11",
    "relationalDatabaseBundleId": "micro_1_0",
    "masterDatabaseName": "dbmaster",
    "hardware": {
        "cpuCount": 1,
        "diskSizeInGb": 40,
        "ramSizeInGb": 1.0
    },

```



```

    "state": "available",
    "backupRetentionEnabled": true,
    "pendingModifiedValues": {},
    "engine": "postgres",
    "engineVersion": "11.1",
    "latestRestorableTime": 1571686339.0,
    "masterUsername": "dbmasteruser",
    "parameterApplyStatus": "in-sync",
    "preferredBackupWindow": "06:19-06:49",
    "preferredMaintenanceWindow": "sun:10:19-sun:10:49",
    "publiclyAccessible": false,
    "masterEndpoint": {
      "port": 5432,
      "address":
"1s-3EXAMPLEb4ffffb5cec056220c734713eEXAMPLEd.czowadgeezqi.us-
west-2.rds.amazonaws.com"
    },
    "pendingMaintenanceActions": []
  }
]
}

```

- Para API obtener más información, consulte [GetRelationalDatabases](#) la Referencia de AWS CLI comandos.

## get-static-ip

En el siguiente ejemplo de código, se muestra cómo usar `get-static-ip`.

AWS CLI

Para obtener información sobre una IP estática

En el siguiente `get-static-ip` ejemplo, se muestran detalles sobre la IP estática especificada.

```

aws lightsail get-static-ip \
  --static-ip-name StaticIp-1

```

Salida:

```
{
```

```

    "staticIp": {
      "name": "StaticIp-1",
      "arn": "arn:aws:lightsail:us-
west-2:111122223333:StaticIp/2257cd76-1f0e-4ac0-82e2-2EXAMPLE23ad",
      "supportCode": "6EXAMPLE3362/192.0.2.0",
      "createdAt": 1571071325.076,
      "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
      },
      "resourceType": "StaticIp",
      "ipAddress": "192.0.2.0",
      "isAttached": false
    }
  }
}

```

- Para API obtener más información, consulte [GetStaticIp](#) la Referencia de AWS CLI comandos.

## get-static-ips

En el siguiente ejemplo de código, se muestra cómo usar `get-static-ips`.

### AWS CLI

Para obtener información sobre toda la estática IPs

En el siguiente `get-static-ips` ejemplo, se muestran detalles sobre toda la estática IPs de la AWS región configurada.

```
aws lightsail get-static-ips
```

Salida:

```

{
  "staticIps": [
    {
      "name": "StaticIp-1",
      "arn": "arn:aws:lightsail:us-
west-2:111122223333:StaticIp/2257cd76-1f0e-4ac0-8EXAMPLE16f9423ad",
      "supportCode": "6EXAMPLE3362/192.0.2.0",
      "createdAt": 1571071325.076,
      "location": {

```

```

        "availabilityZone": "all",
        "regionName": "us-west-2"
    },
    "resourceType": "StaticIp",
    "ipAddress": "192.0.2.0",
    "isAttached": false
},
{
    "name": "StaticIP-2",
    "arn": "arn:aws:lightsail:us-west-2:111122223333:StaticIp/c61edb40-
e5f0-4fd6-ae7c-8EXAMPLE19f8",
    "supportCode": "6EXAMPLE3362/192.0.2.2",
    "createdAt": 1568305385.681,
    "location": {
        "availabilityZone": "all",
        "regionName": "us-west-2"
    },
    "resourceType": "StaticIp",
    "ipAddress": "192.0.2.2",
    "attachedTo": "WordPress-1",
    "isAttached": true
}
]
}

```

- Para API obtener más información, consulte [GetStaticIps](#) la Referencia de AWS CLI comandos.

## is-vpc-peered

En el siguiente ejemplo de código, se muestra cómo usar `is-vpc-peered`.

### AWS CLI

Para identificar si su nube privada virtual de Amazon Lightsail está sincronizada

El siguiente `is-vpc-peered` ejemplo devuelve el estado de interconexión de la VPC nube privada virtual de Amazon Lightsail () para la región especificada. AWS

```
aws lightsail is-vpc-peered \
  --region us-west-2
```

Salida:

```
{
  "isPeered": true
}
```

- Para API obtener más información, consulte la Referencia de [IsVpcPeered](#) comandos AWS CLI .

## open-instance-public-ports

En el siguiente ejemplo de código, se muestra cómo usar `open-instance-public-ports`.

### AWS CLI

Para abrir los puertos de firewall de una instancia

En el siguiente `open-instance-public-ports` ejemplo, se abre el TCP puerto 22 en la instancia especificada.

```
aws lightsail open-instance-public-ports \
  --instance-name MEAN-2 \
  --port-info fromPort=22,protocol=TCP,toPort=22
```

Salida:

```
{
  "operation": {
    "id": "719744f0-a022-46f2-9f11-6EXAMPLE4642",
    "resourceName": "MEAN-2",
    "resourceType": "Instance",
    "createdAt": 1571072906.849,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": true,
    "operationDetails": "22/tcp",
    "operationType": "OpenInstancePublicPorts",
    "status": "Succeeded",
    "statusChangedAt": 1571072906.849
  }
}
```

- Para API obtener más información, consulte [OpenInstancePublicPorts](#) la Referencia de AWS CLI comandos.

## peer-vpc

En el siguiente ejemplo de código, se muestra cómo usar `peer-vpc`.

### AWS CLI

Para comparar la nube privada virtual de Amazon Lightsail

El siguiente `peer-vpc` ejemplo compara la nube privada virtual de Amazon Lightsail VPC () de la región especificada. AWS

```
aws lightsail peer-vpc \  
  --region us-west-2
```

Salida:

```
{  
  "operation": {  
    "id": "787e846a-54ac-497f-bce2-9EXAMPLE5d91",  
    "resourceName": "vpc-0EXAMPLEa5261efb3",  
    "resourceType": "PeeredVpc",  
    "createdAt": 1571694233.104,  
    "location": {  
      "availabilityZone": "all",  
      "regionName": "us-west-2"  
    },  
    "isTerminal": true,  
    "operationDetails": "vpc-e2b3eb9b",  
    "operationType": "PeeredVpc",  
    "status": "Succeeded",  
    "statusChangedAt": 1571694233.104  
  }  
}
```

- Para API obtener más información, consulte la Referencia [PeerVpc](#) de AWS CLI comandos.

## reboot-instance

En el siguiente ejemplo de código, se muestra cómo usar `reboot-instance`.

### AWS CLI

Para reiniciar una instancia

En el siguiente `reboot-instance` ejemplo, se reinicia la instancia especificada.

```
aws lightsail reboot-instance \  
  --instance-name MEAN-1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "2b679f1c-8b71-4bb4-8e97-8EXAMPLEed93",  
      "resourceName": "MEAN-1",  
      "resourceType": "Instance",  
      "createdAt": 1571694445.49,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationDetails": "",  
      "operationType": "RebootInstance",  
      "status": "Succeeded",  
      "statusChangedAt": 1571694445.49  
    }  
  ]  
}
```

- Para API obtener más información, consulte [RebootInstance](#) la Referencia de AWS CLI comandos.

## reboot-relational-database

En el siguiente ejemplo de código, se muestra cómo usar `reboot-relational-database`.

## AWS CLI

Para reiniciar una base de datos relacional

En el siguiente `reboot-relational-database` ejemplo, se reinicia la base de datos relacional especificada.

```
aws lightsail reboot-relational-database \  
  --relational-database-name Database-1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "e4c980c0-3137-496c-9c91-1EXAMPLEdec2",  
      "resourceName": "Database-1",  
      "resourceType": "RelationalDatabase",  
      "createdAt": 1571694532.91,  
      "location": {  
        "availabilityZone": "us-west-2a",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": false,  
      "operationDetails": "",  
      "operationType": "RebootRelationalDatabase",  
      "status": "Started",  
      "statusChangedAt": 1571694532.91  
    }  
  ]  
}
```

- Para API obtener más información, consulte la Referencia [RebootRelationalDatabase](#) de AWS CLI comandos.

## **release-static-ip**

En el siguiente ejemplo de código, se muestra cómo usar `release-static-ip`.

## AWS CLI

Para eliminar una IP estática

En el siguiente `release-static-ip` ejemplo, se elimina la IP estática especificada.

```
aws lightsail release-static-ip \  
  --static-ip-name StaticIp-1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "e374c002-dc6d-4c7f-919f-2EXAMPLE13ce",  
      "resourceName": "StaticIp-1",  
      "resourceType": "StaticIp",  
      "createdAt": 1571694962.003,  
      "location": {  
        "availabilityZone": "all",  
        "regionName": "us-west-2"  
      },  
      "isTerminal": true,  
      "operationType": "ReleaseStaticIp",  
      "status": "Succeeded",  
      "statusChangedAt": 1571694962.003  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ReleaseStaticIp](#) la Referencia de AWS CLI comandos.

## start-instance

En el siguiente ejemplo de código, se muestra cómo usar `start-instance`.

AWS CLI

Para iniciar una instancia

En el siguiente `start-instance` ejemplo, se inicia la instancia especificada.

```
aws lightsail start-instance \  
  --instance-name WordPress-1
```



Salida:

```
{
  "operations": [
    {
      "id": "f88d2a93-7cea-4165-afce-2d688cb18f23",
      "resourceName": "WordPress-1",
      "resourceType": "Instance",
      "createdAt": 1571695583.463,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      },
      "isTerminal": false,
      "operationType": "StartInstance",
      "status": "Started",
      "statusChangedAt": 1571695583.463
    }
  ]
}
```

- Para API obtener más información, consulte [StartInstance](#) la Referencia de AWS CLI comandos.

## start-relational-database

En el siguiente ejemplo de código, se muestra cómo usar `start-relational-database`.

AWS CLI

Para iniciar una base de datos relacional

En el siguiente `start-relational-database` ejemplo, se inicia la base de datos relacional especificada.

```
aws lightsail start-relational-database \
  --relational-database-name Database-1
```

Salida:

```
{
```

```
"operations": [  
  {  
    "id": "4d5294ec-a38a-4fda-9e37-aEXAMPLE0d24",  
    "resourceName": "Database-1",  
    "resourceType": "RelationalDatabase",  
    "createdAt": 1571695998.822,  
    "location": {  
      "availabilityZone": "us-west-2a",  
      "regionName": "us-west-2"  
    },  
    "isTerminal": false,  
    "operationType": "StartRelationalDatabase",  
    "status": "Started",  
    "statusChangedAt": 1571695998.822  
  }  
]
```

- Para API obtener más información, consulte [StartRelationalDatabase](#) la Referencia de AWS CLI comandos.

## stop-instance

En el siguiente ejemplo de código, se muestra cómo usar `stop-instance`.

### AWS CLI

Para detener una instancia

El siguiente `stop-instance` ejemplo detiene la instancia especificada.

```
aws lightsail stop-instance \  
--instance-name WordPress-1
```

Salida:

```
{  
  "operations": [  
    {  
      "id": "265357e2-2943-4d51-888a-1EXAMPLE7585",  
      "resourceName": "WordPress-1",
```

```

    "resourceType": "Instance",
    "createdAt": 1571695471.134,
    "location": {
      "availabilityZone": "us-west-2a",
      "regionName": "us-west-2"
    },
    "isTerminal": false,
    "operationType": "StopInstance",
    "status": "Started",
    "statusChangedAt": 1571695471.134
  }
]
}

```

- Para API obtener más información, consulte [StopInstance](#) la Referencia de AWS CLI comandos.

## stop-relational-database

En el siguiente ejemplo de código, se muestra cómo usar `stop-relational-database`.

### AWS CLI

Para detener una base de datos relacional

El siguiente `stop-relational-database` ejemplo detiene la base de datos relacional especificada.

```

aws lightsail stop-relational-database \
  --relational-database-name Database-1

```

Salida:

```

{
  "operations": [
    {
      "id": "cc559c19-4adb-41e4-b75b-5EXAMPLE4e61",
      "resourceName": "Database-1",
      "resourceType": "RelationalDatabase",
      "createdAt": 1571695526.29,
      "location": {
        "availabilityZone": "us-west-2a",
        "regionName": "us-west-2"
      }
    }
  ]
}

```

```

        },
        "isTerminal": false,
        "operationType": "StopRelationalDatabase",
        "status": "Started",
        "statusChangedAt": 1571695526.29
    }
]
}

```

- Para API obtener más información, consulte [StopRelationalDatabase](#) la Referencia de AWS CLI comandos.

## unpeer-vpc

En el siguiente ejemplo de código, se muestra cómo usar `unpeer-vpc`.

### AWS CLI

Para eliminar la nube privada virtual de Amazon Lightsail

En el siguiente `unpeer-vpc` ejemplo, se desvincula la VPC nube privada virtual de Amazon Lightsail () de la región especificada. AWS

```

aws lightsail unpeer-vpc \
  --region us-west-2

```

Salida:

```

{
  "operation": {
    "id": "531aca64-7157-47ab-84c6-eEXAMPLEd898",
    "resourceName": "vpc-0EXAMPLEa5261efb3",
    "resourceType": "PeeredVpc",
    "createdAt": 1571694109.945,
    "location": {
      "availabilityZone": "all",
      "regionName": "us-west-2"
    },
  },
  "isTerminal": true,
  "operationDetails": "vpc-e2b3eb9b",
  "operationType": "UnpeeredVpc",
}

```

```
    "status": "Succeeded",  
    "statusChangedAt": 1571694109.945  
  }  
}
```

- Para API obtener más información, consulte la Referencia de [UnpeerVpccomandos AWS CLI](#) .

## Ejemplos de Macie usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el AWS Command Line Interface uso de Macie.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde encontrará instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **describe-buckets**

En el siguiente ejemplo de código, se muestra cómo usar `describe-buckets`.

AWS CLI

Para consultar datos sobre uno o más depósitos de S3 que Amazon Macie supervisa y analiza para su cuenta

En el siguiente `describe-buckets` ejemplo, se consultan los metadatos de todos los buckets de S3 cuyos nombres comiencen por MY-S3 y se encuentren en la región actual. AWS

```
aws macie2 describe-buckets \  
  --criteria '{"bucketName":{"prefix":"my-S3"}}'
```

## Salida:

```
{
  "buckets": [
    {
      "accountId": "123456789012",
      "allowsUnencryptedObjectUploads": "FALSE",
      "bucketArn": "arn:aws:s3:::MY-S3-DOC-EXAMPLE-BUCKET1",
      "bucketCreatedAt": "2020-05-18T19:54:00+00:00",
      "bucketName": "MY-S3-DOC-EXAMPLE-BUCKET1",
      "classifiableObjectCount": 13,
      "classifiableSizeInBytes": 1592088,
      "jobDetails": {
        "isDefinedInJob": "TRUE",
        "isMonitoredByJob": "TRUE",
        "lastJobId": "08c81dc4a2f3377fae45c9ddaexample",
        "lastJobRunTime": "2021-04-26T14:55:30.270000+00:00"
      },
      "lastAutomatedDiscoveryTime": "2022-12-10T19:11:25.364000+00:00",
      "lastUpdated": "2022-12-13T07:33:06.337000+00:00",
      "objectCount": 13,
      "objectCountByEncryptionType": {
        "customerManaged": 0,
        "kmsManaged": 2,
        "s3Managed": 7,
        "unencrypted": 4,
        "unknown": 0
      },
      "publicAccess": {
        "effectivePermission": "NOT_PUBLIC",
        "permissionConfiguration": {
          "accountLevelPermissions": {
            "blockPublicAccess": {
              "blockPublicAcls": true,
              "blockPublicPolicy": true,
              "ignorePublicAcls": true,
              "restrictPublicBuckets": true
            }
          },
          "bucketLevelPermissions": {
            "accessControlList": {
              "allowsPublicReadAccess": false,
              "allowsPublicWriteAccess": false
            }
          }
        }
      }
    }
  ]
}
```

```
        "blockPublicAccess": {
            "blockPublicAcls": true,
            "blockPublicPolicy": true,
            "ignorePublicAcls": true,
            "restrictPublicBuckets": true
        },
        "bucketPolicy": {
            "allowsPublicReadAccess": false,
            "allowsPublicWriteAccess": false
        }
    }
},
"region": "us-west-2",
"replicationDetails": {
    "replicated": false,
    "replicatedExternally": false,
    "replicationAccounts": []
},
"sensitivityScore": 78,
"serverSideEncryption": {
    "kmsMasterKeyId": null,
    "type": "NONE"
},
"sharedAccess": "NOT_SHARED",
"sizeInBytes": 4549746,
"sizeInBytesCompressed": 0,
"tags": [
    {
        "key": "Division",
        "value": "HR"
    },
    {
        "key": "Team",
        "value": "Recruiting"
    }
],
"unclassifiableObjectCount": {
    "fileType": 0,
    "storageClass": 0,
    "total": 0
},
"unclassifiableObjectSizeInBytes": {
    "fileType": 0,
```

```
        "storageClass": 0,
        "total": 0
    },
    "versioning": true
},
{
    "accountId": "123456789012",
    "allowsUnencryptedObjectUploads": "TRUE",
    "bucketArn": "arn:aws:s3:::MY-S3-DOC-EXAMPLE-BUCKET2",
    "bucketCreatedAt": "2020-11-25T18:24:38+00:00",
    "bucketName": "MY-S3-DOC-EXAMPLE-BUCKET2",
    "classifiableObjectCount": 8,
    "classifiableSizeInBytes": 133810,
    "jobDetails": {
        "isDefinedInJob": "TRUE",
        "isMonitoredByJob": "FALSE",
        "lastJobId": "188d4f6044d621771ef7d65f2example",
        "lastJobRunTime": "2021-04-09T19:37:11.511000+00:00"
    },
    "lastAutomatedDiscoveryTime": "2022-12-12T19:11:25.364000+00:00",
    "lastUpdated": "2022-12-13T07:33:06.337000+00:00",
    "objectCount": 8,
    "objectCountByEncryptionType": {
        "customerManaged": 0,
        "kmsManaged": 0,
        "s3Managed": 8,
        "unencrypted": 0,
        "unknown": 0
    },
    "publicAccess": {
        "effectivePermission": "NOT_PUBLIC",
        "permissionConfiguration": {
            "accountLevelPermissions": {
                "blockPublicAccess": {
                    "blockPublicAcls": true,
                    "blockPublicPolicy": true,
                    "ignorePublicAcls": true,
                    "restrictPublicBuckets": true
                }
            },
            "bucketLevelPermissions": {
                "accessControlList": {
                    "allowsPublicReadAccess": false,
                    "allowsPublicWriteAccess": false
                }
            }
        }
    }
}
```



```
    },
    "blockPublicAccess": {
      "blockPublicAcls": true,
      "blockPublicPolicy": true,
      "ignorePublicAcls": true,
      "restrictPublicBuckets": true
    },
    "bucketPolicy": {
      "allowsPublicReadAccess": false,
      "allowsPublicWriteAccess": false
    }
  }
},
"region": "us-west-2",
"replicationDetails": {
  "replicated": false,
  "replicatedExternally": false,
  "replicationAccounts": []
},
"sensitivityScore": 95,
"serverSideEncryption": {
  "kmsMasterKeyId": null,
  "type": "AES256"
},
"sharedAccess": "EXTERNAL",
"sizeInBytes": 175978,
"sizeInBytesCompressed": 0,
"tags": [
  {
    "key": "Division",
    "value": "HR"
  },
  {
    "key": "Team",
    "value": "Recruiting"
  }
],
"unclassifiableObjectCount": {
  "fileType": 3,
  "storageClass": 0,
  "total": 3
},
"unclassifiableObjectSizeInBytes": {
```

```
        "fileType": 2999826,  
        "storageClass": 0,  
        "total": 2999826  
    },  
    "versioning": true  
  }  
]  
}
```

Para obtener más información, consulte [Filtrar el inventario de cubos de S3](#) en la Guía del usuario de Amazon Macie.

- Para API obtener más información, consulte [DescribeBuckets](#) la Referencia de AWS CLI comandos.

## Ejemplos de Grafana gestionados por Amazon utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface Grafana gestionado por Amazon.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **list-workspaces**

En el siguiente ejemplo de código, se muestra cómo usar `list-workspaces`.

AWS CLI

Para enumerar los espacios de trabajo de la cuenta en la región especificada por la credencial del usuario

El siguiente `list-workspaces` ejemplo muestra los espacios de trabajo de Grafana para la región de la cuenta.

```
aws grafana list-workspaces
```

Salida:

```
{
  "workspaces": [
    {
      "authentication": {
        "providers": [
          "AWS_SSO"
        ]
      },
      "created": "2022-04-04T16:20:21.796000-07:00",
      "description": "to test tags",
      "endpoint": "g-949e7b44df.grafana-workspace.us-east-1.amazonaws.com",
      "grafanaVersion": "8.2",
      "id": "g-949e7b44df",
      "modified": "2022-04-04T16:20:21.796000-07:00",
      "name": "testtag2",
      "notificationDestinations": [
        "SNS"
      ],
      "status": "ACTIVE"
    },
    {
      "authentication": {
        "providers": [
          "AWS_SSO"
        ]
      },
      "created": "2022-04-20T10:22:15.115000-07:00",
      "description": "ww",
      "endpoint": "g-bffa51ed1b.grafana-workspace.us-east-1.amazonaws.com",
      "grafanaVersion": "8.2",
      "id": "g-bffa51ed1b",
      "modified": "2022-04-20T10:22:15.115000-07:00",
      "name": "ww",
      "notificationDestinations": [
        "SNS"
      ],
    },
  ],
}
```

```

        "status": "ACTIVE"
      }
    ]
  }

```

- Para API obtener más información, consulte la Referencia [ListWorkspaces](#) de AWS CLI comandos.

## MediaConnect ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with MediaConnect.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **add-flow-outputs**

En el siguiente ejemplo de código, se muestra cómo usar `add-flow-outputs`.

AWS CLI

Para añadir salidas a un flujo

El siguiente `add-flow-outputs` ejemplo agrega salidas al flujo especificado.

```

aws mediacconnect add-flow-outputs \
--flow-arn arn:aws:mediacconnect:us-east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
--outputs Description='NYC stream',Destination=192.0.2.12,Name=NYC,Port=3333,Protocol=rtp-

```

```
fec,SmoothingLatency=100 Description='LA
stream',Destination=203.0.113.9,Name=LA,Port=4444,Protocol=rtp-
fec,SmoothingLatency=100
```

Salida:

```
{
  "Outputs": [
    {
      "Port": 3333,
      "OutputArn": "arn:aws:mediacconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC",
      "Name": "NYC",
      "Description": "NYC stream",
      "Destination": "192.0.2.12",
      "Transport": {
        "Protocol": "rtp-fec",
        "SmoothingLatency": 100
      }
    },
    {
      "Port": 4444,
      "OutputArn": "arn:aws:mediacconnect:us-
east-1:111122223333:output:2-987655dEF67hiJ89-c34de5fG678h:LA",
      "Name": "LA",
      "Description": "LA stream",
      "Destination": "203.0.113.9",
      "Transport": {
        "Protocol": "rtp-fec",
        "SmoothingLatency": 100
      }
    }
  ],
  "FlowArn": "arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame"
}
```

Para obtener más información, consulte [Añadir salidas a un flujo](#) en la Guía del MediaConnect usuario de AWS Elemental.

- Para API obtener más información, consulte [AddFlowOutputs](#) la Referencia de AWS CLI comandos.

## create-flow

En el siguiente ejemplo de código, se muestra cómo usar `create-flow`.

### AWS CLI

Para crear un flujo

El siguiente `create-flow` ejemplo crea un flujo con la configuración especificada.

```
aws mediacconnect create-flow \  
  --availability-zone us-west-2c \  
  --name ExampleFlow \  
  --source Description='Example source,  
backup',IngestPort=1055,Name=BackupSource,Protocol=rtp,WhitelistCidr=10.24.34.0/23
```

Salida:

```
{  
  "Flow": {  
    "FlowArn": "arn:aws:mediacconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:ExampleFlow",  
    "AvailabilityZone": "us-west-2c",  
    "EgressIp": "54.245.71.21",  
    "Source": {  
      "IngestPort": 1055,  
      "SourceArn": "arn:aws:mediacconnect:us-  
east-1:123456789012:source:2-3aBC45dEF67hiJ89-c34de5fG678h:BackupSource",  
      "Transport": {  
        "Protocol": "rtp",  
        "MaxBitrate": 80000000  
      },  
      "Description": "Example source, backup",  
      "IngestIp": "54.245.71.21",  
      "WhitelistCidr": "10.24.34.0/23",  
      "Name": "mySource"  
    },  
    "Entitlements": [],  
    "Name": "ExampleFlow",  
    "Outputs": [],  
    "Status": "STANDBY",  
    "Description": "Example source, backup"  
  }  
}
```

```
}
```

Para obtener más información, consulte [Creación de un flujo](#) en la Guía del MediaConnect usuario de AWS Elemental.

- Para API obtener más información, consulte [CreateFlow](#) la Referencia de AWS CLI comandos.

## delete-flow

En el siguiente ejemplo de código, se muestra cómo usar `delete-flow`.

### AWS CLI

Para eliminar un flujo

En el siguiente `delete-flow` ejemplo, se elimina el flujo especificado.

```
aws mediaconnect delete-flow \  
  --flow-arn arn:aws:mediaconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow
```

Salida:

```
{  
  "FlowArn": "arn:aws:mediaconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow",  
  "Status": "DELETING"  
}
```

Para obtener más información, consulte [Eliminar un flujo](#) en la Guía del MediaConnect usuario de AWS Elemental.

- Para API obtener más información, consulte [DeleteFlow](#) la Referencia de AWS CLI comandos.

## describe-flow

En el siguiente ejemplo de código, se muestra cómo usar `describe-flow`.

### AWS CLI

Para ver los detalles de un flujo

El siguiente describe-flow ejemplo muestra los detalles del flujo especificado, como la zona de disponibilidadARN, el estado, el origen, las autorizaciones y las salidas.

```
aws mediacconnect describe-flow \
  --flow-arn arn:aws:mediacconnect:us-
  east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow
```

Salida:

```
{
  "Flow": {
    "EgressIp": "54.201.4.39",
    "AvailabilityZone": "us-west-2c",
    "Status": "ACTIVE",
    "FlowArn": "arn:aws:mediacconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow",
    "Entitlements": [
      {
        "EntitlementArn": "arn:aws:mediacconnect:us-
west-2:123456789012:entitlement:1-AaBb11CcDd22EeFf-34DE5fG12AbC:MyEntitlement",
        "Description": "Assign to this account",
        "Name": "MyEntitlement",
        "Subscribers": [
          "444455556666"
        ]
      }
    ],
    "Description": "NYC awards show",
    "Name": "AwardsShow",
    "Outputs": [
      {
        "Port": 2355,
        "Name": "NYC",
        "Transport": {
          "SmoothingLatency": 0,
          "Protocol": "rtp-fec"
        },
        "OutputArn": "arn:aws:mediacconnect:us-
east-1:123456789012:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC",
        "Destination": "192.0.2.0"
      },
      {
        "Port": 3025,
```



```

        "Name": "LA",
        "Transport": {
            "SmoothingLatency": 0,
            "Protocol": "rtp-fec"
        },
        "OutputArn": "arn:aws:mediacconnect:us-
east-1:123456789012:output:2-987655dEF67hiJ89-c34de5fG678h:LA",
        "Destination": "192.0.2.0"
    }
],
"Source": {
    "IngestIp": "54.201.4.39",
    "SourceArn": "arn:aws:mediacconnect:us-
east-1:123456789012:source:3-4aBC56dEF78hiJ90-4de5fG6Hi78Jk:ShowSource",
    "Transport": {
        "MaxBitrate": 80000000,
        "Protocol": "rtp"
    },
    "IngestPort": 1069,
    "Description": "Saturday night show",
    "Name": "ShowSource",
    "WhitelistCidr": "10.24.34.0/23"
}
}
}

```

Para obtener más información, consulte [Visualización de los detalles de un flujo](#) en la Guía del MediaConnect usuario de AWS Elemental.

- Para API obtener más información, consulte [DescribeFlow](#) la Referencia de AWS CLI comandos.

## grant-flow-entitlements

En el siguiente ejemplo de código, se muestra cómo usar `grant-flow-entitlements`.

### AWS CLI

Para conceder un derecho a un flujo

El siguiente `grant-flow-entitlements` ejemplo otorga el derecho al flujo existente especificado a compartir su contenido con otra cuenta. AWS

```
aws mediacconnect grant-flow-entitlements \
  --flow-arn arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
  --entitlements Description='For
AnyCompany',Encryption={"Algorithm=aes128,KeyType=static-
key,RoleArn=arn:aws:iam::111122223333:role/MediaConnect-
ASM,SecretArn=arn:aws:secretsmanager:us-
west-2:111122223333:secret:mySecret1"},Name=AnyCompany_Entitlement,Subscribers=444455556666
Description='For Example Corp',Name=ExampleCorp,Subscribers=777788889999
```

Salida:

```
{
  "Entitlements": [
    {
      "Name": "AnyCompany_Entitlement",
      "EntitlementArn": "arn:aws:mediacconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement",
      "Subscribers": [
        "444455556666"
      ],
      "Description": "For AnyCompany",
      "Encryption": {
        "SecretArn": "arn:aws:secretsmanager:us-
west-2:111122223333:secret:mySecret1",
        "Algorithm": "aes128",
        "RoleArn": "arn:aws:iam::111122223333:role/MediaConnect-ASM",
        "KeyType": "static-key"
      }
    },
    {
      "Name": "ExampleCorp",
      "EntitlementArn": "arn:aws:mediacconnect:us-
west-2:111122223333:entitlement:1-3333cccc4444dddd-1111aaaa2222:ExampleCorp",
      "Subscribers": [
        "777788889999"
      ],
      "Description": "For Example Corp"
    }
  ],
  "FlowArn": "arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame"
```

```
}
```

Para obtener más información, consulta Cómo [conceder un derecho a un flujo en la Guía del usuario de Elemental.AWS MediaConnect](#)

- Para API obtener más información, consulte la Referencia [GrantFlowEntitlements](#) de AWS CLI comandos.

## list-entitlements

En el siguiente ejemplo de código, se muestra cómo usar `list-entitlements`.

### AWS CLI

Para ver una lista de derechos

En el siguiente `list-entitlements` ejemplo, se muestra una lista de todos los derechos que se han concedido a la cuenta.

```
aws mediacconnect list-entitlements
```

Salida:

```
{
  "Entitlements": [
    {
      "EntitlementArn": "arn:aws:mediacconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:MyEntitlement",
      "EntitlementName": "MyEntitlement"
    }
  ]
}
```

Para obtener más información, consulte [ListEntitlements](#) la Referencia AWS elemental MediaConnect API.

- Para API obtener más información, consulte [ListEntitlements](#) la Referencia de AWS CLI comandos.

## list-flows

En el siguiente ejemplo de código, se muestra cómo usar `list-flows`.

## AWS CLI

Para ver una lista de flujos

En el siguiente `list-flows` ejemplo, se muestra una lista de flujos.

```
aws mediaconnect list-flows
```

Salida:

```
{
  "Flows": [
    {
      "Status": "STANDBY",
      "SourceType": "OWNED",
      "AvailabilityZone": "us-west-2a",
      "Description": "NYC awards show",
      "Name": "AwardsShow",
      "FlowArn": "arn:aws:mediaconnect:us-east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow"
    },
    {
      "Status": "STANDBY",
      "SourceType": "OWNED",
      "AvailabilityZone": "us-west-2c",
      "Description": "LA basketball game",
      "Name": "BasketballGame",
      "FlowArn": "arn:aws:mediaconnect:us-east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BasketballGame"
    }
  ]
}
```

Para obtener más información, consulte [Visualización de una lista de flujos](#) en la Guía del MediaConnect usuario de AWS Elemental.

- Para API obtener más información, consulte [ListFlows](#) la Referencia de AWS CLI comandos.

## `list-tags-for-resource`

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

## AWS CLI

Para enumerar las etiquetas de un MediaConnect recurso

En el siguiente `list-tags-for-resource` ejemplo, se muestran las claves y los valores de las etiquetas asociados al MediaConnect recurso especificado.

```
aws mediacconnect list-tags-for-resource \  
  --resource-arn arn:aws:mediacconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BasketballGame
```

Salida:

```
{  
  "Tags": {  
    "region": "west",  
    "stage": "prod"  
  }  
}
```

Para obtener más información, consulte [ListTagsForResource](#), [TagResource](#), [UntagResource](#) en la MediaConnect APIReferencia AWS elemental.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## remove-flow-output

En el siguiente ejemplo de código, se muestra cómo usar `remove-flow-output`.

## AWS CLI

Para eliminar una salida de un flujo

El siguiente `remove-flow-output` ejemplo elimina una salida del flujo especificado.

```
aws mediacconnect remove-flow-output \  
  --flow-arn arn:aws:mediacconnect:us-  
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \  
  --output-arn arn:aws:mediacconnect:us-  
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC
```

Salida:

```
{
  "FlowArn": "arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame",
  "OutputArn": "arn:aws:mediacconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC"
}
```

Para obtener más información, consulte [Eliminar salidas de un flujo](#) en la Guía del MediaConnect usuario de AWS Elemental.

- Para API obtener más información, consulte [RemoveFlowOutput](#) la Referencia de AWS CLI comandos.

## revoke-flow-entitlement

En el siguiente ejemplo de código, se muestra cómo usar `revoke-flow-entitlement`.

AWS CLI

Para revocar un derecho

En el siguiente `revoke-flow-entitlement` ejemplo, se revoca un derecho del flujo especificado.

```
aws mediacconnect revoke-flow-entitlement \
  --flow-arn arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
  --entitlement-arn arn:aws:mediacconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement
```

Salida:

```
{
  "FlowArn": "arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame",
  "EntitlementArn": "arn:aws:mediacconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement"
}
```

Para obtener más información, consulte [Revocación de un derecho en la Guía](#) del usuario de Elemental.AWS MediaConnect

- Para API obtener más información, consulte la Referencia de [RevokeFlowEntitlement](#) comandos AWS CLI .

## start-flow

En el siguiente ejemplo de código, se muestra cómo usar `start-flow`.

### AWS CLI

Para iniciar un flujo

El siguiente `start-flow` ejemplo inicia el flujo especificado.

```
aws mediaconnect start-flow \  
  --flow-arn arn:aws:mediaconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow
```

Este comando no genera ninguna salida. Salida:

```
{  
  "FlowArn": "arn:aws:mediaconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow",  
  "Status": "STARTING"  
}
```

Para obtener más información, consulte Cómo [iniciar un flujo](#) en la Guía del MediaConnect usuario de AWS Elemental.

- Para API obtener más información, consulte [StartFlow](#) la Referencia de AWS CLI comandos.

## stop-flow

En el siguiente ejemplo de código, se muestra cómo usar `stop-flow`.

### AWS CLI

Para detener un flujo

El siguiente `stop-flow` ejemplo detiene el flujo especificado.

```
aws mediacconnect stop-flow \  
  --flow-arn arn:aws:mediacconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow
```

Salida:

```
{  
  "Status": "STOPPING",  
  "FlowArn": "arn:aws:mediacconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow"  
}
```

Para obtener más información, consulte [Detener un flujo](#) en la Guía del MediaConnect usuario de AWS Elemental.

- Para API obtener más información, consulte [StopFlow](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

AWS CLI

Para añadir etiquetas a un MediaConnect recurso

En el siguiente tag-resource ejemplo, se agrega una etiqueta con un nombre y un valor clave al MediaConnect recurso especificado.

```
aws mediacconnect tag-resource \  
  --resource-arn arn:aws:mediacconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BasketballGame  
  --tags region=west
```

Este comando no genera ninguna salida.

Para obtener más información [ListTagsForResource](#), consulte [TagResource](#), [UntagResource](#) en la MediaConnect API Referencia AWS elemental.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.



## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar etiquetas de un MediaConnect recurso

En el siguiente `untag-resource` ejemplo, se elimina de un MediaConnect recurso la etiqueta con el nombre de clave especificado y su valor asociado.

```
aws mediacconnect untag-resource \  
  --resource-arn arn:aws:mediacconnect:us-  
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BasketballGame \  
  --tag-keys region
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [ListTagsForResource](#), [TagResource](#), [UntagResource](#) en la MediaConnect API Referencia AWS elemental.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-flow-entitlement

En el siguiente ejemplo de código, se muestra cómo usar `update-flow-entitlement`.

### AWS CLI

Para actualizar un derecho

En el siguiente `update-flow-entitlement` ejemplo, se actualiza el derecho especificado con una nueva descripción y un suscriptor nuevos.

```
aws mediacconnect update-flow-entitlement \  
  --flow-arn arn:aws:mediacconnect:us-  
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \  
  --entitlement-arn arn:aws:mediacconnect:us-  
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement \  
  --description 'For AnyCompany Affiliate' \  
  \
```

```
--subscribers 777788889999
```

Salida:

```
{
  "FlowArn": "arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame",
  "Entitlement": {
    "Name": "AnyCompany_Entitlement",
    "Description": "For AnyCompany Affiliate",
    "EntitlementArn": "arn:aws:mediacconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement",
    "Encryption": {
      "KeyType": "static-key",
      "Algorithm": "aes128",
      "RoleArn": "arn:aws:iam::111122223333:role/MediaConnect-ASM",
      "SecretArn": "arn:aws:secretsmanager:us-
west-2:111122223333:secret:mySecret1"
    },
    "Subscribers": [
      "777788889999"
    ]
  }
}
```

Para obtener más información, consulte [Actualización de un derecho](#) en la Guía del usuario de Elemental.AWS MediaConnect

- Para API obtener más información, consulte la Referencia [UpdateFlowEntitlement](#) de AWS CLI comandos.

## update-flow-output

En el siguiente ejemplo de código, se muestra cómo usar update-flow-output.

AWS CLI

Para actualizar una salida de un flujo

En el siguiente update-flow-output ejemplo, se actualiza una salida en el flujo especificado.

```
aws mediacconnect update-flow-output \
```

```

--flow-arn arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
--output-arn arn:aws:mediacconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC \
--port 3331

```

Salida:

```

{
  "FlowArn": "arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame",
  "Output": {
    "Name": "NYC",
    "Port": 3331,
    "Description": "NYC stream",
    "Transport": {
      "Protocol": "rtp-fec",
      "SmoothingLatency": 100
    },
    "OutputArn": "arn:aws:mediacconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC",
    "Destination": "192.0.2.12"
  }
}

```

Para obtener más información, consulte [Actualización de las salidas de un flujo](#) en la Guía del MediaConnect usuario de AWS Elemental.

- Para API obtener más información, consulte [UpdateFlowOutput](#) la Referencia de AWS CLI comandos.

## update-flow-source

En el siguiente ejemplo de código, se muestra cómo usar `update-flow-source`.

AWS CLI

Para actualizar la fuente de un flujo existente

El siguiente `update-flow-source` ejemplo actualiza la fuente de un flujo existente.

```
aws mediacconnect update-flow-source \
```

```

--flow-arn arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow \
--source-arn arn:aws:mediacconnect:us-
east-1:111122223333:source:3-4aBC56dEF78hiJ90-4de5fG6Hi78Jk:ShowSource \
--description 'Friday night show' \
--ingest-port 3344 \
--protocol rtp-fec \
--whitelist-cidr 10.24.34.0/23

```

Salida:

```

{
  "FlowArn": "arn:aws:mediacconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow",
  "Source": {
    "IngestIp": "34.210.136.56",
    "WhitelistCidr": "10.24.34.0/23",
    "Transport": {
      "Protocol": "rtp-fec"
    },
    "IngestPort": 3344,
    "Name": "ShowSource",
    "Description": "Friday night show",
    "SourceArn": "arn:aws:mediacconnect:us-
east-1:111122223333:source:3-4aBC56dEF78hiJ90-4de5fG6Hi78Jk:ShowSource"
  }
}

```

Para obtener más información, consulte [Actualización del origen de un flujo](#) en la Guía del MediaConnect usuario de AWS Elemental.

- Para API obtener más información, consulte [UpdateFlowSource](#) la Referencia de AWS CLI comandos.

## MediaConvert ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with MediaConvert.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **cancel-job**

En el siguiente ejemplo de código, se muestra cómo usar `cancel-job`.

#### AWS CLI

Para cancelar un trabajo que está en cola

En el siguiente `cancel-job` ejemplo, se cancela el trabajo con un ID. `1234567891234-abc123`  
No puede cancelar un trabajo que el servicio ha empezado a procesar.

```
aws mediaconvert cancel-job \  
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \  
  --region region-name-1 \  
  --id 1234567891234-abc123
```

Para obtener el punto de conexión específico de su cuenta, utilice `describe-endpoints` o envíe el comando sin el punto de conexión. El servicio devuelve un error y su punto de conexión.

Para obtener más información, consulte [Trabajar con MediaConvert trabajos de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [CancelJob](#) la Referencia de AWS CLI comandos.

### **create-job-template**

En el siguiente ejemplo de código, se muestra cómo usar `create-job-template`.

#### AWS CLI

Para crear una nueva plantilla de trabajo

En el siguiente `create-job-template` ejemplo, se crea una plantilla de trabajo con la configuración de transcodificación que se especifica en el archivo `job-template.json` que reside en el sistema.

```
aws mediaconvert create-job-template \  
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \  
  --region region-name-1 \  
  --name JobTemplate1 \  
  --cli-input-json file://~/job-template.json
```

Si crea el JSON archivo de plantilla de trabajo utilizando `get-job-template` y, a continuación, modificando el archivo, elimine el `JobTemplate` objeto, pero mantenga el objeto secundario de Configuración dentro de él. Además, asegúrese de eliminar los siguientes pares clave-valor: `LastUpdated`, `ArnType`, y `CreatedAt`. Puede especificar la categoría, la descripción, el nombre y la cola en el JSON archivo o en la línea de comandos.

Para obtener el punto de conexión específico de su cuenta, utilice `describe-endpoints` o envíe el comando sin el punto de conexión. El servicio devuelve un error y su punto de conexión.

Si la solicitud se realiza correctamente, el servicio devuelve la JSON especificación de la plantilla de trabajo que ha creado.

Para obtener más información, consulte [Trabajar con plantillas de MediaConvert tareas de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [CreateJobTemplate](#) la Referencia de AWS CLI comandos.

## create-job

En el siguiente ejemplo de código, se muestra cómo usar `create-job`.

### AWS CLI

#### Creación de un trabajo

En el siguiente ejemplo de `create-job` se crea un trabajo de transcodificación con la configuración especificada en un archivo `job.json` que reside en el sistema desde el que se envía el comando. Esta especificación de JSON trabajo puede especificar cada configuración de forma individual, hacer referencia a una plantilla de trabajo o hacer referencia a ajustes preestablecidos de salida.

```
aws mediaconvert create-job \  
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \  
  --region region-name-1 \  
  --cli-input-json file://~/job.json
```

Puede usar la MediaConvert consola AWS Elemental para generar la especificación del JSON trabajo seleccionando la configuración del trabajo y, a continuación, seleccionando Mostrar trabajo JSON en la parte inferior de la sección Trabajo.

Para obtener el punto de conexión específico de su cuenta, utilice `describe-endpoints` o envíe el comando sin el punto de conexión. El servicio devuelve un error y su punto de conexión.

Si la solicitud se ha realizado correctamente, el servicio devuelve la especificación del JSON puesto que enviaste con la solicitud.

Para obtener más información, consulte [Trabajar con MediaConvert trabajos de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [CreateJob](#) la Referencia de AWS CLI comandos.

## create-preset

En el siguiente ejemplo de código, se muestra cómo usar `create-preset`.

### AWS CLI

Para crear un ajuste preestablecido de salida personalizado

En el siguiente `create-preset` ejemplo, se crea un ajuste preestablecido de salida personalizado en función de los ajustes de salida especificados en el archivo `preset.json`. Puede especificar la categoría, la descripción y el nombre en el JSON archivo o en la línea de comandos.

```
aws mediaconvert create-preset \  
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \  
  --region region-name-1 \  
  --cli-input-json file://~/preset.json
```

Si crea el JSON archivo preestablecido utilizando `get-preset` y, a continuación, modificando el archivo de salida, asegúrese de eliminar los siguientes pares clave-valor: `LastUpdated`, `ArnType`, y `CreatedAt`

Para obtener el punto de conexión específico de su cuenta, utilice `describe-endpoints` o envíe el comando sin el punto de conexión. El servicio devuelve un error y su punto de conexión.

Para obtener más información, consulte [Uso de ajustes preestablecidos de MediaConvert salida de AWS Elemental](#) en la Guía del usuario de AWS Elemental MediaConvert .

- Para API obtener más información, consulte [CreatePreset](#) la Referencia de AWS CLI comandos.

## create-queue

En el siguiente ejemplo de código, se muestra cómo usar `create-queue`.

### AWS CLI

Para crear una cola personalizada

En el siguiente `create-queue` ejemplo, se crea una cola de transcodificación personalizada.

```
aws mediaconvert create-queue \  
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \  
  --region region-name-1 \  
  --name Queue1 \  
  --description "Keep this queue empty unless job is urgent."
```

Para obtener el punto de conexión específico de su cuenta, utilice `describe-endpoints` o envíe el comando sin el punto de conexión. El servicio devuelve un error y su punto de conexión.

Salida:

```
{  
  "Queue": {  
    "Status": "ACTIVE",  
    "Name": "Queue1",  
    "LastUpdated": 1518034928,  
    "Arn": "arn:aws:mediaconvert:region-name-1:012345678998:queues/Queue1",  
    "Type": "CUSTOM",  
    "CreatedAt": 1518034928,  
    "Description": "Keep this queue empty unless job is urgent."  
  }  
}
```



Para obtener más información, consulte [Cómo trabajar con MediaConvert colas AWS elementales](#) en la Guía del usuario de AWS Elemental MediaConvert .

- Para API obtener más información, consulte [CreateQueue](#) la Referencia de AWS CLI comandos.

## delete-job-template

En el siguiente ejemplo de código, se muestra cómo usar delete-job-template.

### AWS CLI

Para eliminar una plantilla de trabajo

En el siguiente delete-job-template ejemplo, se elimina la plantilla de trabajo personalizada especificada.

```
aws mediaconvert delete-job-template \  
  --name "DASH Streaming" \  
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Este comando no genera ninguna salida. Ejecute `aws mediaconvert list-job-templates` para confirmar que la plantilla se ha eliminado.

Para obtener más información, consulte [Trabajar con plantillas de MediaConvert tareas de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [DeleteJobTemplate](#) la Referencia de AWS CLI comandos.

## delete-preset

En el siguiente ejemplo de código, se muestra cómo usar delete-preset.

### AWS CLI

Para eliminar una cola personalizada bajo demanda

En el siguiente delete-preset ejemplo, se elimina el ajuste preestablecido personalizado especificado.

```
aws mediaconvert delete-preset \  
  --name "DASH Streaming" \  
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

```
--name SimpleMP4 \  
--endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Este comando no genera ninguna salida. Ejecute `aws mediaconvert list-presets` para confirmar que se ha eliminado el ajuste preestablecido.

Para obtener más información, consulte Cómo [trabajar con ajustes preestablecidos de MediaConvert salida de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [DeletePreset](#) la Referencia de AWS CLI comandos.

## delete-queue

En el siguiente ejemplo de código, se muestra cómo usar `delete-queue`.

### AWS CLI

Para eliminar una cola personalizada bajo demanda

En el siguiente `delete-queue` ejemplo, se elimina la cola bajo demanda personalizada especificada.

No puedes eliminar la cola predeterminada. No se puede eliminar una cola reservada que tenga un plan de precios activo o que contenga trabajos sin procesar.

```
aws mediaconvert delete-queue \  
--name Customer1 \  
--endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Este comando no genera ninguna salida. Ejecuta `aws mediaconvert list-queues` para confirmar que se ha eliminado la cola.

Para obtener más información, consulte Cómo [trabajar con MediaConvert colas de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [DeleteQueue](#) la Referencia de AWS CLI comandos.

## describe-endpoints

En el siguiente ejemplo de código, se muestra cómo usar `describe-endpoints`.

## AWS CLI

Para obtener el punto final específico de su cuenta

En el siguiente `describe-endpoints` ejemplo, se recupera el punto de conexión que necesitas para enviar cualquier otra solicitud al servicio.

```
aws mediaconvert describe-endpoints
```

Salida:

```
{
  "Endpoints": [
    {
      "Url": "https://abcd1234.mediaconvert.region-name-1.amazonaws.com"
    }
  ]
}
```

Para obtener más información, consulte [Introducción al MediaConvert uso de API in the AWS Elemental MediaConvert API Reference](#).

- Para API obtener más información, consulte [DescribeEndpoints](#) la Referencia de AWS CLI comandos.

## get-job-template

En el siguiente ejemplo de código, se muestra cómo usar `get-job-template`.

## AWS CLI

Para obtener detalles de una plantilla de trabajo

En el siguiente `get-job-template` ejemplo, se muestra la JSON definición de la plantilla de trabajo personalizada especificada.

```
aws mediaconvert get-job-template \
  --name "DASH Streaming" \
  --endpoint-url https://abcd1234.mediaconvert.us-east-1.amazonaws.com
```

Salida:

```
{
  "JobTemplate": {
    "StatusUpdateInterval": "SECONDS_60",
    "LastUpdated": 1568652998,
    "Description": "Create a DASH streaming ABR stack",
    "CreatedAt": 1568652998,
    "Priority": 0,
    "Name": "DASH Streaming",
    "Settings": {
      ...<truncatedforbrevity>...
    },
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:jobTemplates/DASH
Streaming",
    "Type": "CUSTOM"
  }
}
```

Para obtener más información, consulte [Trabajar con plantillas de MediaConvert tareas de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [GetJobTemplate](#) la Referencia de AWS CLI comandos.

## get-job

En el siguiente ejemplo de código, se muestra cómo usar `get-job`.

### AWS CLI

Obtención de los detalles de un trabajo en particular

En el siguiente ejemplo, se solicita la información del trabajo con el ID `1234567890987-1ab2c3`, que en este ejemplo finalizó con un error.

```
aws mediaconvert get-job \
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \
  --region region-name-1 \
  --id 1234567890987-1ab2c3
```

Para obtener el punto de conexión específico de su cuenta, utilice `describe-endpoints` o envíe el comando sin el punto de conexión. El servicio devuelve un error y su punto de conexión.

Si la solicitud se realiza correctamente, el servicio devuelve un JSON archivo con la información del trabajo, incluida la configuración del trabajo, los errores devueltos y otros datos del trabajo, de la siguiente manera:

```
{
  "Job": {
    "Status": "ERROR",
    "Queue": "arn:aws:mediaconvert:region-name-1:012345678998:queues/Queue1",
    "Settings": {
      ...<truncated for brevity>...
    },
    "ErrorMessage": "Unable to open input file [s3://my-input-bucket/file-name.mp4]: [Failed probe/open: [Failed to read data: AssumeRole failed]]",
    "ErrorCode": 1434,
    "Role": "arn:aws:iam::012345678998:role/MediaConvertServiceRole",
    "Arn": "arn:aws:mediaconvert:us-west-1:012345678998:jobs/1234567890987-1ab2c3",
    "UserMetadata": {},
    "Timing": {
      "FinishTime": 1517442131,
      "SubmitTime": 1517442103,
      "StartTime": 1517442104
    },
    "Id": "1234567890987-1ab2c3",
    "CreatedAt": 1517442103
  }
}
```

Para obtener más información, consulte [Trabajar con MediaConvert trabajos de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [GetJob](#) la Referencia de AWS CLI comandos.

## get-preset

En el siguiente ejemplo de código, se muestra cómo usar `get-preset`.

### AWS CLI

Para obtener detalles de un ajuste preestablecido en particular

En el siguiente `get-preset` ejemplo, se solicita la JSON definición del ajuste preestablecido personalizado especificado.

```
aws mediaconvert get-preset \  
  --name SimpleMP4 \  
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Salida:

```
{  
  "Preset": {  
    "Description": "Creates basic MP4 file. No filtering or preprocessing.",  
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:presets/SimpleMP4",  
    "LastUpdated": 1568843141,  
    "Name": "SimpleMP4",  
    "Settings": {  
      "ContainerSettings": {  
        "Mp4Settings": {  
          "FreeSpaceBox": "EXCLUDE",  
          "CslgAtom": "INCLUDE",  
          "MoovPlacement": "PROGRESSIVE_DOWNLOAD"  
        },  
        "Container": "MP4"  
      },  
      "AudioDescriptions": [  
        {  
          "LanguageCodeControl": "FOLLOW_INPUT",  
          "AudioTypeControl": "FOLLOW_INPUT",  
          "CodecSettings": {  
            "AacSettings": {  
              "RawFormat": "NONE",  
              "CodecProfile": "LC",  
              "AudioDescriptionBroadcasterMix": "NORMAL",  
              "SampleRate": 48000,  
              "Bitrate": 96000,  
              "RateControlMode": "CBR",  
              "Specification": "MPEG4",  
              "CodingMode": "CODING_MODE_2_0"  
            },  
            "Codec": "AAC"  
          }  
        }  
      ],  
      "VideoDescription": {  
        "RespondToAfd": "NONE",  
        "TimecodeInsertion": "DISABLED",
```

```
"Sharpness": 50,
"ColorMetadata": "INSERT",
"CodecSettings": {
  "H264Settings": {
    "FramerateControl": "INITIALIZE_FROM_SOURCE",
    "SpatialAdaptiveQuantization": "ENABLED",
    "Softness": 0,
    "Telecine": "NONE",
    "CodecLevel": "AUTO",
    "QualityTuningLevel": "SINGLE_PASS",
    "UnregisteredSeiTimecode": "DISABLED",
    "Slices": 1,
    "Syntax": "DEFAULT",
    "GopClosedCadence": 1,
    "AdaptiveQuantization": "HIGH",
    "EntropyEncoding": "CABAC",
    "InterlaceMode": "PROGRESSIVE",
    "ParControl": "INITIALIZE_FROM_SOURCE",
    "NumberBFramesBetweenReferenceFrames": 2,
    "GopSizeUnits": "FRAMES",
    "RepeatPps": "DISABLED",
    "CodecProfile": "MAIN",
    "FieldEncoding": "PAFF",
    "GopSize": 90.0,
    "SlowPal": "DISABLED",
    "SceneChangeDetect": "ENABLED",
    "GopBReference": "DISABLED",
    "RateControlMode": "CBR",
    "FramerateConversionAlgorithm": "DUPLICATE_DROP",
    "FlickerAdaptiveQuantization": "DISABLED",
    "DynamicSubGop": "STATIC",
    "MinIInterval": 0,
    "TemporalAdaptiveQuantization": "ENABLED",
    "Bitrate": 400000,
    "NumberReferenceFrames": 3
  },
  "Codec": "H_264"
},
"AfdSignaling": "NONE",
"AntiAlias": "ENABLED",
"ScalingBehavior": "DEFAULT",
"DropFrameTimecode": "ENABLED"
},
},
```

```
    "Type": "CUSTOM",
    "CreatedAt": 1568841521
  }
}
```

Para obtener más información, consulte [Uso de ajustes preestablecidos de MediaConvert salida de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [GetPreset](#) la Referencia de AWS CLI comandos.

## get-queue

En el siguiente ejemplo de código, se muestra cómo usar `get-queue`.

### AWS CLI

Para obtener detalles de una cola

En el siguiente `get-queue` ejemplo, se recuperan los detalles de la cola personalizada especificada.

```
aws mediaconvert get-queue \
  --name Customer1 \
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Salida:

```
{
  "Queue": {
    "LastUpdated": 1526428502,
    "Type": "CUSTOM",
    "SubmittedJobsCount": 0,
    "Status": "ACTIVE",
    "PricingPlan": "ON_DEMAND",
    "CreatedAt": 1526428502,
    "ProgressingJobsCount": 0,
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/Customer1",
    "Name": "Customer1"
  }
}
```



Para obtener más información, consulte [Trabajar con MediaConvert colas AWS elementales](#) en la Guía del usuario de AWS Elemental MediaConvert .

- Para API obtener más información, consulte [GetQueue](#) la Referencia de AWS CLI comandos.

## list-job-templates

En el siguiente ejemplo de código, se muestra cómo usar `list-job-templates`.

### AWS CLI

Ejemplo 1: Para enumerar sus plantillas de trabajo personalizadas

El siguiente `list-job-templates` ejemplo muestra todas las plantillas de trabajos personalizadas de la región actual. Para ver una lista de las plantillas de trabajos del sistema, consulte el siguiente ejemplo.

```
aws mediaconvert list-job-templates \  
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Salida:

```
{  
  "JobTemplates": [  
    {  
      "Description": "Create a DASH streaming ABR stack",  
      "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:jobTemplates/DASH  
Streaming",  
      "Name": "DASH Streaming",  
      "LastUpdated": 1568653007,  
      "Priority": 0,  
      "Settings": {  
        ...<truncatedforbrevity>...  
      },  
      "Type": "CUSTOM",  
      "StatusUpdateInterval": "SECONDS_60",  
      "CreatedAt": 1568653007  
    },  
    {  
      "Description": "Create a high-res file",  
      "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:jobTemplates/File",  
      "Name": "File",  
      "LastUpdated": 1568653007,  
    }  
  ]  
}
```

```

    "Priority": 0,
    "Settings": {
      ...<truncatedforbrevity>...
    },
    "Type": "CUSTOM",
    "StatusUpdateInterval": "SECONDS_60",
    "CreatedAt": 1568653023
  }
]
}

```

Ejemplo 2: Para enumerar las plantillas de trabajos MediaConvert del sistema

En el siguiente `list-job-templates` ejemplo, se enumeran todas las plantillas de trabajos del sistema.

```

aws mediaconvert list-job-templates \
  --endpoint-url https://abcd1234.mediaconvert.us-east-1.amazonaws.com \
  --list-by SYSTEM

```

Salida:

```

{
  "JobTemplates": [
    {
      "CreatedAt": 1568321779,
      "Arn": "arn:aws:mediaconvert:us-east-1:123456789012:jobTemplates/System-
Generic_Mp4_Hev1_Avc_Aac_Sdr_Qvbr",
      "Name": "System-Generic_Mp4_Hev1_Avc_Aac_Sdr_Qvbr",
      "Description": "GENERIC, MP4, AVC + HEV1(HEVC,SDR), AAC, SDR, QVBR",
      "Category": "GENERIC",
      "Settings": {
        "AdAvailOffset": 0,
        "OutputGroups": [
          {
            "Outputs": [
              {
                "Extension": "mp4",
                "Preset": "System-
Generic_Hd_Mp4_Avc_Aac_16x9_Sdr_1280x720p_30Hz_5Mbps_Qvbr_Vq9",
                "NameModifier":
                "_Generic_Hd_Mp4_Avc_Aac_16x9_Sdr_1280x720p_30Hz_5000Kbps_Qvbr_Vq9"
              },
            ],
          },
        ],
      },
    },
  ],
}

```

```

        {
            "Extension": "mp4",
            "Preset": "System-
Generic_Hd_Mp4_Avc_Aac_16x9_Sdr_1920x1080p_30Hz_10Mbps_Qvbr_Vq9",
            "NameModifier":
            "_Generic_Hd_Mp4_Avc_Aac_16x9_Sdr_1920x1080p_30Hz_10000Kbps_Qvbr_Vq9"
        },
        {
            "Extension": "mp4",
            "Preset": "System-
Generic_Sd_Mp4_Avc_Aac_16x9_Sdr_640x360p_30Hz_0.8Mbps_Qvbr_Vq7",
            "NameModifier":
            "_Generic_Sd_Mp4_Avc_Aac_16x9_Sdr_640x360p_30Hz_800Kbps_Qvbr_Vq7"
        },
        {
            "Extension": "mp4",
            "Preset": "System-
Generic_Hd_Mp4_Hev1_Aac_16x9_Sdr_1280x720p_30Hz_4Mbps_Qvbr_Vq9",
            "NameModifier":
            "_Generic_Hd_Mp4_Hev1_Aac_16x9_Sdr_1280x720p_30Hz_4000Kbps_Qvbr_Vq9"
        },
        {
            "Extension": "mp4",
            "Preset": "System-
Generic_Hd_Mp4_Hev1_Aac_16x9_Sdr_1920x1080p_30Hz_8Mbps_Qvbr_Vq9",
            "NameModifier":
            "_Generic_Hd_Mp4_Hev1_Aac_16x9_Sdr_1920x1080p_30Hz_8000Kbps_Qvbr_Vq9"
        },
        {
            "Extension": "mp4",
            "Preset": "System-
Generic_Uhd_Mp4_Hev1_Aac_16x9_Sdr_3840x2160p_30Hz_12Mbps_Qvbr_Vq9",
            "NameModifier":
            "_Generic_Uhd_Mp4_Hev1_Aac_16x9_Sdr_3840x2160p_30Hz_12000Kbps_Qvbr_Vq9"
        }
    ],
    "OutputGroupSettings": {
        "FileGroupSettings": {

        },
        "Type": "FILE_GROUP_SETTINGS"
    },
    "Name": "File Group"
}

```

```
    ]
    },
    "Type": "SYSTEM",
    "LastUpdated": 1568321779
  },
  ...<truncatedforbrevity>...
]
}
```

Para obtener más información, consulte [Trabajar con plantillas de MediaConvert tareas de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [ListJobTemplates](#) la Referencia de AWS CLI comandos.

## list-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-jobs`.

AWS CLI

Obtención de los detalles de todos los trabajos de una región

En el siguiente ejemplo se solicita la información de todos los trabajos de la región especificada.

```
aws mediaconvert list-jobs \
  --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \
  --region region-name-1
```

Para obtener el punto de conexión específico de su cuenta, utilice `describe-endpoints` o envíe el comando sin el punto de conexión. El servicio devuelve un error y su punto de conexión.

Para obtener más información, consulte [Trabajar con MediaConvert tareas AWS elementales](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [ListJobs](#) la Referencia de AWS CLI comandos.

## list-presets

En el siguiente ejemplo de código, se muestra cómo usar `list-presets`.

## AWS CLI

Ejemplo 1: Para enumerar sus ajustes preestablecidos de salida personalizados

En el siguiente `list-presets` ejemplo, se enumeran los ajustes preestablecidos de salida personalizados. Para ver una lista de los ajustes preestablecidos del sistema, consulte el siguiente ejemplo.

```
aws mediaconvert list-presets \  
--endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Salida:

```
{  
  "Presets": [  
    {  
      "Name": "SimpleMP4",  
      "CreatedAt": 1568841521,  
      "Settings": {  
        .....  
      },  
      "Arn": "arn:aws:mediaconvert:us-east-1:003235472598:presets/SimpleMP4",  
      "Type": "CUSTOM",  
      "LastUpdated": 1568843141,  
      "Description": "Creates basic MP4 file. No filtering or preprocessing."  
    },  
    {  
      "Name": "SimpleTS",  
      "CreatedAt": 1568843113,  
      "Settings": {  
        ... truncated for brevity ...  
      },  
      "Arn": "arn:aws:mediaconvert:us-east-1:003235472598:presets/SimpleTS",  
      "Type": "CUSTOM",  
      "LastUpdated": 1568843113,  
      "Description": "Create a basic transport stream."  
    }  
  ]  
}
```

Ejemplo 2: Para enumerar los ajustes preestablecidos de salida del sistema

En el siguiente `list-presets` ejemplo, se enumeran los ajustes preestablecidos MediaConvert del sistema disponibles. Para ver una lista de los ajustes preestablecidos personalizados, consulte el ejemplo anterior.

```
aws mediaconvert list-presets \  
  --list-by SYSTEM \  
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Salida:

```
{  
  "Presets": [  
    {  
      "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:presets/System-Avc_16x9_1080p_29_97fps_8500kbps",  
      "Name": "System-Avc_16x9_1080p_29_97fps_8500kbps",  
      "CreatedAt": 1568321789,  
      "Description": "Wifi, 1920x1080, 16:9, 29.97fps, 8500kbps",  
      "LastUpdated": 1568321789,  
      "Type": "SYSTEM",  
      "Category": "HLS",  
      "Settings": {  
        ...<output settings removed for brevity>...  
      }  
    },  
    ...<list of presets shortened for brevity>...  
    {  
      "Arn": "arn:aws:mediaconvert:us-east-1:123456789012:presets/System-Xdcam_HD_1080i_29_97fps_35mpbs",  
      "Name": "System-Xdcam_HD_1080i_29_97fps_35mpbs",  
      "CreatedAt": 1568321790,  
      "Description": "XDCAM MPEG HD, 1920x1080i, 29.97fps, 35mbps",  
      "LastUpdated": 1568321790,  
      "Type": "SYSTEM",  
      "Category": "MXF",  
      "Settings": {  
        ...<output settings removed for brevity>...  
      }  
    }  
  ]  
}
```

```
}
```

Para obtener más información, consulte [Cómo trabajar con ajustes preestablecidos de MediaConvert salida de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [ListPresets](#) la Referencia de AWS CLI comandos.

## list-queues

En el siguiente ejemplo de código, se muestra cómo usar `list-queues`.

### AWS CLI

Para enumerar sus colas

En el siguiente `list-queues` ejemplo, se muestran todas las MediaConvert colas.

```
aws mediaconvert list-queues \  
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Salida:

```
{  
  "Queues": [  
    {  
      "PricingPlan": "ON_DEMAND",  
      "Type": "SYSTEM",  
      "Status": "ACTIVE",  
      "CreatedAt": 1503451595,  
      "Name": "Default",  
      "SubmittedJobsCount": 0,  
      "ProgressingJobsCount": 0,  
      "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/Default",  
      "LastUpdated": 1534549158  
    },  
    {  
      "PricingPlan": "ON_DEMAND",  
      "Type": "CUSTOM",  
      "Status": "ACTIVE",  
      "CreatedAt": 1537460025,  
      "Name": "Custom",  
      "SubmittedJobsCount": 0,  
      "ProgressingJobsCount": 0,  
      "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/Custom",  
      "LastUpdated": 1537460025  
    }  
  ]  
}
```

```

    "Name": "Customer1",
    "SubmittedJobsCount": 0,
    "Description": "Jobs we run for our cusotmer.",
    "ProgressingJobsCount": 0,
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/Customer1",
    "LastUpdated": 1537460025
  },
  {
    "ProgressingJobsCount": 0,
    "Status": "ACTIVE",
    "Name": "transcode-library",
    "SubmittedJobsCount": 0,
    "LastUpdated": 1564066204,
    "ReservationPlan": {
      "Status": "ACTIVE",
      "ReservedSlots": 1,
      "PurchasedAt": 1564066203,
      "Commitment": "ONE_YEAR",
      "ExpiresAt": 1595688603,
      "RenewalType": "EXPIRE"
    },
    "PricingPlan": "RESERVED",
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/transcode-
library",
    "Type": "CUSTOM",
    "CreatedAt": 1564066204
  }
]
}

```

Para obtener más información, consulte Cómo [trabajar con MediaConvert colas AWS elementales](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [ListQueues](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de una MediaConvert cola, plantilla de trabajo o ajuste preestablecido de salida



En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas del ajuste preestablecido de salida especificado.

```
aws mediaconvert list-tags-for-resource \  
  --arn arn:aws:mediaconvert:us-west-2:123456789012:presets/SimpleMP4 \  
  --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Salida:

```
{  
  "ResourceTags": {  
    "Tags": {  
      "customer": "zippyVideo"  
    },  
    "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:presets/SimpleMP4"  
  }  
}
```

Para obtener más información, consulte [Etiquetado de MediaConvert colas AWS elementales, plantillas de trabajos y ajustes preestablecidos de salida](#) en la Guía del usuario de AWS Elemental MediaConvert .

- Para API obtener más información, consulte la Referencia de [ListTagsForResource](#) comandos AWS CLI .

## update-job-template

En el siguiente ejemplo de código, se muestra cómo usar `update-job-template`.

AWS CLI

Para cambiar una plantilla de trabajo

El siguiente `update-job-template` ejemplo reemplaza la JSON definición de la plantilla de trabajo personalizada especificada por la JSON definición del archivo proporcionado.

```
aws mediaconvert update-job-template --name File1 https://abcd1234.mediaconvert.us-  
west-2.amazonaws.com --endpoint-url -- file: //~/ .json cli-input-json job-template-update
```

Contenidos de `job-template-update.json`:

```
{
```

```
"Description": "A simple job template that generates a single file output.",
"Queue": "arn:aws:mediaconvert:us-east-1:012345678998:queues/Default",
"Name": "SimpleFile",
"Settings": {
  "OutputGroups": [
    {
      "Name": "File Group",
      "Outputs": [
        {
          "ContainerSettings": {
            "Container": "MP4",
            "Mp4Settings": {
              "CslgAtom": "INCLUDE",
              "FreeSpaceBox": "EXCLUDE",
              "MoovPlacement": "PROGRESSIVE_DOWNLOAD"
            }
          }
        }
      ],
      "VideoDescription": {
        "ScalingBehavior": "DEFAULT",
        "TimecodeInsertion": "DISABLED",
        "AntiAlias": "ENABLED",
        "Sharpness": 50,
        "CodecSettings": {
          "Codec": "H_264",
          "H264Settings": {
            "InterlaceMode": "PROGRESSIVE",
            "NumberReferenceFrames": 3,
            "Syntax": "DEFAULT",
            "Softness": 0,
            "GopClosedCadence": 1,
            "GopSize": 90,
            "Slices": 1,
            "GopBReference": "DISABLED",
            "SlowPal": "DISABLED",
            "SpatialAdaptiveQuantization": "ENABLED",
            "TemporalAdaptiveQuantization": "ENABLED",
            "FlickerAdaptiveQuantization": "DISABLED",
            "EntropyEncoding": "CABAC",
            "Bitrate": 400000,
            "FramerateControl": "INITIALIZE_FROM_SOURCE",
            "RateControlMode": "CBR",
            "CodecProfile": "MAIN",
            "Telecine": "NONE",
            "MinIInterval": 0,

```

```
        "AdaptiveQuantization": "HIGH",
        "CodecLevel": "AUTO",
        "FieldEncoding": "PAFF",
        "SceneChangeDetect": "ENABLED",
        "QualityTuningLevel": "SINGLE_PASS",
        "FramerateConversionAlgorithm": "DUPLICATE_DROP",
        "UnregisteredSeiTimecode": "DISABLED",
        "GopSizeUnits": "FRAMES",
        "ParControl": "INITIALIZE_FROM_SOURCE",
        "NumberBFramesBetweenReferenceFrames": 2,
        "RepeatPps": "DISABLED",
        "DynamicSubGop": "STATIC"
    }
},
"AfdSignaling": "NONE",
"DropFrameTimecode": "ENABLED",
"RespondToAfd": "NONE",
"ColorMetadata": "INSERT"
},
"AudioDescriptions": [
    {
        "AudioTypeControl": "FOLLOW_INPUT",
        "CodecSettings": {
            "Codec": "AAC",
            "AacSettings": {
                "AudioDescriptionBroadcasterMix": "NORMAL",
                "Bitrate": 96000,
                "RateControlMode": "CBR",
                "CodecProfile": "LC",
                "CodingMode": "CODING_MODE_2_0",
                "RawFormat": "NONE",
                "SampleRate": 48000,
                "Specification": "MPEG4"
            }
        },
        "LanguageCodeControl": "FOLLOW_INPUT"
    }
]
}
},
"OutputGroupSettings": {
    "Type": "FILE_GROUP_SETTINGS",
    "FileGroupSettings": {}
}
```

```
    }
  ],
  "AdAvailOffset": 0
},
"StatusUpdateInterval": "SECONDS_60",
"Priority": 0
}
```

El sistema devuelve la JSON carga útil que usted envía con su solicitud, incluso si la solicitud produce un error. Por lo tanto, lo JSON devuelto no es necesariamente la nueva definición de la plantilla de trabajo.

Como la JSON carga útil puede ser larga, es posible que tengas que desplazarte hacia arriba para ver los mensajes de error.

Para obtener más información, consulte [Trabajar con plantillas de MediaConvert tareas de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [UpdateJobTemplate](#) la Referencia de AWS CLI comandos.

## update-preset

En el siguiente ejemplo de código, se muestra cómo usar update-preset.

### AWS CLI

Para cambiar un ajuste preestablecido

El siguiente update-preset ejemplo reemplaza la descripción del ajuste preestablecido especificado.

```
aws mediaconvert update-preset \
--name Customer1 \
--description "New description text."
--endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Este comando no genera ninguna salida. Salida:

```
{
```

```

    "Preset": {
      "Arn": "arn:aws:mediaconvert:us-east-1:003235472598:presets/SimpleMP4",
      "Settings": {
        ...<output settings removed for brevity>...
      },
      "Type": "CUSTOM",
      "LastUpdated": 1568938411,
      "Description": "New description text.",
      "Name": "SimpleMP4",
      "CreatedAt": 1568938240
    }
  }
}

```

Para obtener más información, consulte [Trabajo con ajustes preestablecidos de MediaConvert salida de AWS Elemental](#) en la Guía del MediaConvert usuario de AWS Elemental.

- Para API obtener más información, consulte [UpdatePreset](#) la Referencia de AWS CLI comandos.

## update-queue

En el siguiente ejemplo de código, se muestra cómo usar update-queue.

### AWS CLI

Para cambiar una cola

En el update-queue ejemplo siguiente se detiene la cola especificada cambiando su estado a PAUSED

```

aws mediaconvert update-queue \
--name Customer1 \
--status PAUSED
--endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com

```

Salida:

```

{
  "Queue": {
    "LastUpdated": 1568839845,
    "Status": "PAUSED",

```

```
    "ProgressingJobsCount": 0,  
    "CreatedAt": 1526428516,  
    "Arn": "arn:aws:mediaconvert:us-west-1:123456789012:queues/Customer1",  
    "Name": "Customer1",  
    "SubmittedJobsCount": 0,  
    "PricingPlan": "ON_DEMAND",  
    "Type": "CUSTOM"  
  }  
}
```

Para obtener más información, consulte [Trabajar con MediaConvert colas AWS elementales](#) en la Guía del usuario de AWS Elemental MediaConvert .

- Para API obtener más información, consulte [UpdateQueue](#) la Referencia de AWS CLI comandos.

## MediaLive ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with MediaLive.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-channel**

En el siguiente ejemplo de código, se muestra cómo usar `create-channel`.

AWS CLI

Para crear un canal

En el siguiente `create-channel` ejemplo, se crea un canal pasando un JSON archivo que contiene los parámetros que se desean especificar.

El canal de este ejemplo ingiere una HLS PULL entrada que se conecta a una fuente que contiene vídeo, audio y subtítulos incrustados. El canal crea un grupo HLS de salida con un servidor de Akamai como destino. El grupo de salidas contiene dos salidas: una para el vídeo y el AAC audio H.265 y otra para los VTT subtítulos web, solo en inglés.

El canal de este JSON ejemplo incluye los parámetros mínimos necesarios para un canal que utilice una HLS PULL entrada y que produzca un grupo de HLS salida con Akamai como destino. El JSON contiene las siguientes secciones principales:

`InputAttachments`, que especifica una fuente para el audio y una fuente para los subtítulos. No especifica un selector de vídeo, lo que significa que MediaLive extrae el primer vídeo que encuentra en la fuente. `Destinations`, que contiene las dos direcciones IP (URLs) del único grupo de salida de este canal. Estas direcciones requieren contraseñas. `EncoderSettings`, que contiene subsecciones. `AudioDescriptions`, que especifica que el canal contiene un recurso de salida de audio, que utiliza la fuente de `InputAttachments` origen y produce audio en AAC formato. `CaptionDescriptions`, que especifica que el canal contiene un recurso de salida de subtítulos, que utiliza la fuente de `InputAttachments` y produce subtítulos en formato web. `VTT VideoDescriptions`, que especifica que el canal contiene un recurso de salida de vídeo con la resolución especificada. `OutputGroups`, que especifica los grupos de salida. En este ejemplo, hay un grupo llamado `Akamai`. La conexión se realiza mediante `HLSPUT`. El grupo de salidas contiene dos salidas. Una salida es para el recurso de vídeo (con nombre `Video_high`) y el recurso de audio (con nombre `Audio_EN`). Una salida es para el recurso de subtítulos (denominado `WebVTT_EN`).

En este ejemplo, algunos de los parámetros no contienen ningún valor o contienen parámetros vacíos anidados. Por `OutputSettings` ejemplo, la `Video_and_audio` salida contiene varios parámetros anidados que terminan en un parámetro vacío `M3U8Settings`. Este parámetro debe estar incluido, pero puede omitir uno, varios o todos sus elementos secundarios, lo que significa que el elemento secundario tomará su valor predeterminado o será nulo.

Todos los parámetros que se apliquen a este canal de ejemplo pero que no estén especificados en este archivo tomarán el valor predeterminado, se establecerán en nulo o tomarán un valor único generado por MediaLive.

```
aws medialive create-channel \  
  --cli-input-json file://channel-in-hls-out-hls-akamai.json
```

## Contenidos de channel-in-hls-out-hls-akamai.json:

```
{
  "Name": "News_West",
  "RoleArn": "arn:aws:iam::111122223333:role/MediaLiveAccessRole",
  "InputAttachments": [
    {
      "InputAttachmentName": "local_news",
      "InputId": "1234567",
      "InputSettings": {
        "AudioSelectors": [
          {
            "Name": "English-Audio",
            "SelectorSettings": {
              "AudioLanguageSelection": {
                "LanguageCode": "EN"
              }
            }
          }
        ],
        "CaptionSelectors": [
          {
            "LanguageCode": "ENE",
            "Name": "English_embedded"
          }
        ]
      }
    }
  ],
  "Destinations": [
    {
      "Id": "akamai-server-west",
      "Settings": [
        {
          "PasswordParam": "/medialive/examplecorp1",
          "Url": "http://203.0.113.55/news/news_west",
          "Username": "examplecorp"
        },
        {
          "PasswordParam": "/medialive/examplecorp2",
          "Url": "http://203.0.113.82/news/news_west",
          "Username": "examplecorp"
        }
      ]
    }
  ]
}
```



```
    }
  ],
  "EncoderSettings": {
    "AudioDescriptions": [
      {
        "AudioSelectorName": "English-Audio",
        "CodecSettings": {
          "AacSettings": {}
        },
        "Name": "Audio_EN"
      }
    ],
    "CaptionDescriptions": [
      {
        "CaptionSelectorName": "English_embedded",
        "DestinationSettings": {
          "WebvttDestinationSettings": {}
        },
        "Name": "WebVTT_EN"
      }
    ],
    "VideoDescriptions": [
      {
        "Height": 720,
        "Name": "Video_high",
        "Width": 1280
      }
    ],
    "OutputGroups": [
      {
        "Name": "Akamai",
        "OutputGroupSettings": {
          "HlsGroupSettings": {
            "Destination": {
              "DestinationRefId": "akamai-server-west"
            },
            "HlsCdnSettings": {
              "HlsBasicPutSettings": {}
            }
          }
        },
        "Outputs": [
          {
            "AudioDescriptionNames": [
```

```

        "Audio_EN"
    ],
    "OutputName": "Video_and_audio",
    "OutputSettings": {
        "HlsOutputSettings": {
            "HlsSettings": {
                "StandardHlsSettings": {
                    "M3u8Settings": {}
                }
            },
            "NameModifier": "_1"
        }
    },
    "VideoDescriptionName": "Video_high"
},
{
    "CaptionDescriptionNames": [
        "WebVTT_EN"
    ],
    "OutputName": "Captions-WebVTT",
    "OutputSettings": {
        "HlsOutputSettings": {
            "HlsSettings": {
                "StandardHlsSettings": {
                    "M3u8Settings": {}
                }
            },
            "NameModifier": "_2"
        }
    }
}
]
}
},
"TimecodeConfig": {
    "Source": "EMBEDDED"
}
}
}

```

Salida:

La salida repite el contenido del JSON archivo, además de los valores siguientes. Todos los parámetros están ordenados alfabéticamente.

ARN para el canal. La última parte de ARN es el identificador único del canal.

EgressEndpoint se está en blanco en este canal de ejemplo porque solo se usa para PUSH entradas. Cuando se aplica, muestra las direcciones a las MediaLive que se envía el contenido.

OutputGroups, Outputs. Muestran todos los parámetros del grupo de salida y las salidas, incluidos los que no incluiste pero que son relevantes para este canal. Es posible que los parámetros estén vacíos (lo que podría indicar que el parámetro o la función están deshabilitados en esta configuración de canal) o pueden mostrar el valor predeterminado que se aplicará.

LogLevel está establecido en el valor predeterminado (DISABLED). Tag se está establecido en el valor predeterminado (null). PipelinesRunningCounty State muestran el estado actual del canal.

Para obtener más información, consulta [Cómo crear un canal desde cero](#) en la Guía del MediaLive usuario de AWS Elemental.

- Para API obtener más información, consulte [CreateChannel](#) en la Referencia de AWS CLI comandos.

## create-input

En el siguiente ejemplo de código, se muestra cómo usar create-input.

### AWS CLI

Para crear una entrada

El siguiente create-input ejemplo crea una HLS PULL entrada pasando un JSON archivo que contiene los parámetros que se aplican a este tipo de entrada. La entrada de este JSON ejemplo especifica dos fuentes (direcciones) para la entrada, a fin de admitir la redundancia en la ingesta. Estas direcciones requieren contraseñas.

```
aws medialive create-input \  
  --cli-input-json file://input-hls-pull-news.json
```

Contenidos de input-hls-pull-news.json:

```
{  
  "Name": "local_news",  
  "RequestId": "cli000059",
```

```

    "Sources": [
      {
        "Url": "https://203.0.113.13/newschannel/anytownusa.m3u8",
        "Username": "examplecorp",
        "PasswordParam": "/medialive/examplecorp1"
      },
      {
        "Url": "https://198.51.100.54/fillervideos/oceanwaves.mp4",
        "Username": "examplecorp",
        "PasswordParam": "examplecorp2"
      }
    ],
    "Type": "URL_PULL"
  }

```

Salida:

El resultado repite el contenido del JSON archivo, además de los valores siguientes. Todos los parámetros están ordenados alfabéticamente.

`Arn` para la entrada. La última parte de ARN es el identificador de entrada único. `Attached Channels`, que siempre está vacía cuando se trata de una entrada recién creada. `Destinations`, que está vacía en este ejemplo porque solo se usa con una PUSH entrada. `Id` para la entrada, igual que el identificador de ARN. `MediaConnectFlows`, que está vacía en este ejemplo porque solo se usa con una entrada de tipo MediaConnect. `SecurityGroups`, que está vacía en este ejemplo porque solo se usa con una PUSH entrada. `State` de esta entrada. `Tags`, que está vacía (el valor predeterminado para este parámetro).

Para obtener más información, consulte [Creación de una entrada](#) en la Guía del MediaLive usuario de AWS Elemental.

- Para API obtener más información, consulte [CreateInput](#) la Referencia de AWS CLI comandos.

## MediaPackage ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with MediaPackage.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **create-channel**

En el siguiente ejemplo de código, se muestra cómo usar `create-channel`.

#### AWS CLI

Para crear un canal

El siguiente `create-channel` comando crea un canal con el nombre `sportschannel` de la cuenta corriente.

```
aws mediapackage create-channel --id sportschannel
```

Salida:

```
{
  "Arn": "arn:aws:mediapackage:us-
west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0",
  "HlsIngest": {
    "IngestEndpoints": [
      {
        "Id": "6d345804ec3f46c9b454a91d4a80d0e0",
        "Password": "generatedwebdavpassword1",
        "Url": "https://f31c86aed53b815a.mediapackage.us-
west-2.amazonaws.com/in/
v2/6d345804ec3f46c9b454a91d4a80d0e0/6d345804ec3f46c9b454a91d4a80d0e0/channel",
        "Username": "generatedwebdavusername1"
      },
      {
        "Id": "2daa32878af24803b24183727211b8ff",
        "Password": "generatedwebdavpassword2",
        "Url": "https://6ebbe7e04c4b0afa.mediapackage.us-
west-2.amazonaws.com/in/
v2/6d345804ec3f46c9b454a91d4a80d0e0/2daa32878af24803b24183727211b8ff/channel",
```

```

        "Username": "generatedwebdavusername2"
      }
    ]
  },
  "Id": "sportschannel",
  "Tags": {
    "region": "west"
  }
}

```

Para obtener más información, consulte [Creación de un canal](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [CreateChannel](#) en la Referencia de AWS CLI comandos.

## create-origin-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `create-origin-endpoint`.

### AWS CLI

Para crear un punto final de origen

El siguiente `create-origin-endpoint` comando crea un punto final de origen denominado `cmf sports` con la configuración del paquete proporcionada en un JSON archivo y la configuración de punto final especificada.

```

aws mediapackage create-origin-endpoint \
  --channel-id sportschannel \
  --id cmf sports \
  --cmf-package file:///file/path/cmafpkg.json --description "cmf output of sports" \
  --id cmf_sports \
  --manifest-name sports_channel \
  --startover-window-seconds 300 \
  --tags region=west,media=sports \
  --time-delay-seconds 10

```

Salida:

```
{
```

```

    "Arn": "arn:aws:mediapackage:us-
west-2:111222333:origin_endpoints/1dc6718be36f4f34bb9cd86bc50925e6",
    "ChannelId": "sportschannel",
    "CmafPackage": {
      "HlsManifests": [
        {
          "AdMarkers": "PASSTHROUGH",
          "Id": "cmf_sports_endpoint",
          "IncludeIframeOnlyStream": true,
          "ManifestName": "index",
          "PlaylistType": "EVENT",
          "PlaylistWindowSeconds": 300,
          "ProgramDateTimeIntervalSeconds": 300,
          "Url": "https://c4af3793bf76b33c.mediapackage.us-
west-2.amazonaws.com/out/v1/1dc6718be36f4f34bb9cd86bc50925e6/cmaf_sports_endpoint/
index.m3u8"
        }
      ],
      "SegmentDurationSeconds": 2,
      "SegmentPrefix": "sportschannel"
    },
    "Description": "cmf output of sports",
    "Id": "cmf_sports",
    "ManifestName": "sports_channel",
    "StartoverWindowSeconds": 300,
    "Tags": {
      "region": "west",
      "media": "sports"
    },
    "TimeDelaySeconds": 10,
    "Url": "",
    "Whitelist": []
  }
}

```

Para obtener más información, consulte [Creación de un punto final](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [CreateOriginEndpoint](#) la Referencia de AWS CLI comandos.

## delete-channel

En el siguiente ejemplo de código, se muestra cómo usar `delete-channel`.

## AWS CLI

Para eliminar un canal

El siguiente `delete-channel` comando elimina el canal nombrado `test`.

```
aws mediapackage delete-channel \  
  --id test
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un canal](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [DeleteChannel](#) la Referencia de AWS CLI comandos.

## `delete-origin-endpoint`

En el siguiente ejemplo de código, se muestra cómo usar `delete-origin-endpoint`.

## AWS CLI

Para eliminar un punto final de origen

El siguiente `delete-origin-endpoint` comando elimina el punto final de origen denominado `tester2`.

```
aws mediapackage delete-origin-endpoint \  
  --id tester2
```

Para obtener más información, consulte [Eliminar un punto final](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [DeleteOriginEndpoint](#) la Referencia de AWS CLI comandos.

## `describe-channel`

En el siguiente ejemplo de código, se muestra cómo usar `describe-channel`.



## AWS CLI

Para describir un canal

El siguiente `describe-channel` comando muestra todos los detalles del canal nombrado `test`.

```
aws mediapackage describe-channel \
  --id test
```

Salida:

```
{
  "Arn": "arn:aws:mediapackage:us-
west-2:111222333:channels/584797f1740548c389a273585dd22a63",
  "HlsIngest": {
    "IngestEndpoints": [
      {
        "Id": "584797f1740548c389a273585dd22a63",
        "Password": "webdavgeneratedpassword1",
        "Url": "https://9be9c4405c474882.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/584797f1740548c389a273585dd22a63/channel",
        "Username": "webdavgeneratedusername1"
      },
      {
        "Id": "7d187c8616fd455f88aaa5a9fcf74442",
        "Password": "webdavgeneratedpassword2",
        "Url": "https://7bf454c57220328d.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/7d187c8616fd455f88aaa5a9fcf74442/channel",
        "Username": "webdavgeneratedusername2"
      }
    ]
  },
  "Id": "test",
  "Tags": {}
}
```

Para obtener más información, consulte [Visualización de los detalles del canal](https://docs.aws.amazon.com/mediapackage/latest/ug/channels-view.html) en la Guía del usuario de Elemental AWS MediaPackage

- Para obtener API más información, [DescribeChannel](#) consulte AWS CLI la Referencia de comandos.

## describe-origin-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `describe-origin-endpoint`.

### AWS CLI

Para describir un punto final de origen

El siguiente `describe-origin-endpoint` comando muestra todos los detalles del punto final de origen denominado `cmaf_sports`.

```
aws mediapackage describe-origin-endpoint \  
  --id cmaf_sports
```

Salida:

```
{  
  "Arn": "arn:aws:mediapackage:us-  
west-2:111222333:origin_endpoints/1dc6718be36f4f34bb9cd86bc50925e6",  
  "ChannelId": "sportschannel",  
  "CmafPackage": {  
    "HlsManifests": [  
      {  
        "AdMarkers": "NONE",  
        "Id": "cmaf_sports_endpoint",  
        "IncludeIframeOnlyStream": false,  
        "PlaylistType": "EVENT",  
        "PlaylistWindowSeconds": 60,  
        "ProgramDateTimeIntervalSeconds": 0,  
        "Url": "https://c4af3793bf76b33c.mediapackage.us-  
west-2.amazonaws.com/out/v1/1dc6718be36f4f34bb9cd86bc50925e6/cmaf_sports_endpoint/  
index.m3u8"  
      }  
    ],  
    "SegmentDurationSeconds": 2,  
    "SegmentPrefix": "sportschannel"  
  },  
  "Id": "cmaf_sports",  
  "ManifestName": "index",
```

```
"StartoverWindowSeconds": 0,
"Tags": {
  "region": "west",
  "media": "sports"
},
"TimeDelaySeconds": 0,
"Url": "",
"Whitelist": []
}
```

Para obtener más información, consulte [Visualización de un único punto final](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [DescribeOriginEndpoint](#) la Referencia de AWS CLI comandos.

## list-channels

En el siguiente ejemplo de código, se muestra cómo usar `list-channels`.

### AWS CLI

Creación de una lista de todos los canales

El siguiente `list-channels` comando muestra todos los canales que están configurados en la AWS cuenta corriente.

```
aws mediapackage list-channels
```

Salida:

```
{
  "Channels": [
    {
      "Arn": "arn:aws:mediapackage:us-
west-2:111222333:channels/584797f1740548c389a273585dd22a63",
      "HlsIngest": {
        "IngestEndpoints": [
          {
            "Id": "584797f1740548c389a273585dd22a63",
            "Password": "webdavgeneratedpassword1",
```

```

        "Url": "https://9be9c4405c474882.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/584797f1740548c389a273585dd22a63/channel",
        "Username": "webdavgeneratedusername1"
    },
    {
        "Id": "7d187c8616fd455f88aaa5a9fcf74442",
        "Password": "webdavgeneratedpassword2",
        "Url": "https://7bf454c57220328d.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/7d187c8616fd455f88aaa5a9fcf74442/channel",
        "Username": "webdavgeneratedusername2"
    }
]
},
"Id": "test",
"Tags": {}
}
]
}

```

Para obtener más información, consulte [Visualización de los detalles de los canales](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [ListChannels](#) la Referencia de AWS CLI comandos.

## list-origin-endpoints

En el siguiente ejemplo de código, se muestra cómo usar `list-origin-endpoints`.

### AWS CLI

Creación de una lista de todos los puntos de conexión de origen de un canal

El siguiente comando `list-origin-endpoints` muestra todos los puntos de conexión de origen que están configurados en el canal denominado `test`.

```
aws mediapackage list-origin-endpoints \
  --channel-id test
```

Salida:

```
{
```

```
"OriginEndpoints": [  
  {  
    "Arn": "arn:aws:mediapackage:us-  
west-2:111222333:origin_endpoints/247cff871f2845d3805129be22f2c0a2",  
    "ChannelId": "test",  
    "DashPackage": {  
      "ManifestLayout": "FULL",  
      "ManifestWindowSeconds": 60,  
      "MinBufferTimeSeconds": 30,  
      "MinUpdatePeriodSeconds": 15,  
      "PeriodTriggers": [],  
      "Profile": "NONE",  
      "SegmentDurationSeconds": 2,  
      "SegmentTemplateFormat": "NUMBER_WITH_TIMELINE",  
      "StreamSelection": {  
        "MaxVideoBitsPerSecond": 2147483647,  
        "MinVideoBitsPerSecond": 0,  
        "StreamOrder": "ORIGINAL"  
      },  
      "SuggestedPresentationDelaySeconds": 25  
    },  
    "Id": "tester2",  
    "ManifestName": "index",  
    "StartoverWindowSeconds": 0,  
    "Tags": {},  
    "TimeDelaySeconds": 0,  
    "Url": "https://8343f7014c0ea438.mediapackage.us-west-2.amazonaws.com/  
out/v1/247cff871f2845d3805129be22f2c0a2/index.mpd",  
    "Whitelist": []  
  },  
  {  
    "Arn": "arn:aws:mediapackage:us-  
west-2:111222333:origin_endpoints/869e237f851549e9bcf10e3bc2830839",  
    "ChannelId": "test",  
    "HlsPackage": {  
      "AdMarkers": "NONE",  
      "IncludeIframeOnlyStream": false,  
      "PlaylistType": "EVENT",  
      "PlaylistWindowSeconds": 60,  
      "ProgramDateTimeIntervalSeconds": 0,  
      "SegmentDurationSeconds": 6,  
      "StreamSelection": {  
        "MaxVideoBitsPerSecond": 2147483647,  
        "MinVideoBitsPerSecond": 0,  

```

```

        "StreamOrder": "ORIGINAL"
      },
      "UseAudioRenditionGroup": false
    },
    "Id": "tester",
    "ManifestName": "index",
    "StartoverWindowSeconds": 0,
    "Tags": {},
    "TimeDelaySeconds": 0,
    "Url": "https://8343f7014c0ea438.mediapackage.us-west-2.amazonaws.com/
out/v1/869e237f851549e9bcf10e3bc2830839/index.m3u8",
    "Whitelist": []
  }
]
}

```

Para obtener más información, consulte [Visualización de todos los puntos finales asociados a un canal](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [ListOriginEndpoints](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas asignadas a un recurso

El siguiente `list-tags-for-resource` comando muestra las etiquetas asignadas al recurso especificado.

```

aws mediapackage list-tags-for-resource \
  --resource-arn arn:aws:mediapackage:us-
west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0

```

Salida:

```

{
  "Tags": {
    "region": "west"
  }
}

```

```
}  
}
```

Para obtener más información, consulte [Etiquetar recursos en AWS Elemental MediaPackage en la Guía del MediaPackage usuario de AWS Elemental](#).

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## rotate-ingest-endpoint-credentials

En el siguiente ejemplo de código, se muestra cómo usar `rotate-ingest-endpoint-credentials`.

### AWS CLI

Para rotar las credenciales de ingesta

El siguiente `rotate-ingest-endpoint-credentials` comando rota el nombre de DAV usuario y la contraseña web del punto final de ingesta especificado.

```
aws mediapackage rotate-ingest-endpoint-credentials \  
  --id test \  
  --ingest-endpoint-id 584797f1740548c389a273585dd22a63
```

Salida:

```
{  
  "Arn": "arn:aws:mediapackage:us-west-2:111222333:channels/584797f1740548c389a273585dd22a63",  
  "HlsIngest": {  
    "IngestEndpoints": [  
      {  
        "Id": "584797f1740548c389a273585dd22a63",  
        "Password": "webdavregeneratedpassword1",  
        "Url": "https://9be9c4405c474882.mediapackage.us-west-2.amazonaws.com/in/v2/584797f1740548c389a273585dd22a63/584797f1740548c389a273585dd22a63/channel",  
        "Username": "webdavregeneratedusername1"  
      },  
      {
```

```

        "Id": "7d187c8616fd455f88aaa5a9fcf74442",
        "Password": "webdavgeneratedpassword2",
        "Url": "https://7bf454c57220328d.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/7d187c8616fd455f88aaa5a9fcf74442/channel",
        "Username": "webdavgeneratedusername2"
    }
]
},
    "Id": "test",
    "Tags": {}
}

```

Para obtener más información, consulte [Rotación de credenciales en una entrada URL en](#) la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [RotateIngestEndpointCredentials](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

### AWS CLI

Cómo añadir una etiqueta a un recurso

Los siguientes tag-resource comandos añaden un par de region=west clave y valor al recurso especificado.

```

aws mediapackage tag-resource \
  --resource-arn arn:aws:mediapackage:us-
west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0 \
  --tags region=west

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar recursos en AWS Elemental MediaPackage en](#) la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.



## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar una etiqueta de un recurso

El siguiente `untag-resource` comando elimina la etiqueta con la clave `region` del canal especificado.

```
aws mediapackage untag-resource \  
  --resource-arn arn:aws:mediapackage:us-west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0 \  
  --tag-keys region
```

Para obtener más información, consulte [Etiquetar recursos en AWS Elemental MediaPackage en la Guía del MediaPackage usuario de AWS Elemental](#).

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-channel

En el siguiente ejemplo de código, se muestra cómo usar `update-channel`.

### AWS CLI

Para actualizar un canal

El siguiente `update-channel` comando actualiza el canal nombrado `sportschannel` para incluir la descripción `24x7 sports`.

```
aws mediapackage update-channel \  
  --id sportschannel \  
  --description "24x7 sports"
```

Salida:

```
{  
  "Arn": "arn:aws:mediapackage:us-west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0",
```

```

    "Description": "24x7 sports",
    "HlsIngest": {
      "IngestEndpoints": [
        {
          "Id": "6d345804ec3f46c9b454a91d4a80d0e0",
          "Password": "generatedwebdavpassword1",
          "Url": "https://f31c86aed53b815a.mediapackage.us-
west-2.amazonaws.com/in/
v2/6d345804ec3f46c9b454a91d4a80d0e0/6d345804ec3f46c9b454a91d4a80d0e0/channel",
          "Username": "generatedwebdavusername1"
        },
        {
          "Id": "2daa32878af24803b24183727211b8ff",
          "Password": "generatedwebdavpassword2",
          "Url": "https://6ebbe7e04c4b0afa.mediapackage.us-
west-2.amazonaws.com/in/
v2/6d345804ec3f46c9b454a91d4a80d0e0/2daa32878af24803b24183727211b8ff/channel",
          "Username": "generatedwebdavusername2"
        }
      ]
    },
    "Id": "sportschannel",
    "Tags": {}
  }
}

```

Para obtener más información, consulte [Edición de un canal](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [UpdateChannel](#) en la Referencia de AWS CLI comandos.

## update-origin-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `update-origin-endpoint`.

### AWS CLI

Para actualizar un punto final de origen

El siguiente `update-origin-endpoint` comando actualiza el punto final de origen denominado `ocmaf_sports`. Cambia el tiempo de retardo a 0 segundos.

```
aws mediapackage update-origin-endpoint \
```

```
--id cmf_sports \  
--time-delay-seconds 0
```

Salida:

```
{  
  "Arn": "arn:aws:mediapackage:us-  
west-2:111222333:origin_endpoints/1dc6718be36f4f34bb9cd86bc50925e6",  
  "ChannelId": "sportschannel",  
  "CmafPackage": {  
    "HlsManifests": [  
      {  
        "AdMarkers": "NONE",  
        "Id": "cmf_sports_endpoint",  
        "IncludeIframeOnlyStream": false,  
        "PlaylistType": "EVENT",  
        "PlaylistWindowSeconds": 60,  
        "ProgramDateTimeIntervalSeconds": 0,  
        "Url": "https://c4af3793bf76b33c.mediapackage.us-  
west-2.amazonaws.com/out/v1/1dc6718be36f4f34bb9cd86bc50925e6/cmf_sports_endpoint/  
index.m3u8"  
      }  
    ],  
    "SegmentDurationSeconds": 2,  
    "SegmentPrefix": "sportschannel"  
  },  
  "Id": "cmf_sports",  
  "ManifestName": "index",  
  "StartoverWindowSeconds": 0,  
  "Tags": {  
    "region": "west",  
    "media": "sports"  
  },  
  "TimeDelaySeconds": 0,  
  "Url": "",  
  "Whitelist": []  
}
```

Para obtener más información, consulte [Edición de un punto final](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [UpdateOriginEndpoint](#) la Referencia de AWS CLI comandos.

## MediaPackage VODejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with MediaPackage VOD.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-asset**

En el siguiente ejemplo de código, se muestra cómo usar create-asset.

#### AWS CLI

Para crear un activo

En el siguiente create-asset ejemplo, se crea un activo con un nombre Chicken\_Asset en la AWS cuenta corriente. El activo ingiere el archivo 30sec\_chicken.smil en. MediaPackage

```
aws mediapackage-vod create-asset \  
  --id chicken_asset \  
  --packaging-group-id hls_chicken_gp \  
  --source-role-arn arn:aws:iam::111122223333:role/EMP_Vod \  
  --source-arn arn:aws:s3::111122223333:video-bucket/A/30sec_chicken.smil
```

Salida:

```
{  
  "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:assets/chicken_asset",  
  "Id": "chicken_asset",  
  "PackagingGroupId": "hls_chicken_gp",  
  "SourceArn": "arn:aws:s3::111122223333:video-bucket/A/30sec_chicken.smil",
```

```

"SourceRoleArn": "arn:aws:iam::111122223333:role/EMP_Vod",
"EgressEndpoints": [
  {
    "PackagingConfigurationId": "New_config_1",
    "Url": "https://c75ea2668ab49d02bca7ae10ef31c59e.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/
v1/6644b55df1744261ab3732a8e5cdaf07/904b06a58c7645e08d57d40d064216ac/
f5b2e633ff4942228095d164c10074f3/index.m3u8"
  },
  {
    "PackagingConfigurationId": "new_hls",
    "Url": " https://c75ea2668ab49d02bca7ae10ef31c59e.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/v1/6644b55df1744261ab3732a8e5cdaf07/
fe8f1f00a80e424cb4f8da4095835e9e/7370ec57432343af816332356d2bd5c6/string.m3u8"
  }
]
}

```

Para obtener más información, consulte [Ingerir un activo](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [CreateAsset](#) la Referencia de AWS CLI comandos.

## create-packaging-configuration

En el siguiente ejemplo de código, se muestra cómo usar `create-packaging-configuration`.

### AWS CLI

Para crear una configuración de empaquetado

En el siguiente `create-packaging-configuration` ejemplo, se crea una configuración de empaquetado nombrada `new_hls` en el grupo de empaquetado denominado `hls_chicken`. En este ejemplo, se utiliza un archivo en el disco denominado `hls_pc.json` para proporcionar los detalles.

```

aws mediapackage-vod create-packaging-configuration \
  --id new_hls \
  --packaging-group-id hls_chicken \
  --hls-package file://hls_pc.json

```

Contenidos de `hls_pc.json`:

```
{
  "HlsManifests":[
    {
      "AdMarkers":"NONE",
      "IncludeIframeOnlyStream":false,
      "ManifestName":"string",
      "ProgramDateTimeIntervalSeconds":60,
      "RepeatExtXKey":true,
      "StreamSelection":{
        "MaxVideoBitsPerSecond":1000,
        "MinVideoBitsPerSecond":0,
        "StreamOrder":"ORIGINAL"
      }
    }
  ],
  "SegmentDurationSeconds":6,
  "UseAudioRenditionGroup":false
}
```

**Salida:**

```
{
  "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-configurations/new_hls",
  "Id":"new_hls",
  "PackagingGroupId":"hls_chicken",
  "HlsManifests":{
    "SegmentDurationSeconds":6,
    "UseAudioRenditionGroup":false,
    "HlsMarkers":[
      {
        "AdMarkers":"NONE",
        "IncludeIframeOnlyStream":false,
        "ManifestName":"string",
        "ProgramDateTimeIntervalSeconds":60,
        "RepeatExtXKey":true,
        "StreamSelection":{
          "MaxVideoBitsPerSecond":1000,
          "MinVideoBitsPerSecond":0,
          "StreamOrder":"ORIGINAL"
        }
      }
    ]
  }
}
```

```
}  
}
```

Para obtener más información, consulte [Creación de una configuración de empaquetado](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [CreatePackagingConfiguration](#) la Referencia de AWS CLI comandos.

## create-packaging-group

En el siguiente ejemplo de código, se muestra cómo usar `create-packaging-group`.

### AWS CLI

Para crear un grupo de empaquetado

En el siguiente `create-packaging-group` ejemplo, se enumeran todos los grupos de empaquetado que están configurados en la AWS cuenta actual.

```
aws mediapackage-vod create-packaging-group \  
  --id hls_chicken
```

Salida:

```
{  
  "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-groups/  
hls_chicken",  
  "Id": "hls_chicken"  
}
```

Para obtener más información, consulte [Creación de un grupo de empaquetado](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [CreatePackagingGroup](#) la Referencia de AWS CLI comandos.

## delete-asset

En el siguiente ejemplo de código, se muestra cómo usar `delete-asset`.

## AWS CLI

Para eliminar un activo

En el siguiente `delete-asset` ejemplo, se elimina el activo denominado `30sec_chicken`.

```
aws mediapackage-vod delete-asset \  
  --id 30sec_chicken
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un activo](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [DeleteAsset](#) la Referencia de AWS CLI comandos.

## `delete-packaging-configuration`

En el siguiente ejemplo de código, se muestra cómo usar `delete-packaging-configuration`.

## AWS CLI

Para eliminar una configuración de empaquetado

En el siguiente `delete-packaging-configuration` ejemplo, se elimina la configuración de empaquetado denominada `CMAF`.

```
aws mediapackage-vod delete-packaging-configuration \  
  --id CMAF
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar una configuración de empaquetado](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [DeletePackagingConfiguration](#) la Referencia de AWS CLI comandos.

## `delete-packaging-group`

En el siguiente ejemplo de código, se muestra cómo usar `delete-packaging-group`.



## AWS CLI

Para eliminar un grupo de empaquetado

En el siguiente `delete-packaging-group` ejemplo, se elimina el grupo de empaquetado denominado `Dash_widevine`.

```
aws mediapackage-vod delete-packaging-group \  
  --id Dash_widevine
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un grupo de empaquetado](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [DeletePackagingGroup](#) la Referencia de AWS CLI comandos.

## `describe-asset`

En el siguiente ejemplo de código, se muestra cómo usar `describe-asset`.

## AWS CLI

Para describir un activo

En el siguiente `describe-asset` ejemplo, se muestran todos los detalles del activo denominado `30sec_chicken`.

```
aws mediapackage-vod describe-asset \  
  --id 30sec_chicken
```

Salida:

```
{  
  "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:assets/30sec_chicken",  
  "Id": "30sec_chicken",  
  "PackagingGroupId": "Packaging_group_1",  
  "SourceArn": "arn:aws:s3::111122223333:video-bucket/A/30sec_chicken.smil",  
  "SourceRoleArn": "arn:aws:iam::111122223333:role/EMP_Vod",  
  "EgressEndpoints": [  
    {
```

```

        "PackagingConfigurationId": "DASH",
        "Url": "https://a5f46a44118ba3e3724ef39ef532e701.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/v1/
aad7962c569946119c2d5a691be5663c/66c25aff456d463aae0855172b3beb27/4ddfda6da17c4c279a1b8401cb
index.mpd"
    },
    {
        "PackagingConfigurationId": "HLS",
        "Url": "https://a5f46a44118ba3e3724ef39ef532e701.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/v1/
aad7962c569946119c2d5a691be5663c/6e5bf286a3414254a2bf0d22ae148d7e/06b5875b4d004c3cbdc4da2dc4
index.m3u8"
    },
    {
        "PackagingConfigurationId": "CMAF",
        "Url": "https://a5f46a44118ba3e3724ef39ef532e701.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/v1/
aad7962c569946119c2d5a691be5663c/628fb5d8d89e4702958b020af27fde0e/05eb062214064238ad6330a443
index.m3u8"
    }
]
}

```

Para obtener más información, consulte [Visualización de los detalles del activo](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [DescribeAsset](#) la Referencia de AWS CLI comandos.

## describe-packaging-configuration

En el siguiente ejemplo de código, se muestra cómo usar `describe-packaging-configuration`.

### AWS CLI

Para describir una configuración de empaquetado

El siguiente `describe-packaging-configuration` ejemplo muestra todos los detalles de la configuración de empaquetado denominada DASH.

```
aws mediapackage-vod describe-packaging-configuration \
```

```
--id DASH
```

Salida:

```
{
  "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-configurations/
DASH",
  "Id": "DASH",
  "PackagingGroupId": "Packaging_group_1",
  "DashPackage": [
    {
      "SegmentDurationSeconds": "2"
    },
    {
      "DashManifests": {
        "ManifestName": "index",
        "MinBufferTimeSeconds": "30",
        "Profile": "NONE"
      }
    }
  ]
}
```

Para obtener más información, consulte [Visualización de los detalles de la configuración del embalaje](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [DescribePackagingConfiguration](#) la Referencia de AWS CLI comandos.

## describe-packaging-group

En el siguiente ejemplo de código, se muestra cómo usar describe-packaging-group.

AWS CLI

Para describir un grupo de empaquetado

En el siguiente describe-packaging-group ejemplo, se muestran todos los detalles del grupo de embalaje denominado Packaging\_group\_1.

```
aws mediapackage-vod describe-packaging-group \
  --id Packaging_group_1
```

Salida:

```
{
  "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-groups/
Packaging_group_1",
  "Id": "Packaging_group_1"
}
```

Para obtener más información, consulte [Visualización de los detalles del grupo de embalaje](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [DescribePackagingGroup](#) la Referencia de AWS CLI comandos.

## list-assets

En el siguiente ejemplo de código, se muestra cómo usar `list-assets`.

AWS CLI

Para enumerar todos los activos

En el siguiente `list-assets` ejemplo, se enumeran todos los activos que están configurados en la AWS cuenta corriente.

```
aws mediapackage-vod list-assets
```

Salida:

```
{
  "Assets": [
    "Arn": "arn:aws:mediapackage-vod:us-
west-2:111122223333:assets/30sec_chicken",
    "Id": "30sec_chicken",
    "PackagingGroupId": "Packaging_group_1",
    "SourceArn": "arn:aws:s3::111122223333:video-bucket/A/30sec_chicken.smil",
    "SourceRoleArn": "arn:aws:iam::111122223333:role/EMP_Vod"
  ]
}
```

Para obtener más información, consulte [Visualización de los detalles de los activos](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [ListAssets](#) la Referencia de AWS CLI comandos.

## list-packaging-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-packaging-configurations`.

### AWS CLI

Para ver una lista de todas las configuraciones de empaquetado

En el siguiente `list-packaging-configurations` ejemplo, se enumeran todas las configuraciones de empaquetado que están configuradas en el grupo de empaquetado denominado `Packaging_group_1`.

```
aws mediapackage-vod list-packaging-configurations \
  --packaging-group-id Packaging_group_1
```

Salida:

```
{
  "PackagingConfigurations": [
    {
      "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
configurations/CMAF",
      "Id": "CMAF",
      "PackagingGroupId": "Packaging_group_1",
      "CmafPackage": [
        {
          "SegmentDurationSeconds": "2"
        },
        {
          "HlsManifests": {
            "AdMarkers": "NONE",
            "RepeatExtXKey": "False",
            "ManifestName": "index",
            "ProgramDateTimeIntervalSeconds": "0",
            "IncludeIframeOnlyStream": "False"
          }
        }
      ]
    },
    {
```

```

        "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
configurations/DASH",
        "Id":"DASH",
        "PackagingGroupId":"Packaging_group_1",
        "DashPackage":[
            {
                "SegmentDurationSeconds":"2"
            },
            {
                "DashManifests":{
                    "ManifestName":"index",
                    "MinBufferTimeSeconds":"30",
                    "Profile":"NONE"
                }
            }
        ]
    },
    {
        "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
configurations/HLS",
        "Id":"HLS",
        "PackagingGroupId":"Packaging_group_1",
        "HlsPackage":[
            {
                "SegmentDurationSeconds":"6",
                "UseAudioRenditionGroup":"False"
            },
            {
                "HlsManifests":{
                    "AdMarkers":"NONE",
                    "RepeatExtXKey":"False",
                    "ManifestName":"index",
                    "ProgramDateTimeIntervalSeconds":"0",
                    "IncludeIframeOnlyStream":"False"
                }
            }
        ]
    },
    {
        "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
configurations/New_config_0_copy",
        "Id":"New_config_0_copy",
        "PackagingGroupId":"Packaging_group_1",
        "HlsPackage":[

```

```

    {
      "SegmentDurationSeconds":"6",
      "UseAudioRenditionGroup":"False"
    },
    {
      "Encryption":{
        "EncryptionMethod":"AWS_128",
        "SpekeKeyProvider":{
          "RoleArn":"arn:aws:iam:111122223333::role/SPEKERole",
          "Url":"https://lfgubdvs97.execute-api.us-
west-2.amazonaws.com/EkeStage/copyProtection/",
          "SystemIds":[
            "81376844-f976-481e-a84e-cc25d39b0b33"
          ]
        }
      }
    },
    {
      "HlsManifests":{
        "AdMarkers":"NONE",
        "RepeatExtXKey":"False",
        "ManifestName":"index",
        "ProgramDateTimeIntervalSeconds":"0",
        "IncludeIframeOnlyStream":"False"
      }
    }
  ]
}

```

Para obtener más información, consulte [Visualización de los detalles de la configuración de empaquetado](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [ListPackagingConfigurations](#) la Referencia de AWS CLI comandos.

## list-packaging-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-packaging-groups`.

## AWS CLI

Para ver una lista de todos los grupos de empaquetado

En el siguiente `list-packaging-groups` ejemplo, se enumeran todos los grupos de empaquetado que están configurados en la AWS cuenta actual.

```
aws mediapackage-vod list-packaging-groups
```

Salida:

```
{
  "PackagingGroups": [
    {
      "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
groups/Dash_widevine",
      "Id": "Dash_widevine"
    },
    {
      "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
groups/Encrypted_HLS",
      "Id": "Encrypted_HLS"
    },
    {
      "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
groups/Packaging_group_1",
      "Id": "Packaging_group_1"
    }
  ]
}
```

Para obtener más información, consulte [Visualización de los detalles de los grupos de embalaje](#) en la Guía del MediaPackage usuario de AWS Elemental.

- Para API obtener más información, consulte [ListPackagingGroups](#) la Referencia de AWS CLI comandos.

## MediaStore Ejemplos de planos de datos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el AWS Command Line Interface uso de MediaStore Data Plane.



Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **delete-object**

En el siguiente ejemplo de código, se muestra cómo usar `delete-object`.

AWS CLI

Para eliminar un objeto

En el siguiente `delete-object` ejemplo, se elimina el objeto especificado.

```
aws mediastore-data delete-object \  
  --endpoint=https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --path=/folder_name/README.md
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un objeto](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [DeleteObject](#) la Referencia de AWS CLI comandos.

### **describe-object**

En el siguiente ejemplo de código, se muestra cómo usar `describe-object`.

AWS CLI

Para ver los encabezados de un objeto

En el siguiente describe-object ejemplo, se muestran los encabezados de un objeto en la ruta especificada.

```
aws mediastore-data describe-object \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --path events/baseball/setup.jpg
```

Salida:

```
{  
  "LastModified": "Fri, 19 Jul 2019 21:50:31 GMT",  
  "ContentType": "image/jpeg",  
  "ContentLength": "3860266",  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9e9999e4dd89ff7f5555555555555555da6d3"  
}
```

Para obtener más información, consulte [Visualización de los detalles de un objeto](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [DescribeObject](#) la Referencia de AWS CLI comandos.

## get-object

En el siguiente ejemplo de código, se muestra cómo usar get-object.

AWS CLI

Ejemplo 1: Para descargar un objeto completo

En el siguiente get-object ejemplo, se descarga el objeto especificado.

```
aws mediastore-data get-object \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --path events/baseball/setup.jpg setup.jpg
```

Salida:

```
{  
  "ContentType": "image/jpeg",  
  "StatusCode": 200,  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9e9999e4dd89ff7f5555555555555555da6d3",  
}
```

```
"ContentLength": "3860266",  
"LastModified": "Fri, 19 Jul 2019 21:50:31 GMT"  
}
```

Ejemplo 2: Para descargar parte de un objeto

En el siguiente `get-object` ejemplo, se descarga la parte especificada de un objeto.

```
aws mediastore-data get-object \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --path events/baseball/setup.jpg setup.jpg \  
  --range "bytes=0-100"
```

Salida:

```
{  
  "StatusCode": 206,  
  "LastModified": "Fri, 19 Jul 2019 21:50:31 GMT",  
  "ContentType": "image/jpeg",  
  "ContentRange": "bytes 0-100/3860266",  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f5555555555555555da6d3",  
  "ContentLength": "101"  
}
```

Para obtener más información, consulte [Descarga de un objeto](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [GetObject](#) la Referencia de AWS CLI comandos.

## list-items

En el siguiente ejemplo de código, se muestra cómo usar `list-items`.

### AWS CLI

Ejemplo 1: Para ver una lista de elementos (objetos y carpetas) almacenados en un contenedor

En el siguiente `list-items` ejemplo, se muestra una lista de elementos (objetos y carpetas) almacenados en el contenedor especificado.

```
aws mediastore-data list-items \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com
```

**Salida:**

```
{
  "Items": [
    {
      "Type": "OBJECT",
      "ContentLength": 3784,
      "Name": "setup.jpg",
      "ETag":
"2aa333bbcc8d8d22d777e999c88d4aa9e9999e4dd89ff7f5555555555555555da6d3",
      "ContentType": "image/jpeg",
      "LastModified": 1563571859.379
    },
    {
      "Type": "FOLDER",
      "Name": "events"
    }
  ]
}
```

Ejemplo 2: Para ver una lista de elementos (objetos y carpetas) almacenados en una carpeta

En el siguiente `list-items` ejemplo, se muestra una lista de elementos (objetos y carpetas) almacenados en la carpeta especificada.

```
aws mediastore-data list-items \
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com \
  --path events/baseball
```

**Salida:**

```
{
  "Items": [
    {
      "ETag":
"2aa333bbcc8d8d22d777e999c88d4aa9e9999e4dd89ff7f5555555555555555da6d3",
      "ContentType": "image/jpeg",
      "Type": "OBJECT",
      "ContentLength": 3860266,
      "LastModified": 1563573031.872,
      "Name": "setup.jpg"
    }
  ]
}
```

```
}
```

Para obtener más información, consulte [Visualización de una lista de objetos](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [ListItems](#) la Referencia de AWS CLI comandos.

## put-object

En el siguiente ejemplo de código, se muestra cómo usar `put-object`.

### AWS CLI

Ejemplo 1: Para cargar un objeto en un contenedor

En el siguiente `put-object` ejemplo, se carga un objeto en el contenedor especificado.

```
aws mediastore-data put-object \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --body ReadMe.md \  
  --path ReadMe.md \  
  --cache-control "max-age=6, public" \  
  --content-type binary/octet-stream
```

Salida:

```
{  
  "ContentSHA256":  
    "f29bc64a9d3732b4b9035125fdb3285f5b6455778edca72414671e0ca3b2e0de",  
  "StorageClass": "TEMPORAL",  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9e0000004dd89ff7f55555555555555da6d3"  
}
```

Ejemplo 2: Para cargar un objeto en una carpeta dentro de un contenedor

En el siguiente `put-object` ejemplo, se carga un objeto en la carpeta especificada dentro de un contenedor.

```
aws mediastore-data put-object \  
  --endpoint https://aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com \  
  --body ReadMe.md \  
  --path /september-events/ReadMe.md \  
  
```

```
--cache-control "max-age=6, public" \  
--content-type binary/octet-stream
```

Salida:

```
{  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9e4dd89ff7f5555555555555555da6d3",  
  "ContentSHA256":  
  "f29bc64a9d3732b4b9035125fdb3285f5b6455778edca72414671e0ca3b2e0de",  
  "StorageClass": "TEMPORAL"  
}
```

Para obtener más información, consulte [Carga de un objeto](#) en la Guía del MediaStore usuario de AWS Elemental.

- Para API obtener más información, consulte [PutObject](#) la Referencia de AWS CLI comandos.

## MediaTailor ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with MediaTailor.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **delete-playback-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `delete-playback-configuration`.

AWS CLI

Para eliminar una configuración

Lo siguiente `delete-playback-configuration` elimina una configuración denominada `campaign_short`.

```
aws mediatailor delete-playback-configuration \  
  --name campaign_short
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar una configuración](#) en la Guía del MediaTailor usuario de AWS Elemental.

- Para API obtener más información, consulte [DeletePlaybackConfiguration](#) la Referencia de AWS CLI comandos.

## get-playback-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-playback-configuration`.

AWS CLI

Para describir una configuración

A continuación `get-playback-configuration` se muestran todos los detalles de la configuración nombrada `west_campaign`.

```
aws mediatailor get-playback-configuration \  
  --name west_campaign
```

Salida:

```
{  
  "AdDecisionServerUrl": "http://your.ads.url",  
  "CdnConfiguration": {},  
  "DashConfiguration": {  
    "ManifestEndpointPrefix":  
    "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/  
dash/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/",  
    "MpdLocation": "EMT_DEFAULT",  
    "OriginManifestType": "MULTI_PERIOD"  
  },  
  "HlsConfiguration": {
```

```

    "ManifestEndpointPrefix":
      "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
      master/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/"
    },
    "Name": "west_campaign",
    "PlaybackConfigurationArn": "arn:aws:mediatailor:us-
    west-2:123456789012:playbackConfiguration/west_campaign",
    "PlaybackEndpointPrefix":
      "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com",
    "SessionInitializationEndpointPrefix":
      "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
      session/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/",
    "Tags": {},
    "VideoContentSourceUrl": "https://8343f7014c0ea438.mediapackage.us-
    west-2.amazonaws.com/out/v1/683f0f2ff7cd43a48902e6dcd5e16dcf/index.m3u8"
  }

```

Para obtener más información, consulte [Visualización de una configuración](#) en la Guía del MediaTailor usuario de AWS Elemental.

- Para API obtener más información, consulte [GetPlaybackConfiguration](#) la Referencia de AWS CLI comandos.

## list-playback-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-playback-configurations`.

### AWS CLI

Para ver una lista de todas las configuraciones

A continuación, se `list-playback-configurations` muestran todos los detalles de la configuración de la AWS cuenta corriente.

```
aws mediatailor list-playback-configurations
```

Salida:

```

{
  "Items": [
    {
      "AdDecisionServerUrl": "http://your.ads.url",
      "CdnConfiguration": {},

```



```

    "DashConfiguration": {
      "ManifestEndpointPrefix":
        "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
dash/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/",
      "MpdLocation": "EMT_DEFAULT",
      "OriginManifestType": "MULTI_PERIOD"
    },
    "HlsConfiguration": {
      "ManifestEndpointPrefix":
        "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
master/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/"
    },
    "Name": "west_campaign",
    "PlaybackConfigurationArn": "arn:aws:mediatailor:us-
west-2:123456789012:playbackConfiguration/west_campaign",
    "PlaybackEndpointPrefix":
      "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com",
    "SessionInitializationEndpointPrefix":
      "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
session/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/",
    "Tags": {},
    "VideoContentSourceUrl": "https://8343f7014c0ea438.mediapackage.us-
west-2.amazonaws.com/out/v1/683f0f2ff7cd43a48902e6dcd5e16dcf/index.m3u8"
  },
  {
    "AdDecisionServerUrl": "http://your.ads.url",
    "CdnConfiguration": {},
    "DashConfiguration": {
      "ManifestEndpointPrefix":
        "https://73511f91d6a24ca2b93f3cf1d7cedd67.mediatailor.us-west-2.amazonaws.com/v1/
dash/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/sports_campaign/",
      "MpdLocation": "DISABLED",
      "OriginManifestType": "MULTI_PERIOD"
    },
    "HlsConfiguration": {
      "ManifestEndpointPrefix":
        "https://73511f91d6a24ca2b93f3cf1d7cedd67.mediatailor.us-west-2.amazonaws.com/v1/
master/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/sports_campaign/"
    },
    "Name": "sports_campaign",
    "PlaybackConfigurationArn": "arn:aws:mediatailor:us-
west-2:123456789012:playbackConfiguration/sports_campaign",
    "PlaybackEndpointPrefix":
      "https://73511f91d6a24ca2b93f3cf1d7cedd67.mediatailor.us-west-2.amazonaws.com",

```

```

    "SessionInitializationEndpointPrefix":
      "https://73511f91d6a24ca2b93f3cf1d7cedd67.mediatailor.us-west-2.amazonaws.com/v1/
      session/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/sports_campaign/",
      "SlateAdUrl": "http://s3.bucket/slate_ad.mp4",
      "Tags": {},
      "VideoContentSourceUrl": "https://c4af3793bf76b33c.mediapackage.us-
      west-2.amazonaws.com/out/v1/1dc6718be36f4f34bb9cd86bc50925e6/sports_endpoint/
      index.m3u8"
    }
  ]
}

```

Para obtener más información, consulte [Visualización de una configuración](https://docs.aws.amazon.com/mediatailor/latest/ug/configurations-view.html) en la Guía del usuario de Elemental.AWS MediaTailor

- Para obtener API más información, [ListPlaybackConfigurations](#) consulte AWS CLI la Referencia de comandos.

## put-playback-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-playback-configuration`.

### AWS CLI

Para crear una configuración

A continuación se `put-playback-configuration` crea una configuración denominada `campaign_short`.

```

aws mediatailor put-playback-configuration \
  --name campaign_short \
  --ad-decision-server-url http://your.ads.url \
  --video-content-source-url http://video.bucket/index.m3u8

```

Salida:

```

{
  "AdDecisionServerUrl": "http://your.ads.url",
  "CdnConfiguration": {},
  "DashConfiguration": {

```

```

    "ManifestEndpointPrefix":
      "https://13484114d38f4383bc0d6a7cb879bd00.mediatailor.us-west-2.amazonaws.com/v1/
dash/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/campaign_short/",
      "MpdLocation": "EMT_DEFAULT",
      "OriginManifestType": "MULTI_PERIOD"
    },
    "HlsConfiguration": {
      "ManifestEndpointPrefix":
        "https://13484114d38f4383bc0d6a7cb879bd00.mediatailor.us-west-2.amazonaws.com/v1/
master/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/campaign_short/"
      },
      "Name": "campaign_short",
      "PlaybackConfigurationArn": "arn:aws:mediatailor:us-
west-2:123456789012:playbackConfiguration/campaign_short",
      "PlaybackEndpointPrefix":
        "https://13484114d38f4383bc0d6a7cb879bd00.mediatailor.us-west-2.amazonaws.com",
      "SessionInitializationEndpointPrefix":
        "https://13484114d38f4383bc0d6a7cb879bd00.mediatailor.us-west-2.amazonaws.com/v1/
session/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/campaign_short/",
      "Tags": {},
      "VideoContentSourceUrl": "http://video.bucket/index.m3u8"
    }
  }

```

Para obtener más información, consulte [Creación de una configuración](#) en la Guía del MediaTailor usuario de AWS Elemental.

- Para API obtener más información, consulte [PutPlaybackConfiguration](#) la Referencia de AWS CLI comandos.

## Ejemplos de MemoryDB usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el AWS Command Line Interface uso de MemoryDB.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### copy-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `copy-snapshot`.

#### AWS CLI

Para copiar una instantánea

En el siguiente `copy-snapshot` ejemplo, se crea una copia de una instantánea.

```
aws memorydb copy-snapshot \  
  --source-snapshot-name my-cluster-snapshot \  
  --target-snapshot-name my-cluster-snapshot-copy
```

#### Salida

```
{  
  "Snapshot": {  
    "Name": "my-cluster-snapshot-copy",  
    "Status": "creating",  
    "Source": "manual",  
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:snapshot/my-cluster-snapshot-copy",  
    "ClusterConfiguration": {  
      "Name": "my-cluster",  
      "Description": " ",  
      "NodeType": "db.r6g.large",  
      "EngineVersion": "6.2",  
      "MaintenanceWindow": "wed:03:00-wed:04:00",  
      "Port": 6379,  
      "ParameterGroupName": "default.memorydb-redis6",  
      "SubnetGroupName": "my-sg",  
      "VpcId": "vpc-xx2574fc",  
      "SnapshotRetentionLimit": 0,  
      "SnapshotWindow": "04:30-05:30",  
      "NumShards": 2  
    }  
  }  
}
```

```
}
```

Para obtener más información, consulte [Copiar una instantánea](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [CopySnapshot](#) de AWS CLI comandos.

## create-acl

En el siguiente ejemplo de código, se muestra cómo usar `create-acl`.

### AWS CLI

Para crear un ACL

El siguiente `create-acl` ejemplo crea una nueva lista de control de acceso.

```
aws memorydb create-acl \  
  --acl-name "new-acl-1" \  
  --user-names "my-user"
```

Salida:

```
{  
  "ACL": {  
    "Name": "new-acl-1",  
    "Status": "creating",  
    "UserNames": [  
      "my-user"  
    ],  
    "MinimumEngineVersion": "6.2",  
    "Clusters": [],  
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:acl/new-acl-1"  
  }  
}
```

Para obtener más información, consulte [Autenticación de usuarios con listas de control de acceso](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [CreateAcl](#) comandos AWS CLI .

## create-cluster

En el siguiente ejemplo de código, se muestra cómo usar `create-cluster`.

### AWS CLI

Para crear un clúster

En el siguiente `create-cluster` ejemplo, se crea un clúster nuevo.

```
aws memorydb create-cluster \  
  --cluster-name my-new-cluster \  
  --node-type db.r6g.large \  
  --acl-name my-acl \  
  --subnet-group my-sg
```

Salida:

```
{  
  "Cluster": {  
    "Name": "my-new-cluster",  
    "Status": "creating",  
    "NumberOfShards": 1,  
    "AvailabilityMode": "MultiAZ",  
    "ClusterEndpoint": {  
      "Port": 6379  
    },  
    "NodeType": "db.r6g.large",  
    "EngineVersion": "6.2",  
    "EnginePatchVersion": "6.2.6",  
    "ParameterGroupName": "default.memorydb-redis6",  
    "ParameterGroupStatus": "in-sync",  
    "SubnetGroupName": "my-sg",  
    "TLSEnabled": true,  
    "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:cluster/my-new-cluster",  
    "SnapshotRetentionLimit": 0,  
    "MaintenanceWindow": "sat:10:00-sat:11:00",  
    "SnapshotWindow": "07:30-08:30",  
    "ACLName": "my-acl",  
    "AutoMinorVersionUpgrade": true  
  }  
}
```

Para obtener más información, consulte [Administración de clústeres](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [CreateCluster](#) de AWS CLI comandos.

## create-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `create-parameter-group`.

### AWS CLI

Para crear un grupo de parámetros

En el siguiente `create-parameter-group` ejemplo, se crea un grupo de parámetros.

```
aws memorydb create-parameter-group \  
  --parameter-group-name myRedis6x \  
  --family memorydb_redis6 \  
  --description "my-parameter-group"
```

Salida:

```
{  
  "ParameterGroup": {  
    "Name": "myredis6x",  
    "Family": "memorydb_redis6",  
    "Description": "my-parameter-group",  
    "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:parametergroup/myredis6x"  
  }  
}
```

Para obtener más información, consulte [Creación de un grupo de parámetros](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [CreateParameterGroup](#) de AWS CLI comandos.

## create-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `create-snapshot`.

## AWS CLI

Para crear una instantánea

En el siguiente `create-snapshot` ejemplo, se crea una instantánea.

```
aws memorydb create-snapshot \  
  --cluster-name my-cluster \  
  --snapshot-name my-cluster-snapshot
```

Salida:

```
{  
  "Snapshot": {  
    "Name": "my-cluster-snapshot1",  
    "Status": "creating",  
    "Source": "manual",  
    "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:snapshot/my-cluster-snapshot",  
    "ClusterConfiguration": {  
      "Name": "my-cluster",  
      "Description": "",  
      "NodeType": "db.r6g.large",  
      "EngineVersion": "6.2",  
      "MaintenanceWindow": "wed:03:00-wed:04:00",  
      "Port": 6379,  
      "ParameterGroupName": "default.memorydb-redis6",  
      "SubnetGroupName": "my-sg",  
      "VpcId": "vpc-862xxxxc",  
      "SnapshotRetentionLimit": 0,  
      "SnapshotWindow": "04:30-05:30",  
      "NumShards": 2  
    }  
  }  
}
```

Para obtener más información, consulte [Creación de instantáneas manuales](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [CreateSnapshot](#) comandos AWS CLI .



## create-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `create-subnet-group`.

### AWS CLI

Para crear un grupo de subredes

En el siguiente `create-subnet-group` ejemplo, se crea un grupo de subredes.

```
aws memorydb create-subnet-group \  
  --subnet-group-name mysubnetgroup \  
  --description "my subnet group" \  
  --subnet-ids subnet-5623xxxx
```

Salida:

```
{  
  "SubnetGroup": {  
    "Name": "mysubnetgroup",  
    "Description": "my subnet group",  
    "VpcId": "vpc-86257xxx",  
    "Subnets": [  
      {  
        "Identifier": "subnet-5623xxxx",  
        "AvailabilityZone": {  
          "Name": "us-east-1a"  
        }  
      }  
    ],  
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:subnetgroup/mysubnetgroup"  
  }  
}
```

Para obtener más información, consulte [Creación de un grupo de subredes](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [CreateSubnetGroup](#) comandos AWS CLI .

## create-user

En el siguiente ejemplo de código, se muestra cómo usar `create-user`.

## AWS CLI

Para crear un usuario

En el `create-user` ejemplo siguiente se crea un usuario nuevo.

```
aws memorydb create-user \  
  --user-name user-name-1 \  
  --access-string "~objects:* ~items:* ~public:*" \  
  --authentication-mode \  
    Passwords="enterapasswordhere",Type=password
```

Salida:

```
{  
  "User": {  
    "Name": "user-name-1",  
    "Status": "active",  
    "AccessString": "off ~objects:* ~items:* ~public:* resetchannels -@all",  
    "ACLNames": [],  
    "MinimumEngineVersion": "6.2",  
    "Authentication": {  
      "Type": "password",  
      "PasswordCount": 1  
    },  
    "ARN": "arn:aws:memorydb:us-west-2:491658xxxxxx:user/user-name-1"  
  }  
}
```

Para obtener más información, consulte [Autenticación de usuarios con listas de control de acceso](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [CreateUser](#) comandos AWS CLI .

## **delete-acl**

En el siguiente ejemplo de código, se muestra cómo usar `delete-acl`.

## AWS CLI

Para eliminar un ACL

En el siguiente `delete-acl` ejemplo, se elimina una lista de control de acceso.

```
aws memorydb delete-acl \  
  --acl-name "new-acl-1"
```

Salida:

```
{  
  "ACL": {  
    "Name": "new-acl-1",  
    "Status": "deleting",  
    "UserNames": [  
      "pat"  
    ],  
    "MinimumEngineVersion": "6.2",  
    "Clusters": [],  
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:acl/new-acl-1"  
  }  
}
```

Para obtener más información, consulte [Autenticación de usuarios con listas de control de acceso](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [DeleteAcl](#) comandos AWS CLI .

## delete-cluster

En el siguiente ejemplo de código, se muestra cómo usar `delete-cluster`.

AWS CLI

Para eliminar un clúster

En el siguiente `delete-cluster` ejemplo, se elimina un clúster.

```
aws memorydb delete-cluster \  
  --cluster-name my-new-cluster
```

Salida:

```
{  
  "Cluster": {  
    "Name": "my-new-cluster",  
    "Status": "deleting",
```

```
    "NumberOfShards": 1,
    "ClusterEndpoint": {
      "Address": "clustercfg.my-new-cluster.xxxxx.memorydb.us-
east-1.amazonaws.com",
      "Port": 6379
    },
    "NodeType": "db.r6g.large",
    "EngineVersion": "6.2",
    "EnginePatchVersion": "6.2.6",
    "ParameterGroupName": "default.memorydb-redis6",
    "ParameterGroupStatus": "in-sync",
    "SubnetGroupName": "my-sg",
    "TLSEnabled": true,
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-new-cluster",
    "SnapshotRetentionLimit": 0,
    "MaintenanceWindow": "sat:10:00-sat:11:00",
    "SnapshotWindow": "07:30-08:30",
    "AutoMinorVersionUpgrade": true
  }
}
```

Para obtener más información, consulte [Eliminar un clúster](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [DeleteCluster](#) de AWS CLI comandos.

## delete-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-parameter-group`.

### AWS CLI

Para eliminar un grupo de parámetros

En el siguiente `delete-parameter-group` ejemplo, se elimina un grupo de parámetros.

```
aws memorydb delete-parameter-group \
  --parameter-group-name myRedis6x
```

Salida:

```
{
  "ParameterGroup": {
```

```

    "Name": "myredis6x",
    "Family": "memorydb_redis6",
    "Description": "my-parameter-group",
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:parametergroup/myredis6x"
  }
}

```

Para obtener más información, consulte [Eliminar un grupo de parámetros](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [DeleteParameterGroup](#) de AWS CLI comandos.

## delete-snapshot

En el siguiente ejemplo de código, se muestra cómo usar delete-snapshot.

### AWS CLI

Eliminar una instantánea

En el siguiente delete-snapshot ejemplo, se elimina una instantánea.

```

aws memorydb delete-snapshot \
  --snapshot-name my-cluster-snapshot

```

Salida:

```

{
  "Snapshot": {
    "Name": "my-cluster-snapshot",
    "Status": "deleting",
    "Source": "manual",
    "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:snapshot/my-cluster-
snapshot",
    "ClusterConfiguration": {
      "Name": "my-cluster",
      "Description": "",
      "NodeType": "db.r6g.large",
      "EngineVersion": "6.2",
      "MaintenanceWindow": "wed:03:00-wed:04:00",
      "Port": 6379,

```

```

        "ParameterGroupName": "default.memorydb-redis6",
        "SubnetGroupName": "my-sg",
        "VpcId": "vpc-862xxxxc",
        "SnapshotRetentionLimit": 0,
        "SnapshotWindow": "04:30-05:30",
        "NumShards": 2
    }
}
}

```

Para obtener más información, consulte [Eliminar una instantánea](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [DeleteSnapshot](#) de AWS CLI comandos.

## delete-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-subnet-group`.

### AWS CLI

Para eliminar un grupo de subredes

En el siguiente `delete-subnet-group` ejemplo, se elimina una subred.

```

aws memorydb delete-subnet-group \
  --subnet-group-name mysubnetgroup

```

Salida:

```

{
  "SubnetGroup": {
    "Name": "mysubnetgroup",
    "Description": "my subnet group",
    "VpcId": "vpc-86xxxx4fc",
    "Subnets": [
      {
        "Identifier": "subnet-56xxx61b",
        "AvailabilityZone": {
          "Name": "us-east-1a"
        }
      }
    ]
  }
}

```

```

    ],
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:subnetgroup/mysubnetgroup"
  }
}

```

Para obtener más información, consulte [Eliminar un grupo de subredes](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [DeleteSubnetGroup](#) comandos AWS CLI .

## delete-user

En el siguiente ejemplo de código, se muestra cómo usar `delete-user`.

### AWS CLI

Para eliminar un usuario

En el siguiente `delete-user` ejemplo, se elimina un usuario.

```

aws memorydb delete-user \
  --user-name my-user

```

Salida:

```

{
  "User": {
    "Name": "my-user",
    "Status": "deleting",
    "AccessString": "on ~app:* resetchannels -@all +@read",
    "ACLNames": [
      "my-acl"
    ],
    "MinimumEngineVersion": "6.2",
    "Authentication": {
      "Type": "password",
      "PasswordCount": 1
    },
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:user/my-user"
  }
}

```

Para obtener más información, consulte [Autenticación de usuarios con listas de control de acceso](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [DeleteUser](#) comandos AWS CLI .

## describe-acls

En el siguiente ejemplo de código, se muestra cómo usar `describe-acls`.

### AWS CLI

Para devolver una lista de ACLs

El siguiente comando `describe-acls`` devuelve una lista de ACLs

```
aws memorydb describe-acls
```

Salida:

```
{
  "ACLs": [
    {
      "Name": "open-access",
      "Status": "active",
      "UserNames": [
        "default"
      ],
      "MinimumEngineVersion": "6.2",
      "Clusters": [],
      "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:acl/open-access"
    },
    {
      "Name": "my-acl",
      "Status": "active",
      "UserNames": [],
      "MinimumEngineVersion": "6.2",
      "Clusters": [
        "my-cluster"
      ],
      "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxxx:acl/my-acl"
    }
  ]
}
```



```
}
```

Para obtener más información, consulte [Autenticación de usuarios con listas de control de acceso](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [DescribeAcls](#) comandos AWS CLI .

## describe-clusters

En el siguiente ejemplo de código, se muestra cómo usar `describe-clusters`.

### AWS CLI

Para devolver una lista de clústeres

El siguiente comando `describe-clusters`` devuelve una lista de clústeres.

```
aws memorydb describe-clusters
```

Salida:

```
{
  "Clusters": [
    {
      "Name": "my-cluster",
      "Status": "available",
      "NumberOfShards": 2,
      "ClusterEndpoint": {
        "Address": "clustercfg.my-cluster.llru6f.memorydb.us-east-1.amazonaws.com",
        "Port": 6379
      },
      "NodeType": "db.r6g.large",
      "EngineVersion": "6.2",
      "EnginePatchVersion": "6.2.6",
      "ParameterGroupName": "default.memorydb-redis6",
      "ParameterGroupStatus": "in-sync",
      "SecurityGroups": [
        {
          "SecurityGroupId": "sg-0a1434xxxxxc9fae",
          "Status": "active"
        }
      ]
    }
  ]
}
```

```

    ],
    "SubnetGroupName": "pat-sg",
    "TLSEnabled": true,
    "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:cluster/my-cluster",
    "SnapshotRetentionLimit": 0,
    "MaintenanceWindow": "wed:03:00-wed:04:00",
    "SnapshotWindow": "04:30-05:30",
    "ACLName": "my-acl",
    "AutoMinorVersionUpgrade": true
  }
]
}

```

Para obtener más información, consulte [Administración de clústeres](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [DescribeClusters](#) de AWS CLI comandos.

## describe-engine-versions

En el siguiente ejemplo de código, se muestra cómo usar `describe-engine-versions`.

### AWS CLI

Para obtener una lista de las versiones del motor

El siguiente `describe-engine-versions` devuelve una lista de versiones de motores.

```
aws memorydb describe-engine-versions
```

Salida:

```

{
  "EngineVersions": [
    {
      "EngineVersion": "6.2",
      "EnginePatchVersion": "6.2.6",
      "ParameterGroupFamily": "memorydb_redis6"
    }
  ]
}

```

Para obtener más información, consulte [Versiones y actualizaciones del motor](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [DescribeEngineVersions](#) de AWS CLI comandos.

## describe-events

En el siguiente ejemplo de código, se muestra cómo usar `describe-events`.

### AWS CLI

Para devolver una lista de eventos

El siguiente comando `describe-events` devuelve una lista de eventos.

```
aws memorydb describe-events
```

Salida:

```
{
  "Events": [
    {
      "SourceName": "my-cluster",
      "SourceType": "cluster",
      "Message": "Increase replica count started for replication group my-cluster on 2022-07-22T14:09:01.440Z",
      "Date": "2022-07-22T07:09:01.443000-07:00"
    },
    {
      "SourceName": "my-user",
      "SourceType": "user",
      "Message": "Create user my-user operation completed.",
      "Date": "2022-07-22T07:00:02.975000-07:00"
    }
  ]
}
```

Para obtener más información, consulte [Supervisión de eventos](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [DescribeEvents](#) de AWS CLI comandos.

## describe-parameter-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-parameter-groups`.

### AWS CLI

Para devolver una lista de grupos de parámetros

El siguiente `describe-parameter-groups` devuelve una lista de grupos de parámetros.

```
aws memorydb describe-parameter-groups
```

Salida:

```
{
  "ParameterGroups": [
    {
      "Name": "default.memorydb-redis6",
      "Family": "memorydb_redis6",
      "Description": "Default parameter group for memorydb_redis6",
      "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:parametergroup/default.memorydb-redis6"
    }
  ]
}
```

Para obtener más información, consulte [Configuración de los parámetros del motor mediante grupos de parámetros](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [DescribeParameterGroups](#) de AWS CLI comandos.

## describe-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-parameters`.

### AWS CLI

Para devolver una lista de parámetros

El siguiente comando `describe-parameters` devuelve una lista de parámetros.

**aws memorydb describe-parameters**

Salida:

```
{
  "Parameters": [
    {
      "Name": "acllog-max-len",
      "Value": "128",
      "Description": "The maximum length of the ACL Log",
      "DataType": "integer",
      "AllowedValues": "1-10000",
      "MinimumEngineVersion": "6.2.4"
    },
    {
      "Name": "activedefrag",
      "Value": "no",
      "Description": "Enabled active memory defragmentation",
      "DataType": "string",
      "AllowedValues": "yes,no",
      "MinimumEngineVersion": "6.2.4"
    },
    {
      "Name": "active-defrag-cycle-max",
      "Value": "75",
      "Description": "Maximal effort for defrag in CPU percentage",
      "DataType": "integer",
      "AllowedValues": "1-75",
      "MinimumEngineVersion": "6.2.4"
    },
    {
      "Name": "active-defrag-cycle-min",
      "Value": "5",
      "Description": "Minimal effort for defrag in CPU percentage",
      "DataType": "integer",
      "AllowedValues": "1-75",
      "MinimumEngineVersion": "6.2.4"
    },
    {
      "Name": "active-defrag-ignore-bytes",
      "Value": "104857600",
      "Description": "Minimum amount of fragmentation waste to start active defrag",

```

```
    "DataType": "integer",
    "AllowedValues": "1048576-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "active-defrag-max-scan-fields",
    "Value": "1000",
    "Description": "Maximum number of set/hash/zset/list fields that will be
processed from the main dictionary scan",
    "DataType": "integer",
    "AllowedValues": "1-1000000",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "active-defrag-threshold-lower",
    "Value": "10",
    "Description": "Minimum percentage of fragmentation to start active
defrag",
    "DataType": "integer",
    "AllowedValues": "1-100",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "active-defrag-threshold-upper",
    "Value": "100",
    "Description": "Maximum percentage of fragmentation at which we use
maximum effort",
    "DataType": "integer",
    "AllowedValues": "1-100",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "active-expire-effort",
    "Value": "1",
    "Description": "The amount of effort that redis uses to expire items in
the active expiration job",
    "DataType": "integer",
    "AllowedValues": "1-10",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "activeresharding",
    "Value": "yes",
    "Description": "Apply resharding or not",
```

```
    "DataType": "string",
    "AllowedValues": "yes,no",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "client-output-buffer-limit-normal-hard-limit",
    "Value": "0",
    "Description": "Normal client output buffer hard limit in bytes",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "client-output-buffer-limit-normal-soft-limit",
    "Value": "0",
    "Description": "Normal client output buffer soft limit in bytes",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "client-output-buffer-limit-normal-soft-seconds",
    "Value": "0",
    "Description": "Normal client output buffer soft limit in seconds",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "client-output-buffer-limit-pubsub-hard-limit",
    "Value": "33554432",
    "Description": "Pubsub client output buffer hard limit in bytes",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "client-output-buffer-limit-pubsub-soft-limit",
    "Value": "8388608",
    "Description": "Pubsub client output buffer soft limit in bytes",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
},
```

```
{
  "Name": "client-output-buffer-limit-pubsub-soft-seconds",
  "Value": "60",
  "Description": "Pubsub client output buffer soft limit in seconds",
  "DataType": "integer",
  "AllowedValues": "0-",
  "MinimumEngineVersion": "6.2.4"
},
{
  "Name": "hash-max-ziplist-entries",
  "Value": "512",
  "Description": "The maximum number of hash entries in order for the
dataset to be compressed",
  "DataType": "integer",
  "AllowedValues": "0-",
  "MinimumEngineVersion": "6.2.4"
},
{
  "Name": "hash-max-ziplist-value",
  "Value": "64",
  "Description": "The threshold of biggest hash entries in order for the
dataset to be compressed",
  "DataType": "integer",
  "AllowedValues": "0-",
  "MinimumEngineVersion": "6.2.4"
},
{
  "Name": "hll-sparse-max-bytes",
  "Value": "3000",
  "Description": "HyperLogLog sparse representation bytes limit",
  "DataType": "integer",
  "AllowedValues": "1-16000",
  "MinimumEngineVersion": "6.2.4"
},
{
  "Name": "lazyfree-lazy-eviction",
  "Value": "no",
  "Description": "Perform an asynchronous delete on evictions",
  "DataType": "string",
  "AllowedValues": "yes,no",
  "MinimumEngineVersion": "6.2.4"
},
{
  "Name": "lazyfree-lazy-expire",
```



```

    "Value": "no",
    "Description": "Perform an asynchronous delete on expired keys",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "lazyfree-lazy-server-del",
    "Value": "no",
    "Description": "Perform an asynchronous delete on key updates",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "lazyfree-lazy-user-del",
    "Value": "no",
    "Description": "Specifies whether the default behavior of DEL command
acts the same as UNLINK",
    "DataType": "string",
    "AllowedValues": "yes,no",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "lfu-decay-time",
    "Value": "1",
    "Description": "The amount of time in minutes to decrement the key
counter for LFU eviction policy",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "lfu-log-factor",
    "Value": "10",
    "Description": "The log factor for incrementing key counter for LFU
eviction policy",
    "DataType": "integer",
    "AllowedValues": "1-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "list-compress-depth",
    "Value": "0",

```

```
    "Description": "Number of quicklist ziplist nodes from each side of
the list to exclude from compression. The head and tail of the list are always
uncompressed for fast push/pop operations",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "maxmemory-policy",
    "Value": "noeviction",
    "Description": "Max memory policy",
    "DataType": "string",
    "AllowedValues": "volatile-lru,allkeys-lru,volatile-lfu,allkeys-
lfu,volatile-random,allkeys-random,volatile-ttl,noeviction",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "maxmemory-samples",
    "Value": "3",
    "Description": "Max memory samples",
    "DataType": "integer",
    "AllowedValues": "1-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "notify-keyspace-events",
    "Description": "The keyspace events for Redis to notify Pub/Sub clients
about. By default all notifications are disabled",
    "DataType": "string",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "set-max-intset-entries",
    "Value": "512",
    "Description": "The limit in the size of the set in order for the
dataset to be compressed",
    "DataType": "integer",
    "AllowedValues": "0-",
    "MinimumEngineVersion": "6.2.4"
  },
  {
    "Name": "slowlog-log-slower-than",
    "Value": "10000",
```

```

        "Description": "The execution time, in microseconds, to exceed in order
for the command to get logged. Note that a negative number disables the slow log,
while a value of zero forces the logging of every command",
        "DataType": "integer",
        "AllowedValues": "-",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "slowlog-max-len",
        "Value": "128",
        "Description": "The length of the slow log. There is no limit to this
length. Just be aware that it will consume memory. You can reclaim memory used by
the slow log with SLOWLOG RESET.",
        "DataType": "integer",
        "AllowedValues": "0-",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "stream-node-max-bytes",
        "Value": "4096",
        "Description": "The maximum size of a single node in a stream in bytes",
        "DataType": "integer",
        "AllowedValues": "0-",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "stream-node-max-entries",
        "Value": "100",
        "Description": "The maximum number of items a single node in a stream
can contain",
        "DataType": "integer",
        "AllowedValues": "0-",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "tcp-keepalive",
        "Value": "300",
        "Description": "If non-zero, send ACKs every given number of seconds",
        "DataType": "integer",
        "AllowedValues": "0-",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "timeout",

```

```

        "Value": "0",
        "Description": "Close connection if client is idle for a given number of
seconds, or never if 0",
        "DataType": "integer",
        "AllowedValues": "0,20-",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "tracking-table-max-keys",
        "Value": "1000000",
        "Description": "The maximum number of keys allowed for the tracking
table for client side caching",
        "DataType": "integer",
        "AllowedValues": "1-1000000000",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "zset-max-ziplist-entries",
        "Value": "128",
        "Description": "The maximum number of sorted set entries in order for
the dataset to be compressed",
        "DataType": "integer",
        "AllowedValues": "0-",
        "MinimumEngineVersion": "6.2.4"
    },
    {
        "Name": "zset-max-ziplist-value",
        "Value": "64",
        "Description": "The threshold of biggest sorted set entries in order for
the dataset to be compressed",
        "DataType": "integer",
        "AllowedValues": "0-",
        "MinimumEngineVersion": "6.2.4"
    }
]
}

```

Para obtener más información, consulte [Configuración de los parámetros del motor mediante grupos de parámetros en la Guía del usuario de MemoryDB](#).

- Para API obtener más información, consulte la Referencia [DescribeParameters](#) de AWS CLI comandos.

## describe-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `describe-snapshots`.

### AWS CLI

Para devolver una lista de instantáneas

La siguiente descripción de `snapshots` devuelve una lista de instantáneas.

```
aws memorydb describe-snapshots
```

Salida:

```
{
  "Snapshots": [
    {
      "Name": "my-cluster-snapshot",
      "Status": "available",
      "Source": "manual",
      "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx2:snapshot/my-cluster-snapshot",
      "ClusterConfiguration": {
        "Name": "my-cluster",
        "Description": " ",
        "NodeType": "db.r6g.large",
        "EngineVersion": "6.2",
        "MaintenanceWindow": "wed:03:00-wed:04:00",
        "Port": 6379,
        "ParameterGroupName": "default.memorydb-redis6",
        "SubnetGroupName": "my-sg",
        "VpcId": "vpc-862574fc",
        "SnapshotRetentionLimit": 0,
        "SnapshotWindow": "04:30-05:30",
        "NumShards": 2
      }
    }
  ]
}
```

Para obtener más información, consulte [Instantánea y restauración](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [DescribeSnapshots](#) de AWS CLI comandos.

## describe-subnet-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-subnet-groups`.

### AWS CLI

Para devolver una lista de grupos de subredes

El siguiente `describe-subnet-groups` devuelve una lista de grupos de subredes.

```
aws memorydb describe-subnet-groups
```

### Salida

```
{
  "SubnetGroups": [
    {
      "Name": "my-sg",
      "Description": "pat-sg",
      "VpcId": "vpc-86xxx4fc",
      "Subnets": [
        {
          "Identifier": "subnet-faxx84a6",
          "AvailabilityZone": {
            "Name": "us-east-1b"
          }
        },
        {
          "Identifier": "subnet-56xxf61b",
          "AvailabilityZone": {
            "Name": "us-east-1a"
          }
        }
      ],
      "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:subnetgroup/my-sg"
    }
  ]
}
```

Para obtener más información, consulte [Subredes y grupos de subredes](#) en la Guía del usuario de MemoryDB.

- Para obtener API más información, consulte la Referencia de comandos [DescribeSubnetGroups](#).AWS CLI

## describe-users

En el siguiente ejemplo de código, se muestra cómo usar describe-users.

### AWS CLI

Para devolver una lista de usuarios

El siguiente comando describe-users` devuelve una lista de usuarios.

```
aws memorydb describe-users
```

### Salida

```
{
  "Users": [
    {
      "Name": "default",
      "Status": "active",
      "AccessString": "on ~* &* +@all",
      "ACLNames": [
        "open-access"
      ],
      "MinimumEngineVersion": "6.0",
      "Authentication": {
        "Type": "no-password"
      },
      "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:user/default"
    },
    {
      "Name": "my-user",
      "Status": "active",
      "AccessString": "off ~objects:* ~items:* ~public:* resetchannels -@all",
      "ACLNames": [],
      "MinimumEngineVersion": "6.2",
      "Authentication": {
        "Type": "password",

```

```

        "PasswordCount": 2
      },
      "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:user/my-user"
    }
  ]
}

```

Para obtener más información, consulte [Autenticación de usuarios con listas de control de acceso](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [DescribeUsers](#) comandos AWS CLI .

## failover-shard

En el siguiente ejemplo de código, se muestra cómo usar `failover-shard`.

### AWS CLI

Para conmutar por error un fragmento

El siguiente fragmento de conmutación por error conmuta por error un fragmento.

```

aws memorydb failover-shard \
  --cluster-name my-cluster --shard-name 0001

```

Salida:

```

{
  "Cluster": {
    "Name": "my-cluster",
    "Status": "available",
    "NumberOfShards": 2,
    "ClusterEndpoint": {
      "Address": "clustercfg.my-cluster.xxxxxx.memorydb.us-east-1.amazonaws.com",
      "Port": 6379
    },
    "NodeType": "db.r6g.large",
    "EngineVersion": "6.2",
    "EnginePatchVersion": "6.2.6",
    "ParameterGroupName": "default.memorydb-redis6",

```



```

    "ParameterGroupStatus": "in-sync",
    "SecurityGroups": [
      {
        "SecurityGroupId": "sg-0a143xxxx45c9fae",
        "Status": "active"
      }
    ],
    "SubnetGroupName": "my-sg",
    "TLSEnabled": true,
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster",
    "SnapshotRetentionLimit": 0,
    "MaintenanceWindow": "wed:03:00-wed:04:00",
    "SnapshotWindow": "04:30-05:30",
    "AutoMinorVersionUpgrade": true
  }
}

```

Para obtener más información, consulte [Minimizar el tiempo de inactividad con MultiAZ](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [FailoverShard](#) comandos AWS CLI .

## list-allowed-node-type-updates

En el siguiente ejemplo de código, se muestra cómo usar `list-allowed-node-type-updates`.

AWS CLI

Para devolver una lista de las actualizaciones de tipo de nodo permitidas

El siguiente comando `list-allowed-node-type-updates` devuelve una lista de las actualizaciones de tipo de nodo disponibles.

```
aws memorydb list-allowed-node-type-updates
```

Salida:

```

{
  "Cluster": {
    "Name": "my-cluster",
    "Status": "available",

```

```
    "NumberOfShards": 2,
    "ClusterEndpoint": {
      "Address": "clustercfg.my-cluster.xxxxxx.memorydb.us-
east-1.amazonaws.com",
      "Port": 6379
    },
    "NodeType": "db.r6g.large",
    "EngineVersion": "6.2",
    "EnginePatchVersion": "6.2.6",
    "ParameterGroupName": "default.memorydb-redis6",
    "ParameterGroupStatus": "in-sync",
    "SecurityGroups": [
      {
        "SecurityGroupId": "sg-0a143xxxx45c9fae",
        "Status": "active"
      }
    ],
    "SubnetGroupName": "my-sg",
    "TLSEnabled": true,
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster",
    "SnapshotRetentionLimit": 0,
    "MaintenanceWindow": "wed:03:00-wed:04:00",
    "SnapshotWindow": "04:30-05:30",
    "AutoMinorVersionUpgrade": true
  }
}
```

Para obtener más información, consulte [Escalado](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [ListAllowedNodeTypeUpdates](#) de AWS CLI comandos.

## list-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-tags`.

### AWS CLI

Para devolver una lista de etiquetas

La siguiente lista de etiquetas devuelve una lista de etiquetas.

```
aws memorydb list-tags \
```

```
--resource-arn arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster
```

Salida:

```
{
  "TagList": [
    {
      "Key": "mytag",
      "Value": "myvalue"
    }
  ]
}
```

Para obtener más información, consulte [Etiquetado de recursos](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [ListTags](#) comandos AWS CLI .

## reset-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `reset-parameter-group`.

AWS CLI

Para restablecer un grupo de parámetros

Lo siguiente `reset-parameter-group` restablece un grupo de parámetros.

```
aws memorydb reset-parameter-group \
  --parameter-group-name my-parameter-group \
  --all-parameters
```

Salida:

```
{
  "ParameterGroup": {
    "Name": "my-parameter-group",
    "Family": "memorydb_redis6",
    "Description": "my parameter group",
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:parametergroup/my-parameter-group"
  }
}
```

```
}
```

Para obtener más información, consulte [Configuración de los parámetros del motor mediante grupos de parámetros](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [ResetParameterGroup](#) de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar un recurso

La siguiente etiqueta, recurso, añade una etiqueta a un recurso.

```
aws memorydb tag-resource \  
  --resource-arn arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster \  
  --tags Key="mykey",Value="myvalue"
```

Salida:

```
{  
  "TagList": [  
    {  
      "Key": "mytag",  
      "Value": "myvalue"  
    },  
    {  
      "Key": "mykey",  
      "Value": "myvalue"  
    }  
  ]  
}
```

Para obtener más información, consulte [Etiquetado de recursos](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [TagResource](#) comandos AWS CLI .

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para actualizar un ACL

La siguiente actualización: `acl` actualiza y añade un usuarioACL.

```
aws memorydb untag-resource \  
  --resource-arn arn:aws:memorydb:us-east-1:491658xxxxx:cluster/my-cluster \  
  --tag-keys mykey
```

Salida:

```
{  
  "TagList": [  
    {  
      "Key": "mytag",  
      "Value": "myvalue"  
    }  
  ]  
}
```

Para obtener más información, consulte [Etiquetado de recursos](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [UntagResource](#) comandos AWS CLI .

## update-cluster

En el siguiente ejemplo de código, se muestra cómo usar `update-cluster`.

### AWS CLI

Para actualizar un clúster

El siguiente `update-cluster` actualiza el grupo de parámetros de un clúster a `my-parameter-group`

```
aws memorydb update-cluster \  
  --parameter-group my-parameter-group
```

```
--cluster-name my-cluster \  
--parameter-group-name my-parameter-group
```

Salida:

```
{  
  "Cluster": {  
    "Name": "my-cluster",  
    "Status": "available",  
    "NumberOfShards": 2,  
    "AvailabilityMode": "MultiAZ",  
    "ClusterEndpoint": {  
      "Address": "clustercfg.my-cluster.llru6f.memorydb.us-  
east-1.amazonaws.com",  
      "Port": 6379  
    },  
    "NodeType": "db.r6g.large",  
    "EngineVersion": "6.2",  
    "EnginePatchVersion": "6.2.6",  
    "ParameterGroupName": "my-parameter-group",  
    "ParameterGroupStatus": "in-sync",  
    "SecurityGroups": [  
      {  
        "SecurityGroupId": "sg-0a143xxxxxc9fae",  
        "Status": "active"  
      }  
    ],  
    "SubnetGroupName": "pat-sg",  
    "TLSEnabled": true,  
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster",  
    "SnapshotRetentionLimit": 0,  
    "MaintenanceWindow": "wed:03:00-wed:04:00",  
    "SnapshotWindow": "04:30-05:30",  
    "ACLName": "my-acl",  
    "AutoMinorVersionUpgrade": true  
  }  
}
```

Para obtener más información, consulte [Modificación de un clúster](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [UpdateCluster](#) de AWS CLI comandos.

## update-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `update-parameter-group`.

### AWS CLI

Para actualizar un grupo de parámetros

Lo siguiente `update-parameter-group` actualiza un grupo de parámetros.

```
aws memorydb update-parameter-group \  
  --parameter-group-name my-parameter-group \  
  --parameter-name-values "ParameterName=activedefrag, ParameterValue=no"
```

Salida:

```
{  
  "ParameterGroup": {  
    "Name": "my-parameter-group",  
    "Family": "memorydb_redis6",  
    "Description": "my parameter group",  
    "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:parametergroup/my-parameter-  
group"  
  }  
}
```

Para obtener más información, consulte [Modificación de un grupo de parámetros](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia [UpdateParameterGroup](#) de AWS CLI comandos.

## update-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `update-subnet-group`.

### AWS CLI

Para actualizar un grupo de subredes

El siguiente `update-subnet-group` actualiza el ID de subred de un grupo de subredes.

```
aws memorydb update-subnet-group \  
  --subnet-group-name my-sg \  
  --subnet-ids subnet-01f29d458f3xxxxx
```

Salida:

```
{  
  "SubnetGroup": {  
    "Name": "my-sg-1",  
    "Description": "my-sg",  
    "VpcId": "vpc-09d2cfc01xxxxxxx",  
    "Subnets": [  
      {  
        "Identifier": "subnet-01f29d458fxxxxxx",  
        "AvailabilityZone": {  
          "Name": "us-east-1a"  
        }  
      }  
    ],  
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:subnetgroup/my-sg"  
  }  
}
```

Para obtener más información, consulte [Subredes y grupos de subredes en la Guía del usuario de MemoryDB](#).

- Para obtener API más información, consulte la Referencia de comandos [UpdateSubnetGroup](#).AWS CLI

## update-user

En el siguiente ejemplo de código, se muestra cómo usar `update-user`.

AWS CLI

Para actualizar un usuario

Lo siguiente `update-user` modifica la cadena de acceso de un usuario.

```
aws memorydb update-user \  
  --user-name my-user \  
  --password my-password
```



```
--access-string "off ~objects:* ~items:* ~public:* resetchannels -@all"
```

Salida:

```
{
  "User": {
    "Name": "my-user",
    "Status": "modifying",
    "AccessString": "off ~objects:* ~items:* ~public:* resetchannels -@all",
    "ACLNames": [
      "myt-acl"
    ],
    "MinimumEngineVersion": "6.2",
    "Authentication": {
      "Type": "password",
      "PasswordCount": 2
    },
    "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:user/my-user"
  }
}
```

Para obtener más información, consulte [Autenticar usuarios con listas de control de acceso](#) en la Guía del usuario de MemoryDB.

- Para API obtener más información, consulte la Referencia de [UpdateUser](#) comandos AWS CLI .

## MSKEjemplos de Amazon que utilizan AWS CLI

En los siguientes ejemplos de código, se muestra cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con AmazonMSK.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **create-cluster**

En el siguiente ejemplo de código, se muestra cómo usar `create-cluster`.

#### AWS CLI

Para crear un MSK clúster de Amazon

En el siguiente `create-cluster` ejemplo, se crea un MSK clúster denominado `MessagingCluster` con tres nodos de intermediario. Un JSON archivo denominado `brokernodegroupinfo.json` especifica las tres subredes en las que desea que Amazon distribuya MSK los nodos de los corredores. En este ejemplo no se especifica el nivel de supervisión, por lo que el clúster obtiene el `DEFAULT` nivel.

```
aws kafka create-cluster \  
  --cluster-name "MessagingCluster" \  
  --broker-node-group-info file://brokernodegroupinfo.json \  
  --kafka-version "2.2.1" \  
  --number-of-broker-nodes 3
```

Contenidos de `brokernodegroupinfo.json`:

```
{  
  "InstanceType": "kafka.m5.xlarge",  
  "BrokerAZDistribution": "DEFAULT",  
  "ClientSubnets": [  
    "subnet-0123456789111abcd",  
    "subnet-0123456789222abcd",  
    "subnet-0123456789333abcd"  
  ]  
}
```

Salida:

```
{  
  "ClusterArn": "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",  
  "ClusterName": "MessagingCluster",  
  "State": "CREATING"  
}
```

Para obtener más información, consulte [Creación de un MSK clúster de Amazon](#) en Amazon Managed Streaming for Apache Kafka.

- Para API obtener más información, consulte [CreateCluster](#) la Referencia de AWS CLI comandos.

## create-configuration

En el siguiente ejemplo de código, se muestra cómo usar create-configuration.

### AWS CLI

Para crear una MSK configuración de Amazon personalizada

El siguiente create-configuration ejemplo crea una MSK configuración personalizada con las propiedades del servidor que se especifican en el archivo de entrada.

```
aws kafka create-configuration \  
  --name "CustomConfiguration" \  
  --description "Topic autocreation enabled; Apache ZooKeeper timeout 2000 ms; Log rolling 604800000 ms." \  
  --kafka-versions "2.2.1" \  
  --server-properties file://configuration.txt
```

Contenidos de configuration.txt:

```
auto.create.topics.enable = true  
zookeeper.connection.timeout.ms = 2000  
log.roll.ms = 604800000
```

Este comando no genera ninguna salida. Salida:

```
{  
  "Arn": "arn:aws:kafka:us-west-2:123456789012:configuration/CustomConfiguration/  
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",  
  "CreationTime": "2019-10-09T15:26:05.548Z",  
  "LatestRevision":  
    {  
      "CreationTime": "2019-10-09T15:26:05.548Z",  
      "Description": "Topic autocreation enabled; Apache ZooKeeper timeout  
2000 ms; Log rolling 604800000 ms.",  
      "Revision": 1
```

```
    },  
    "Name": "CustomConfiguration"  
  }  
}
```

Para obtener más información, consulte [Amazon MSK Configuration Operations](#) en la Guía para desarrolladores de Amazon Managed Streaming for Apache Kafka.

- Para API obtener más información, consulte [CreateConfiguration](#) la Referencia de AWS CLI comandos.

## describe-cluster

En el siguiente ejemplo de código, se muestra cómo usar `describe-cluster`.

### AWS CLI

Para describir un clúster

El siguiente `describe-cluster` ejemplo describe un MSK clúster de Amazon.

```
aws kafka describe-cluster \  
  --cluster-arn arn:aws:kafka:us-east-1:123456789012:cluster/demo-  
cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5
```

Salida:

```
{  
  "ClusterInfo": {  
    "BrokerNodeGroupInfo": {  
      "BrokerAZDistribution": "DEFAULT",  
      "ClientSubnets": [  
        "subnet-cbfff283",  
        "subnet-6746046b"  
      ],  
      "InstanceType": "kafka.m5.large",  
      "SecurityGroups": [  
        "sg-f839b688"  
      ],  
      "StorageInfo": {  
        "EbsStorageInfo": {  
          "VolumeSize": 100  
        }  
      }  
    }  
  }  
}
```

```

    }
  },
  "ClusterArn": "arn:aws:kafka:us-east-1:123456789012:cluster/demo-
cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5",
  "ClusterName": "demo-cluster-1",
  "CreationTime": "2020-07-09T02:31:36.223000+00:00",
  "CurrentBrokerSoftwareInfo": {
    "KafkaVersion": "2.2.1"
  },
  "CurrentVersion": "K3AEGXETSR30VB",
  "EncryptionInfo": {
    "EncryptionAtRest": {
      "DataVolumeKMSKeyId": "arn:aws:kms:us-east-1:123456789012:key/
a7ca56d5-0768-4b64-a670-339a9fbef81c"
    },
    "EncryptionInTransit": {
      "ClientBroker": "TLS_PLAINTEXT",
      "InCluster": true
    }
  },
  "EnhancedMonitoring": "DEFAULT",
  "OpenMonitoring": {
    "Prometheus": {
      "JmxExporter": {
        "EnabledInBroker": false
      },
      "NodeExporter": {
        "EnabledInBroker": false
      }
    }
  },
  "NumberOfBrokerNodes": 2,
  "State": "ACTIVE",
  "Tags": {},
  "ZookeeperConnectionString": "z-2.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181,z-1.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181,z-3.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181"
}
}

```

Para obtener más información, consulte la [lista de MSK clústeres de Amazon](#) en la Guía para desarrolladores de Amazon Managed Streaming for Apache Kafka.

- Para API obtener más información, consulte [DescribeCluster](#) la Referencia de AWS CLI comandos.

## get-bootstrap-brokers

En el siguiente ejemplo de código, se muestra cómo usar `get-bootstrap-brokers`.

### AWS CLI

Para obtener corredores de bootstrap

En el siguiente `get-bootstrap-brokers` ejemplo, se recupera la información del agente de arranque de un clúster de AmazonMSK.

```
aws kafka get-bootstrap-brokers \
  --cluster-arn arn:aws:kafka:us-east-1:123456789012:cluster/demo-
  cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5
```

Salida:

```
{
  "BootstrapBrokerString": "b-1.demo-cluster-1.xuy0sb.c5.kafka.us-
  east-1.amazonaws.com:9092,b-2.demo-cluster-1.xuy0sb.c5.kafka.us-
  east-1.amazonaws.com:9092",
  "BootstrapBrokerStringTls": "b-1.demo-cluster-1.xuy0sb.c5.kafka.us-
  east-1.amazonaws.com:9094,b-2.demo-cluster-1.xuy0sb.c5.kafka.us-
  east-1.amazonaws.com:9094"
}
```

Para obtener más información, consulte [Cómo obtener los agentes de Bootstrap](#) en la Guía para desarrolladores de Amazon Managed Streaming for Apache Kafka.

- Para API obtener más información, consulte la Referencia [GetBootstrapBrokers](#) de AWS CLI comandos.

## list-clusters

En el siguiente ejemplo de código, se muestra cómo usar `list-clusters`.

### AWS CLI

Para ver una lista de los clústeres disponibles

En el siguiente `list-clusters` ejemplo, se muestran los MSK clústeres de Amazon de tu AWS cuenta.

```
aws kafka list-clusters
```

Salida:

```
{
  "ClusterInfoList": [
    {
      "BrokerNodeGroupInfo": {
        "BrokerAZDistribution": "DEFAULT",
        "ClientSubnets": [
          "subnet-cbfff283",
          "subnet-6746046b"
        ],
        "InstanceType": "kafka.m5.large",
        "SecurityGroups": [
          "sg-f839b688"
        ],
        "StorageInfo": {
          "EbsStorageInfo": {
            "VolumeSize": 100
          }
        }
      },
      "ClusterArn": "arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5",
      "ClusterName": "demo-cluster-1",
      "CreationTime": "2020-07-09T02:31:36.223000+00:00",
      "CurrentBrokerSoftwareInfo": {
        "KafkaVersion": "2.2.1"
      },
      "CurrentVersion": "K3AEGXETSR30VB",
      "EncryptionInfo": {
        "EncryptionAtRest": {
          "DataVolumeKMSKeyId": "arn:aws:kms:us-east-1:123456789012:key/a7ca56d5-0768-4b64-a670-339a9fbef81c"
        },
        "EncryptionInTransit": {
          "ClientBroker": "TLS_PLAINTEXT",
          "InCluster": true
        }
      }
    }
  ]
}
```

```

    },
    "EnhancedMonitoring": "DEFAULT",
    "OpenMonitoring": {
      "Prometheus": {
        "JmxExporter": {
          "EnabledInBroker": false
        },
        "NodeExporter": {
          "EnabledInBroker": false
        }
      }
    },
    "NumberOfBrokerNodes": 2,
    "State": "ACTIVE",
    "Tags": {},
    "ZookeeperConnectString": "z-2.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181,z-1.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181,z-3.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181"
  }
]
}

```

Para obtener más información, consulte la [lista de MSK clústeres de Amazon](#) en la Guía para desarrolladores de Amazon Managed Streaming for Apache Kafka.

- Para API obtener más información, consulte [ListClusters](#) la Referencia de AWS CLI comandos.

## update-broker-storage

En el siguiente ejemplo de código, se muestra cómo usar `update-broker-storage`.

### AWS CLI

Para actualizar el EBS almacenamiento para los corredores

En el siguiente `update-broker-storage` ejemplo, se actualiza la cantidad de EBS almacenamiento de todos los corredores del clúster. Amazon MSK establece la cantidad de almacenamiento objetivo para cada agente en la cantidad especificada en el ejemplo. Puede obtener la versión actual del clúster describiéndolo o enumerando todos los clústeres.

```
aws kafka update-broker-storage \
```



```
--cluster-arn "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2" \
--current-version "K21V3IB1VIZYYH" \
--target-broker-ebs-volume-info "KafkaBrokerNodeId=ALL,VolumeSizeGB=1100"
```

El resultado devuelve un ARN para esta `update-broker-storage` operación. Para determinar si la operación se ha completado, utilice el `describe-cluster-operation` comando que contiene esta información ARN como entrada.

```
{
  "ClusterArn": "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",
  "ClusterOperationArn": "arn:aws:kafka:us-west-2:123456789012:cluster-
operation/V123450123/a1b2c3d4-1234-abcd-cdef-22222EXAMPLE-2/a1b2c3d4-abcd-1234-
bcde-33333EXAMPLE"
}
```

Para obtener más información, consulte [Actualización del EBS almacenamiento para intermediarios](#) en la Guía para desarrolladores de Amazon Managed Streaming for Apache Kafka.

- Para API obtener más información, consulte [UpdateBrokerStorage](#) la Referencia de AWS CLI comandos.

## update-cluster-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-cluster-configuration`.

### AWS CLI

Para actualizar la configuración de un MSK clúster de Amazon

El siguiente `update-cluster-configuration` ejemplo actualiza la configuración del MSK clúster existente especificado. Utiliza una MSK configuración personalizada.

```
aws kafka update-cluster-configuration \
  --cluster-arn "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2" \
  --configuration-info file://configuration-info.json \
  --current-version "K21V3IB1VIZYYH"
```

Contenidos de `configuration-info.json`:

```
{
  "Arn": "arn:aws:kafka:us-west-2:123456789012:configuration/CustomConfiguration/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",
  "Revision": 1
}
```

El resultado devuelve un ARN para esta `update-cluster-configuration` operación. Para determinar si la operación se ha completado, utilice el `describe-cluster-operation` comando que contiene esta información ARN como entrada.

```
{
  "ClusterArn": "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",
  "ClusterOperationArn": "arn:aws:kafka:us-west-2:123456789012:cluster-
operation/V123450123/a1b2c3d4-1234-abcd-cdef-22222EXAMPLE-2/a1b2c3d4-abcd-1234-
bcde-33333EXAMPLE"
}
```

Para obtener más información, consulte [Actualización de la configuración de un MSK clúster de Amazon](#) en la Guía para desarrolladores de Amazon Managed Streaming for Apache Kafka.

- Para API obtener más información, consulte [UpdateClusterConfiguration](#) la Referencia de AWS CLI comandos.

## Ejemplos de Network Manager que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante el uso AWS Command Line Interface de Network Manager.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **associate-customer-gateway**

En el siguiente ejemplo de código, se muestra cómo usar `associate-customer-gateway`.

#### AWS CLI

Para asociar una pasarela de clientes

El siguiente `associate-customer-gateway` ejemplo asocia la pasarela de cliente `cgw-11223344556677889` de la red global especificada con el dispositivo `device-07f6fd08867abc123`.

```
aws networkmanager associate-customer-gateway \  
  --customer-gateway-arn arn:aws:ec2:us-west-2:123456789012:customer-gateway/  
cgw-11223344556677889 \  
  --global-network-id global-network-01231231231231231 \  
  --device-id device-07f6fd08867abc123 \  
  --region us-west-2
```

Salida:

```
{  
  "CustomerGatewayAssociation": {  
    "CustomerGatewayArn": "arn:aws:ec2:us-west-2:123456789012:customer-gateway/  
cgw-11223344556677889",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "DeviceId": "device-07f6fd08867abc123",  
    "State": "PENDING"  
  }  
}
```

Para obtener más información, consulte [Customer Gateway Associations](#) en la Guía del administrador de redes de Transit Gateway.

- Para API obtener más información, consulte [AssociateCustomerGateway](#) la Referencia de AWS CLI comandos.

### **associate-link**

En el siguiente ejemplo de código, se muestra cómo usar `associate-link`.

## AWS CLI

Para asociar un enlace

El siguiente `associate-link` ejemplo asocia el enlace `link-11112222aaaabbbb1` con el dispositivo `device-07f6fd08867abc123`. El enlace y el dispositivo se encuentran en la red global especificada.

```
aws networkmanager associate-link \  
  --global-network-id global-network-01231231231231231 \  
  --device-id device-07f6fd08867abc123 \  
  --link-id link-11112222aaaabbbb1 \  
  --region us-west-2
```

Salida:

```
{  
  "LinkAssociation": {  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "DeviceId": "device-07f6fd08867abc123",  
    "LinkId": "link-11112222aaaabbbb1",  
    "LinkAssociationState": "PENDING"  
  }  
}
```

Para obtener más información, consulte [Asociaciones de dispositivos y enlaces](#) en la Guía del administrador de red de Transit Gateway.

- Para API obtener más información, consulte [AssociateLink](#) la Referencia de AWS CLI comandos.

## **create-core-network**

En el siguiente ejemplo de código, se muestra cómo usar `create-core-network`.

## AWS CLI

Para crear una red principal

En el siguiente `create-core-network` ejemplo, se crea una red principal con una descripción y etiquetas opcionales dentro de una red WAN global de AWS nube.

```
aws networkmanager create-core-network \  
  --global-network-id global-network-0d59060f16a73bc41 \  
  --description "Main headquarters location" \  
  --tags Key=Name,Value="New York City office"
```

Salida:

```
{  
  "CoreNetwork": {  
    "GlobalNetworkId": "global-network-0d59060f16a73bc41",  
    "CoreNetworkId": "core-network-0fab62fe438d94db6",  
    "CoreNetworkArn": "arn:aws:networkmanager::987654321012:core-network/core-  
network-0fab62fe438d94db6",  
    "Description": "Main headquarters location",  
    "CreatedAt": "2022-01-10T19:53:59+00:00",  
    "State": "AVAILABLE",  
    "Tags": [  
      {  
        "Key": "Name",  
        "Value": "New York City office"  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte la Guía del WAN usuario sobre [las redes principales](#) en la AWS nube.

- Para API obtener más información, consulte [CreateCoreNetwork](#) la Referencia de AWS CLI comandos.

## create-device

En el siguiente ejemplo de código, se muestra cómo usar create-device.

### AWS CLI

Para crear un dispositivo

El siguiente create-device ejemplo crea un dispositivo en la red global especificada. Los detalles del dispositivo incluyen una descripción, el tipo, el proveedor, el modelo y el número de serie.

```
aws networkmanager create-device
--global-network-id global-network-01231231231231231 \
--description "New York office device" \
--type "office device" \
--vendor "anycompany" \
--model "abcabc" \
--serial-number "1234" \
--region us-west-2
```

Salida:

```
{
  "Device": {
    "DeviceId": "device-07f6fd08867abc123",
    "DeviceArn": "arn:aws:networkmanager::123456789012:device/global-
network-01231231231231231/device-07f6fd08867abc123",
    "GlobalNetworkId": "global-network-01231231231231231",
    "Description": "New York office device",
    "Type": "office device",
    "Vendor": "anycompany",
    "Model": "abcabc",
    "SerialNumber": "1234",
    "CreatedAt": 1575554005.0,
    "State": "PENDING"
  }
}
```

Para obtener más información, consulte [Trabajar con dispositivos](#) en la Guía del administrador de red de Transit Gateway.

- Para API obtener más información, consulte [CreateDevice](#) la Referencia de AWS CLI comandos.

## create-global-network

En el siguiente ejemplo de código, se muestra cómo usar `create-global-network`.

### AWS CLI

Para crear una red global

Los siguientes `create-global-network` ejemplos crean una nueva red global. El estado inicial tras la creación es `PENDING`.

```
aws networkmanager create-global-network
```

Salida:

```
{
  "GlobalNetwork": {
    "GlobalNetworkId": "global-network-00a77fc0f722dae74",
    "GlobalNetworkArn": "arn:aws:networkmanager::987654321012:global-network/global-network-00a77fc0f722dae74",
    "CreatedAt": "2022-03-14T20:31:56+00:00",
    "State": "PENDING"
  }
}
```

- Para API obtener más información, consulte [CreateGlobalNetwork](#) la Referencia de AWS CLI comandos.

## `create-link`

En el siguiente ejemplo de código, se muestra cómo usar `create-link`.

### AWS CLI

Para crear un enlace

El siguiente `create-link` ejemplo crea un enlace en la red global especificada. El enlace incluye una descripción y detalles sobre el tipo de enlace, el ancho de banda y el proveedor. El ID del sitio indica el sitio al que está asociado el enlace.

```
aws networkmanager create-link \
  --global-network-id global-network-01231231231231231 \
  --description "VPN Link" \
  --type "broadband" \
  --bandwidth UploadSpeed=10,DownloadSpeed=20 \
  --provider "AnyCompany" \
  --site-id site-444555aaabbb11223 \
  --region us-west-2
```

**Salida:**

```
{
  "Link": {
    "LinkId": "link-11112222aaaabbbb1",
    "LinkArn": "arn:aws:networkmanager::123456789012:link/global-
network-01231231231231231/link-11112222aaaabbbb1",
    "GlobalNetworkId": "global-network-01231231231231231",
    "SiteId": "site-444555aaaabbb11223",
    "Description": "VPN Link",
    "Type": "broadband",
    "Bandwidth": {
      "UploadSpeed": 10,
      "DownloadSpeed": 20
    },
    "Provider": "AnyCompany",
    "CreatedAt": 1575555811.0,
    "State": "PENDING"
  }
}
```

Para obtener más información, consulte Cómo [trabajar con enlaces](#) en la Guía del administrador de red de Transit Gateway.

- Para API obtener más información, consulte [CreateLink](#) la Referencia de AWS CLI comandos.

**create-site**

En el siguiente ejemplo de código, se muestra cómo usar `create-site`.

**AWS CLI**

Para crear un sitio

El siguiente `create-site` ejemplo crea un sitio en la red global especificada. Los detalles del sitio incluyen una descripción y la información de ubicación.

```
aws networkmanager create-site \
  --global-network-id global-network-01231231231231231 \
  --description "New York head office" \
  --location Latitude=40.7128,Longitude=-74.0060 \
  --region us-west-2
```



Salida:

```
{
  "Site": {
    "SiteId": "site-444555aaabbb11223",
    "SiteArn": "arn:aws:networkmanager::123456789012:site/global-
network-01231231231231231/site-444555aaabbb11223",
    "GlobalNetworkId": "global-network-01231231231231231",
    "Description": "New York head office",
    "Location": {
      "Latitude": "40.7128",
      "Longitude": "-74.0060"
    },
    "CreatedAt": 1575554300.0,
    "State": "PENDING"
  }
}
```

Para obtener más información, consulte [Trabajar con sitios](#) en la Guía del administrador de red de Transit Gateway.

- Para API obtener más información, consulte [CreateSite](#) la Referencia de AWS CLI comandos.

## create-vpc-attachment

En el siguiente ejemplo de código, se muestra cómo usar create-vpc-attachment.

AWS CLI

Para crear un VPC adjunto

En el siguiente create-vpc-attachment ejemplo, se crea un VPC adjunto con IPv6 soporte en una red principal.

```
aws networkmanager create-vpc-attachment \
  --core-network-id core-network-0fab62fe438d94db6 \
  --vpc-arn arn:aws:ec2:us-east-1:987654321012:vpc/vpc-09f37f69e2786eeb8 \
  --subnet-arns arn:aws:ec2:us-east-1:987654321012:subnet/subnet-04ca4e010857e7bb7 \
  --Ipv6Support=true
```

Salida:

```
{
  "VpcAttachment": {
    "Attachment": {
      "CoreNetworkId": "core-network-0fab62fe438d94db6",
      "AttachmentId": "attachment-05e1da6eba87a06e6",
      "OwnerAccountId": "987654321012",
      "AttachmentType": "VPC",
      "State": "CREATING",
      "EdgeLocation": "us-east-1",
      "ResourceArn": "arn:aws:ec2:us-east-1:987654321012:vpc/
vpc-09f37f69e2786eeb8",
      "Tags": [],
      "CreatedAt": "2022-03-10T20:59:14+00:00",
      "UpdatedAt": "2022-03-10T20:59:14+00:00"
    },
    "SubnetArns": [
      "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-04ca4e010857e7bb7"
    ],
    "Options": {
      "Ipv6Support": true
    }
  }
}
```

Para obtener más información, consulte [Crear un archivo adjunto](#) en la Guía del WAN usuario de Cloud.

- Para API obtener más información, consulte [CreateVpcAttachment](#) la Referencia de AWS CLI comandos.

## delete-attachment

En el siguiente ejemplo de código, se muestra cómo usar delete-attachment.

### AWS CLI

Para eliminar un adjunto

En el siguiente delete-attachment ejemplo, se elimina un adjunto de Connect.

```
aws networkmanager delete-attachment \
  --attachment-id attachment-01feddaeae26ab68c
```

Salida:

```
{
  "Attachment": {
    "CoreNetworkId": "core-network-0f4b0a9d5ee7761d1",
    "AttachmentId": "attachment-01feddaeae26ab68c",
    "OwnerAccountId": "987654321012",
    "AttachmentType": "CONNECT",
    "State": "DELETING",
    "EdgeLocation": "us-east-1",
    "ResourceArn": "arn:aws:networkmanager::987654321012:attachment/attachment-02c3964448fedf5aa",
    "CreatedAt": "2022-03-15T19:18:41+00:00",
    "UpdatedAt": "2022-03-15T19:28:59+00:00"
  }
}
```

Para obtener más información, consulte [Eliminar archivos adjuntos](#) en la Guía del WAN usuario de Cloud.

- Para API obtener más información, consulte [DeleteAttachment](#) la Referencia de AWS CLI comandos.

## delete-bucket-analytics-configuration

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-analytics-configuration`.

AWS CLI

Eliminar una configuración de análisis de un bucket

En el siguiente ejemplo de `delete-bucket-analytics-configuration`, se elimina la configuración de análisis para el bucket e ID especificados.

```
aws s3api delete-bucket-analytics-configuration \
  --bucket my-bucket \
  --id 1
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteBucketAnalyticsConfiguration](#) la Referencia de AWS CLI comandos.

## delete-bucket-metrics-configuration

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-metrics-configuration`.

### AWS CLI

Eliminar una configuración de métricas de un bucket

En el siguiente ejemplo de `delete-bucket-metrics-configuration`, se elimina la configuración de métricas para el bucket e ID especificados.

```
aws s3api delete-bucket-metrics-configuration \  
  --bucket my-bucket \  
  --id 123
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteBucketMetricsConfiguration](#) la Referencia de AWS CLI comandos.

## delete-core-network

En el siguiente ejemplo de código, se muestra cómo usar `delete-core-network`.

### AWS CLI

Para eliminar una red principal

En el siguiente `delete-core-network` ejemplo, se elimina una red principal de una red WAN global en la nube.

```
aws networkmanager delete-core-network \  
  --core-network-id core-network-0fab62fe438d94db6
```

Salida:

```
{
```

```
"CoreNetwork": {
  "GlobalNetworkId": "global-network-0d59060f16a73bc41",
  "CoreNetworkId": "core-network-0fab62fe438d94db6",
  "Description": "Main headquarters location",
  "CreatedAt": "2021-12-09T18:31:11+00:00",
  "State": "DELETING",
  "Segments": [
    {
      "Name": "dev",
      "EdgeLocations": [
        "us-east-1"
      ],
      "SharedSegments": []
    }
  ],
  "Edges": [
    {
      "EdgeLocation": "us-east-1",
      "Asn": 64512,
      "InsideCidrBlocks": []
    }
  ]
}
```

Para obtener más información, consulte la Guía del WAN usuario sobre [las redes principales](#) en la nube.

- Para API obtener más información, consulte [DeleteCoreNetwork](#) la Referencia de AWS CLI comandos.

## delete-device

En el siguiente ejemplo de código, se muestra cómo usar delete-device.

### AWS CLI

Para eliminar un dispositivo

En el siguiente delete-device ejemplo, se elimina el dispositivo especificado de la red global especificada.

```
aws networkmanager delete-device \
```

```
--global-network-id global-network-01231231231231231 \  
--device-id device-07f6fd08867abc123 \  
--region us-west-2
```

Salida:

```
{  
  "Device": {  
    "DeviceId": "device-07f6fd08867abc123",  
    "DeviceArn": "arn:aws:networkmanager::123456789012:device/global-  
network-01231231231231231/device-07f6fd08867abc123",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "Description": "New York office device",  
    "Type": "office device",  
    "Vendor": "anycompany",  
    "Model": "abcabc",  
    "SerialNumber": "1234",  
    "SiteId": "site-444555aaabbb11223",  
    "CreatedAt": 1575554005.0,  
    "State": "DELETING"  
  }  
}
```

Para obtener más información, consulte [Trabajar con dispositivos](#) en la Guía del administrador de red de Transit Gateway.

- Para API obtener más información, consulte [DeleteDevice](#) la Referencia de AWS CLI comandos.

## delete-global-network

En el siguiente ejemplo de código, se muestra cómo usar delete-global-network.

AWS CLI

Para eliminar una red global

En el siguiente delete-global-network ejemplo, se elimina una red global.

```
aws networkmanager delete-global-network \  
--global-network-id global-network-052bedddccb193b6b
```

Salida:

```
{
  "GlobalNetwork": {
    "GlobalNetworkId": "global-network-052bedddccb193b6b",
    "GlobalNetworkArn": "arn:aws:networkmanager::987654321012:global-network/global-network-052bedddccb193b6b",
    "CreatedAt": "2021-12-09T18:19:12+00:00",
    "State": "DELETING"
  }
}
```

- Para API obtener más información, consulte [DeleteGlobalNetwork](#) la Referencia de AWS CLI comandos.

## delete-link

En el siguiente ejemplo de código, se muestra cómo usar delete-link.

AWS CLI

Para eliminar un enlace

En el siguiente delete-link ejemplo, se elimina el enlace especificado de la red global especificada.

```
aws networkmanager delete-link \
  --global-network-id global-network-01231231231231231 \
  --link-id link-11112222aaaabbbb1 \
  --region us-west-2
```

Salida:

```
{
  "Link": {
    "LinkId": "link-11112222aaaabbbb1",
    "LinkArn": "arn:aws:networkmanager::123456789012:link/global-network-01231231231231231/link-11112222aaaabbbb1",
    "GlobalNetworkId": "global-network-01231231231231231",
    "SiteId": "site-444555aaaabbb11223",
    "Description": "VPN Link",
  }
}
```

```
    "Type": "broadband",
    "Bandwidth": {
      "UploadSpeed": 20,
      "DownloadSpeed": 20
    },
    "Provider": "AnyCompany",
    "CreatedAt": 1575555811.0,
    "State": "DELETING"
  }
}
```

Para obtener más información, consulte Cómo [trabajar con enlaces](#) en la Guía del administrador de red de Transit Gateway.

- Para API obtener más información, consulte [DeleteLink](#) la Referencia de AWS CLI comandos.

## delete-public-access-block

En el siguiente ejemplo de código, se muestra cómo usar `delete-public-access-block`.

### AWS CLI

Eliminar la configuración de bloqueo de acceso público de un bucket

En el siguiente ejemplo de `delete-public-access-block`, se elimina la configuración de bloqueo de acceso público en el bucket especificado.

```
aws s3api delete-public-access-block \
  --bucket my-bucket
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeletePublicAccessBlock](#) la Referencia de AWS CLI comandos.

## delete-site

En el siguiente ejemplo de código, se muestra cómo usar `delete-site`.

### AWS CLI

Para eliminar un sitio



En el siguiente `delete-site` ejemplo, se elimina el sitio especificado (`site-444555aaabbb11223`) de la red global especificada.

```
aws networkmanager delete-site \  
  --global-network-id global-network-01231231231231231 \  
  --site-id site-444555aaabbb11223 \  
  --region us-west-2
```

Salida:

```
{  
  "Site": {  
    "SiteId": "site-444555aaabbb11223",  
    "SiteArn": "arn:aws:networkmanager::123456789012:site/global-  
network-01231231231231231/site-444555aaabbb11223",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "Description": "New York head office",  
    "Location": {  
      "Latitude": "40.7128",  
      "Longitude": "-74.0060"  
    },  
    "CreatedAt": 1575554300.0,  
    "State": "DELETING"  
  }  
}
```

Para obtener más información, consulte [Trabajar con sitios](#) en la Guía del administrador de red de Transit Gateway.

- Para API obtener más información, consulte [DeleteSite](#) la Referencia de AWS CLI comandos.

## deregister-transit-gateway

En el siguiente ejemplo de código, se muestra cómo usar `deregister-transit-gateway`.

### AWS CLI

Para anular el registro de una puerta de enlace de tránsito de una red global

El siguiente `deregister-transit-gateway` ejemplo anula el registro de la puerta de enlace de tránsito especificada de la red global especificada.

```
aws networkmanager deregister-transit-gateway \  
  --global-network-id global-network-01231231231231231 \  
  --transit-gateway-arn arn:aws:ec2:us-west-2:123456789012:transit-gateway/  
tgw-123abc05e04123abc \  
  --region us-west-2
```

Salida:

```
{  
  "TransitGatewayRegistration": {  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "TransitGatewayArn": "arn:aws:ec2:us-west-2:123456789012:transit-gateway/  
tgw-123abc05e04123abc",  
    "State": {  
      "Code": "DELETING"  
    }  
  }  
}
```

Para obtener más información, consulte [Registros de Transit Gateway](#) en la Guía del administrador de redes de Transit Gateway.

- Para API obtener más información, consulte [DeregisterTransitGateway](#) la Referencia de AWS CLI comandos.

## describe-global-networks

En el siguiente ejemplo de código, se muestra cómo usar `describe-global-networks`.

AWS CLI

Para describir sus redes globales

En el siguiente `describe-global-networks` ejemplo, se describen todas las redes globales de su cuenta.

```
aws networkmanager describe-global-networks \  
  --region us-west-2
```

Salida:

```
{
  "GlobalNetworks": [
    {
      "GlobalNetworkId": "global-network-01231231231231231",
      "GlobalNetworkArn": "arn:aws:networkmanager::123456789012:global-
network/global-network-01231231231231231",
      "Description": "Company 1 global network",
      "CreatedAt": 1575553525.0,
      "State": "AVAILABLE"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeGlobalNetworks](#) la Referencia de AWS CLI comandos.

## disassociate-customer-gateway

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-customer-gateway`.

### AWS CLI

Para desasociar una pasarela de clientes

El siguiente `disassociate-customer-gateway` ejemplo desasocia la pasarela de clientes especificada (`cgw-11223344556677889`) de la red global especificada.

```
aws networkmanager disassociate-customer-gateway \
  --global-network-id global-network-01231231231231231 \
  --customer-gateway-arn arn:aws:ec2:us-west-2:123456789012:customer-gateway/
cgw-11223344556677889 \
  --region us-west-2
```

Salida:

```
{
  "CustomerGatewayAssociation": {
    "CustomerGatewayArn": "arn:aws:ec2:us-west-2:123456789012:customer-gateway/
cgw-11223344556677889",
    "GlobalNetworkId": "global-network-01231231231231231",
    "DeviceId": "device-07f6fd08867abc123",
  }
}
```

```
    "State": "DELETING"
  }
}
```

Para obtener más información, consulte [Customer Gateway Associations](#) en la Guía del administrador de redes de Transit Gateway.

- Para API obtener más información, consulte [DisassociateCustomerGateway](#) la Referencia de AWS CLI comandos.

## disassociate-link

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-link`.

### AWS CLI

Para desasociar un enlace

El siguiente `disassociate-link` ejemplo desasocia el enlace especificado del dispositivo de `device-07f6fd08867abc123` la red global especificada.

```
aws networkmanager disassociate-link \
  --global-network-id global-network-01231231231231231 \
  --device-id device-07f6fd08867abc123 \
  --link-id link-11112222aaaabbbb1 \
  --region us-west-2
```

Salida:

```
{
  "LinkAssociation": {
    "GlobalNetworkId": "global-network-01231231231231231",
    "DeviceId": "device-07f6fd08867abc123",
    "LinkId": "link-11112222aaaabbbb1",
    "LinkAssociationState": "DELETING"
  }
}
```

Para obtener más información, consulte [Asociaciones de dispositivos y enlaces](#) en la Guía del administrador de red de Transit Gateway.

- Para API obtener más información, consulte [DisassociateLink](#) la Referencia de AWS CLI comandos.

## get-bucket-analytics-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-analytics-configuration`.

### AWS CLI

Recuperar la configuración de análisis de un bucket con un ID específico

En el siguiente ejemplo de `get-bucket-analytics-configuration`, se muestra la configuración de análisis para el bucket e ID especificados.

```
aws s3api get-bucket-analytics-configuration \
  --bucket my-bucket \
  --id 1
```

Salida:

```
{
  "AnalyticsConfiguration": {
    "StorageClassAnalysis": {},
    "Id": "1"
  }
}
```

- Para API obtener más información, consulte [GetBucketAnalyticsConfiguration](#) la Referencia de AWS CLI comandos.

## get-bucket-metrics-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-metrics-configuration`.

### AWS CLI

Recuperar la configuración de métricas de un bucket con un ID específico

En el siguiente ejemplo de `get-bucket-metrics-configuration`, se muestra la configuración de métricas para el bucket e ID especificados.

```
aws s3api get-bucket-metrics-configuration \  
  --bucket my-bucket \  
  --id 123
```

Salida:

```
{  
  "MetricsConfiguration": {  
    "Filter": {  
      "Prefix": "logs"  
    },  
    "Id": "123"  
  }  
}
```

- Para API obtener más información, consulte [GetBucketMetricsConfiguration](#) la Referencia de AWS CLI comandos.

## `get-customer-gateway-associations`

En el siguiente ejemplo de código, se muestra cómo usar `get-customer-gateway-associations`.

AWS CLI

Para obtener sus asociaciones de pasarela de clientes

En el siguiente `get-customer-gateway-associations` ejemplo, se obtienen las asociaciones de pasarelas de clientes de la red global especificada.

```
aws networkmanager get-customer-gateway-associations \  
  --global-network-id global-network-01231231231231231 \  
  --region us-west-2
```

Salida:

```
{  
  "CustomerGatewayAssociations": [  

```

```
{
  "CustomerGatewayArn": "arn:aws:ec2:us-west-2:123456789012:customer-
gateway/cgw-11223344556677889",
  "GlobalNetworkId": "global-network-01231231231231231",
  "DeviceId": "device-07f6fd08867abc123",
  "State": "AVAILABLE"
}
]
```

- Para API obtener más información, consulte [GetCustomerGatewayAssociations](#) la Referencia de AWS CLI comandos.

## get-devices

En el siguiente ejemplo de código, se muestra cómo usar `get-devices`.

### AWS CLI

Para obtener tus dispositivos

En el siguiente `get-devices` ejemplo, se obtienen los dispositivos de la red global especificada.

```
aws networkmanager get-devices \
  --global-network-id global-network-01231231231231231 \
  --region us-west-2
```

Salida:

```
{
  "Devices": [
    {
      "DeviceId": "device-07f6fd08867abc123",
      "DeviceArn": "arn:aws:networkmanager::123456789012:device/global-
network-01231231231231231/device-07f6fd08867abc123",
      "GlobalNetworkId": "global-network-01231231231231231",
      "Description": "NY office device",
      "Type": "office device",
      "Vendor": "anycompany",
      "Model": "abcabc",
      "SerialNumber": "1234",
      "CreatedAt": 1575554005.0,
    }
  ]
}
```

```
        "State": "AVAILABLE"
      }
    ]
  }
```

- Para API obtener más información, consulte [GetDevices](#) la Referencia de AWS CLI comandos.

## get-link-associations

En el siguiente ejemplo de código, se muestra cómo usar `get-link-associations`.

### AWS CLI

Para obtener sus asociaciones de enlaces

El siguiente `get-link-associations` ejemplo obtiene las asociaciones de enlaces de la red global especificada.

```
aws networkmanager get-link-associations \
  --global-network-id global-network-01231231231231231 \
  --region us-west-2
```

Salida:

```
{
  "LinkAssociations": [
    {
      "GlobalNetworkId": "global-network-01231231231231231",
      "DeviceId": "device-07f6fd08867abc123",
      "LinkId": "link-11112222aaaabbbb1",
      "LinkAssociationState": "AVAILABLE"
    }
  ]
}
```

- Para API obtener más información, consulte [GetLinkAssociations](#) la Referencia de AWS CLI comandos.

## get-links

En el siguiente ejemplo de código, se muestra cómo usar `get-links`.



## AWS CLI

Para obtener sus enlaces

El siguiente `get-links` ejemplo obtiene los enlaces de la red global especificada.

```
aws networkmanager get-links \  
  --global-network-id global-network-01231231231231231 \  
  --region us-west-2
```

Salida:

```
{  
  "Links": [  
    {  
      "LinkId": "link-11112222aaaabbbb1",  
      "LinkArn": "arn:aws:networkmanager::123456789012:link/global-  
network-01231231231231231/link-11112222aaaabbbb1",  
      "GlobalNetworkId": "global-network-01231231231231231",  
      "SiteId": "site-444555aaaabbb11223",  
      "Description": "VPN Link",  
      "Type": "broadband",  
      "Bandwidth": {  
        "UploadSpeed": 10,  
        "DownloadSpeed": 20  
      },  
      "Provider": "AnyCompany",  
      "CreatedAt": 1575555811.0,  
      "State": "AVAILABLE"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [GetLinks](#) la Referencia de AWS CLI comandos.

## get-object-retention

En el siguiente ejemplo de código, se muestra cómo usar `get-object-retention`.

## AWS CLI

Para recuperar la configuración de retención de un objeto

En el siguiente ejemplo de `get-object-retention`, se recupera la configuración de retención del objeto especificado.

```
aws s3api get-object-retention \  
  --bucket my-bucket-with-object-lock \  
  --key doc1.rtf
```

Salida:

```
{  
  "Retention": {  
    "Mode": "GOVERNANCE",  
    "RetainUntilDate": "2025-01-01T00:00:00.000Z"  
  }  
}
```

- Para API obtener más información, consulte [GetObjectRetention](#) la Referencia de AWS CLI comandos.

## **get-public-access-block**

En el siguiente ejemplo de código, se muestra cómo usar `get-public-access-block`.

AWS CLI

Establecer o modificar la configuración de bloqueo de acceso público de un bucket

En el siguiente ejemplo de `get-public-access-block`, se elimina la configuración de bloqueo de acceso público del bucket especificado.

```
aws s3api get-public-access-block --bucket my-bucket
```

Salida:

```
{  
  "PublicAccessBlockConfiguration": {  
    "IgnorePublicAcls": true,  
    "BlockPublicPolicy": true,  
    "BlockPublicAcls": true,  
    "RestrictPublicBuckets": true
```

```
}  
}
```

- Para API obtener más información, consulte [GetPublicAccessBlock](#) la Referencia de AWS CLI comandos.

## get-sites

En el siguiente ejemplo de código, se muestra cómo usar `get-sites`.

### AWS CLI

Para obtener sus sitios

El siguiente `get-sites` ejemplo obtiene los sitios de la red global especificada.

```
aws networkmanager get-sites \  
  --global-network-id global-network-01231231231231 \  
  --region us-west-2
```

Salida:

```
{  
  "Sites": [  
    {  
      "SiteId": "site-444555aaabbb11223",  
      "SiteArn": "arn:aws:networkmanager::123456789012:site/global-  
network-01231231231231231/site-444555aaabbb11223",  
      "GlobalNetworkId": "global-network-01231231231231231",  
      "Description": "NY head office",  
      "Location": {  
        "Latitude": "40.7128",  
        "Longitude": "-74.0060"  
      },  
      "CreatedAt": 1575554528.0,  
      "State": "AVAILABLE"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [GetSites](#) la Referencia de AWS CLI comandos.

## get-transit-gateway-registrations

En el siguiente ejemplo de código, se muestra cómo usar `get-transit-gateway-registrations`.

### AWS CLI

Para obtener sus registros en Transit Gateway

En el siguiente `get-transit-gateway-registrations` ejemplo, se obtienen las pasarelas de tránsito que están registradas en la red global especificada.

```
aws networkmanager get-transit-gateway-registrations \  
  --global-network-id global-network-01231231231231231 \  
  --region us-west-2
```

Salida:

```
{  
  "TransitGatewayRegistrations": [  
    {  
      "GlobalNetworkId": "global-network-01231231231231231",  
      "TransitGatewayArn": "arn:aws:ec2:us-west-2:123456789012:transit-  
gateway/tgw-123abc05e04123abc",  
      "State": {  
        "Code": "AVAILABLE"  
      }  
    }  
  ]  
}
```

- Para API obtener más información, consulte [GetTransitGatewayRegistrations](#) la Referencia de AWS CLI comandos.

## get-vpc-attachment

En el siguiente ejemplo de código, se muestra cómo usar `get-vpc-attachment`.

### AWS CLI

Para obtener un VPC archivo adjunto

El siguiente `get-vpc-attachment` ejemplo devuelve información sobre un VPC archivo adjunto.

```
aws networkmanager get-vpc-attachment \  
--attachment-id attachment-03b7ea450134787da
```

Salida:

```
{  
  "VpcAttachment": {  
    "Attachment": {  
      "CoreNetworkId": "core-network-0522de1b226a5d7b3",  
      "AttachmentId": "attachment-03b7ea450134787da",  
      "OwnerAccountId": "987654321012",  
      "AttachmentType": "VPC",  
      "State": "CREATING",  
      "EdgeLocation": "us-east-1",  
      "ResourceArn": "arn:aws:ec2:us-east-1:987654321012:vpc/vpc-a7c4bbda",  
      "Tags": [  
        {  
          "Key": "Name",  
          "Value": "DevVPC"  
        }  
      ],  
      "CreatedAt": "2022-03-11T17:48:58+00:00",  
      "UpdatedAt": "2022-03-11T17:48:58+00:00"  
    },  
    "SubnetArns": [  
      "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-202cde6c",  
      "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-e5022dba",  
      "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-2387ae02",  
      "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-cda9dfffc"  
    ],  
    "Options": {  
      "Ipv6Support": false  
    }  
  }  
}
```

Para obtener más información, consulte la Guía del WAN usuario de [Adjuntos](#) en la nube.

- Para API obtener más información, consulte [GetVpcAttachment](#) la Referencia de AWS CLI comandos.

## list-bucket-analytics-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-bucket-analytics-configurations`.

### AWS CLI

Recuperar una lista de configuraciones de análisis para un bucket

El siguiente `list-bucket-analytics-configurations` recupera una lista de configuraciones de análisis para el bucket especificado.

```
aws s3api list-bucket-analytics-configurations \
  --bucket my-bucket
```

Salida:

```
{
  "AnalyticsConfigurationList": [
    {
      "StorageClassAnalysis": {},
      "Id": "1"
    }
  ],
  "IsTruncated": false
}
```

- Para API obtener más información, consulte [ListBucketAnalyticsConfigurations](#) la Referencia de AWS CLI comandos.

## list-bucket-metrics-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-bucket-metrics-configurations`.

### AWS CLI

Para recuperar una lista de configuraciones de métricas para un bucket

En el siguiente `list-bucket-metrics-configurations` ejemplo, se recupera una lista de configuraciones de métricas para el depósito especificado.

```
aws s3api list-bucket-metrics-configurations \  
  --bucket my-bucket
```

Salida:

```
{  
  "IsTruncated": false,  
  "MetricsConfigurationList": [  
    {  
      "Filter": {  
        "Prefix": "logs"  
      },  
      "Id": "123"  
    },  
    {  
      "Filter": {  
        "Prefix": "tmp"  
      },  
      "Id": "234"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListBucketMetricsConfigurations](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

AWS CLI

Para enumerar las etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas del recurso de dispositivo especificado (`device-07f6fd08867abc123`).

```
aws networkmanager list-tags-for-resource \  
  --resource-arn arn:aws:networkmanager::123456789012:device/global-  
network-01231231231231231/device-07f6fd08867abc123 \  
  --region us-west-2
```

Salida:

```
{
  "TagList": [
    {
      "Key": "Network",
      "Value": "Northeast"
    }
  ]
}
```

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## put-bucket-metrics-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-metrics-configuration`.

AWS CLI

Para establecer una configuración de métricas para un bucket

En el siguiente `put-bucket-metrics-configuration` ejemplo, se establece una configuración métrica con el ID 123 para el depósito especificado.

```
aws s3api put-bucket-metrics-configuration \
  --bucket my-bucket \
  --id 123 \
  --metrics-configuration '{"Id": "123", "Filter": {"Prefix": "logs"}}'
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutBucketMetricsConfiguration](#) la Referencia de AWS CLI comandos.

## put-object-retention

En el siguiente ejemplo de código, se muestra cómo usar `put-object-retention`.



## AWS CLI

Para establecer la configuración de retención de un objeto

En el siguiente ejemplo de `put-object-retention`, se establece una configuración de retención del objeto especificado hasta el 1 de enero de 2025.

```
aws s3api put-object-retention \  
  --bucket my-bucket-with-object-lock \  
  --key doc1.rtf \  
  --retention '{ "Mode": "GOVERNANCE", "RetainUntilDate": "2025-01-01T00:00:00" }'
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutObjectRetention](#) la Referencia de AWS CLI comandos.

## put-public-access-block

En el siguiente ejemplo de código, se muestra cómo usar `put-public-access-block`.

## AWS CLI

Para establecer la configuración de acceso público en bloque para un bucket

En el siguiente `put-public-access-block` ejemplo, se establece una configuración restrictiva de acceso público en bloque para el depósito especificado.

```
aws s3api put-public-access-block \  
  --bucket my-bucket \  
  --public-access-block-  
configuration "BlockPublicAcls=true,IgnorePublicAcls=true,BlockPublicPolicy=true,RestrictPub
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutPublicAccessBlock](#) la Referencia de AWS CLI comandos.

## register-transit-gateway

En el siguiente ejemplo de código, se muestra cómo usar `register-transit-gateway`.

## AWS CLI

Para registrar una pasarela de tránsito en una red global

El siguiente `register-transit-gateway` ejemplo registra la puerta de enlace de tránsito `tgw-123abc05e04123abc` en la red global especificada.

```
aws networkmanager register-transit-gateway \  
  --global-network-id global-network-01231231231231231 \  
  --transit-gateway-arn arn:aws:ec2:us-west-2:123456789012:transit-gateway/  
tgw-123abc05e04123abc \  
  --region us-west-2
```

Salida:

```
{  
  "TransitGatewayRegistration": {  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "TransitGatewayArn": "arn:aws:ec2:us-west-2:123456789012:transit-gateway/  
tgw-123abc05e04123abc",  
    "State": {  
      "Code": "PENDING"  
    }  
  }  
}
```

Para obtener más información, consulte [Registros de Transit Gateway](#) en la Guía del administrador de redes de Transit Gateway.

- Para API obtener más información, consulte [RegisterTransitGateway](#) la Referencia de AWS CLI comandos.

## reject-attachment

En el siguiente ejemplo de código, se muestra cómo usar `reject-attachment`.

### AWS CLI

Para rechazar un adjunto

El siguiente `reject-attachment` ejemplo rechaza una solicitud de VPC adjunto.

```
aws networkmanager reject-attachment \  

```

```
--attachment-id attachment-03b7ea450134787da
```

Salida:

```
{
  "Attachment": {
    "CoreNetworkId": "core-network-0522de1b226a5d7b3",
    "AttachmentId": "attachment-03b7ea450134787da",
    "OwnerAccountId": "987654321012",
    "AttachmentType": "VPC",
    "State": "AVAILABLE",
    "EdgeLocation": "us-east-1",
    "ResourceArn": "arn:aws:ec2:us-east-1:987654321012:vpc/vpc-a7c4bbda",
    "CreatedAt": "2022-03-11T17:48:58+00:00",
    "UpdatedAt": "2022-03-11T17:51:25+00:00"
  }
}
```

Para obtener más información, consulte [Aceptación de archivos adjuntos](#) en la Guía del WAN usuario de Cloud.

- Para API obtener más información, consulte [RejectAttachment](#) la Referencia de AWS CLI comandos.

## start-route-analysis

En el siguiente ejemplo de código, se muestra cómo usar `start-route-analysis`.

### AWS CLI

Para iniciar el análisis de rutas

El siguiente `start-route-analysis` ejemplo inicia el análisis entre un origen y un destino, incluido el opcional `include-return-path`.

```
aws networkmanager start-route-analysis \
  --global-network-id global-network-00aa0aaa0b0aaa000 \
  --source TransitGatewayAttachmentArn=arn:aws:ec2:us-east-1:503089527312:transit-  
gateway-attachment/tgw-attach-0d4a2d491bf68c093,IpAddress=10.0.0.0 \
  --destination TransitGatewayAttachmentArn=arn:aws:ec2:us-  
west-1:503089527312:transit-gateway-attachment/tgw-  
attach-002577f30bb181742,IpAddress=11.0.0.0 \
```

**--include-return-path**

Salida:

```
{
  "RouteAnalysis": {
    "GlobalNetworkId": "global-network-00aa0aaa0b0aaa000",
    "OwnerAccountId": "1111222233333",
    "RouteAnalysisId": "a1873de1-273c-470c-1a2bc2345678",
    "StartTimestamp": 1695760154.0,
    "Status": "RUNNING",
    "Source": {
      "TransitGatewayAttachmentArn": "arn:aws:ec2:us-east-1:111122223333:transit-gateway-attachment/tgw-attach-1234567890abcdef0",
      "TransitGatewayArn": "arn:aws:ec2:us-east-1:111122223333:transit-gateway/tgw-abcdef01234567890",
      "IpAddress": "10.0.0.0"
    },
    "Destination": {
      "TransitGatewayAttachmentArn": "arn:aws:ec2:us-west-1:555555555555:transit-gateway-attachment/tgw-attach-021345abcdef6789",
      "TransitGatewayArn": "arn:aws:ec2:us-west-1:111122223333:transit-gateway/tgw-09876543210fedcba0",
      "IpAddress": "11.0.0.0"
    },
    "IncludeReturnPath": true,
    "UseMiddleboxes": false
  }
}
```

Para obtener más información, consulte [Route Analyzer](#) en la Guía del usuario de AWS Global Networks for Transit Gateways.

- Para API obtener más información, consulte la Referencia [StartRouteAnalysis](#) de AWS CLI comandos.

**tag-resource**

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

**AWS CLI**

Para aplicar etiquetas a un recurso

En el siguiente `tag-resource` ejemplo, se aplica la etiqueta `Network=Northeast` al dispositivo `device-07f6fd08867abc123`.

```
aws networkmanager tag-resource \  
  --resource-arn arn:aws:networkmanager::123456789012:device/global-  
network-01231231231231231/device-07f6fd08867abc123 \  
  --tags Key=Network,Value=Northeast \  
  --region us-west-2
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Cómo eliminar etiquetas de un recurso

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta con la clave `Network` del dispositivo `device-07f6fd08867abc123`.

```
aws networkmanager untag-resource \  
  --resource-arn arn:aws:networkmanager::123456789012:device/global-  
network-01231231231231231/device-07f6fd08867abc123 ]  
  --tag-keys Network \  
  --region us-west-2
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-device

En el siguiente ejemplo de código, se muestra cómo usar `update-device`.

## AWS CLI

Para actualizar un dispositivo

El siguiente `update-device` ejemplo actualiza el dispositivo `device-07f6fd08867abc123` especificando un ID de sitio para el dispositivo.

```
aws networkmanager update-device \  
  --global-network-id global-network-01231231231231231 \  
  --device-id device-07f6fd08867abc123 \  
  --site-id site-444555aaabbb11223 \  
  --region us-west-2
```

Salida:

```
{  
  "Device": {  
    "DeviceId": "device-07f6fd08867abc123",  
    "DeviceArn": "arn:aws:networkmanager::123456789012:device/global-  
network-01231231231231231/device-07f6fd08867abc123",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "Description": "NY office device",  
    "Type": "Office device",  
    "Vendor": "anycompany",  
    "Model": "abcabc",  
    "SerialNumber": "1234",  
    "SiteId": "site-444555aaabbb11223",  
    "CreatedAt": 1575554005.0,  
    "State": "UPDATING"  
  }  
}
```

Para obtener más información, consulte [Trabajar con dispositivos](#) en la Guía del administrador de red de Transit Gateway.

- Para API obtener más información, consulte [UpdateDevice](#) la Referencia de AWS CLI comandos.

## update-global-network

En el siguiente ejemplo de código, se muestra cómo usar `update-global-network`.

## AWS CLI

Para actualizar una red global

El siguiente `update-global-network` ejemplo actualiza la descripción de la red `global-network-01231231231231231`.

```
aws networkmanager update-global-network \  
  --global-network-id global-network-01231231231231231 \  
  --description "Head offices" \  
  --region us-west-2
```

Salida:

```
{  
  "GlobalNetwork": {  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "GlobalNetworkArn": "arn:aws:networkmanager::123456789012:global-network/  
global-network-01231231231231231",  
    "Description": "Head offices",  
    "CreatedAt": 1575553525.0,  
    "State": "UPDATING"  
  }  
}
```

Para obtener más información, consulte [Redes globales](#) en la Guía del administrador de redes de Transit Gateway.

- Para API obtener más información, consulte [UpdateGlobalNetwork](#) la Referencia de AWS CLI comandos.

## update-link

En el siguiente ejemplo de código, se muestra cómo usar `update-link`.

## AWS CLI

Para actualizar un enlace

El siguiente `update-link` ejemplo actualiza la información de ancho de banda del enlace `link-11112222aaaabbbb1`.

```
aws networkmanager update-link \  
  --global-network-id global-network-01231231231231231 \  
  --link-id Link-11112222aaaabbbb1 \  
  --bandwidth UploadSpeed=20,DownloadSpeed=20 \  
  --region us-west-2
```

Salida:

```
{  
  "Link": {  
    "LinkId": "link-11112222aaaabbbb1",  
    "LinkArn": "arn:aws:networkmanager::123456789012:link/global-  
network-01231231231231231/link-11112222aaaabbbb1",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "SiteId": "site-444555aaabbb11223",  
    "Description": "VPN Link",  
    "Type": "broadband",  
    "Bandwidth": {  
      "UploadSpeed": 20,  
      "DownloadSpeed": 20  
    },  
    "Provider": "AnyCompany",  
    "CreatedAt": 1575555811.0,  
    "State": "UPDATING"  
  }  
}
```

Para obtener más información, consulte Cómo [trabajar con enlaces](#) en la Guía del administrador de red de Transit Gateway.

- Para API obtener más información, consulte [UpdateLink](#) la Referencia de AWS CLI comandos.

## update-site

En el siguiente ejemplo de código, se muestra cómo usar `update-site`.

### AWS CLI

Para actualizar un sitio

En el siguiente `update-site` ejemplo, se actualiza la descripción del sitio `site-444555aaabbb11223` de la red global especificada.



```
aws networkmanager update-site \  
  --global-network-id global-network-01231231231231231 \  
  --site-id site-444555aaabbb11223 \  
  --description "New York Office site" \  
  --region us-west-2
```

Salida:

```
{  
  "Site": {  
    "SiteId": "site-444555aaabbb11223",  
    "SiteArn": "arn:aws:networkmanager::123456789012:site/global-  
network-01231231231231231/site-444555aaabbb11223",  
    "GlobalNetworkId": "global-network-01231231231231231",  
    "Description": "New York Office site",  
    "Location": {  
      "Latitude": "40.7128",  
      "Longitude": "-74.0060"  
    },  
    "CreatedAt": 1575554528.0,  
    "State": "UPDATING"  
  }  
}
```

Para obtener más información, consulte [Trabajar con sitios](#) en la Guía del administrador de red de Transit Gateway.

- Para API obtener más información, consulte [UpdateSite](#) la Referencia de AWS CLI comandos.

## Ejemplos de Nimble Studio usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Nimble Studio.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puedes encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### get-eula

En el siguiente ejemplo de código, se muestra cómo usar `get-eula`.

#### AWS CLI

Para obtener información sobre tu estudio

El siguiente `get-eula` ejemplo muestra la información sobre un EULA.

```
aws nimble get-eula \  
  --eula-id "EULAid"
```

Salida:

```
{  
  "eula": {  
    "content": "https://www.mozilla.org/en-US/MPL/2.0/",  
    "createdAt": "2021-04-20T16:45:23+00:00",  
    "eulaId": "gJZLygd-Srq_5NNbSfiaLg",  
    "name": "Mozilla-FireFox",  
    "updatedAt": "2021-04-20T16:45:23+00:00"  
  }  
}
```

Para obtener más información, consulte [Aceptar el EULA en la](#) Guía del usuario de Amazon Nimble Studio.

- Para API obtener más información, consulte [GetEula](#) la Referencia de AWS CLI comandos.

### get-launch-profile-details

En el siguiente ejemplo de código, se muestra cómo usar `get-launch-profile-details`.

#### AWS CLI

Para ver una lista de los widgets disponibles

En el siguiente `get-launch-profile-details` ejemplo, se enumeran los detalles de un perfil de lanzamiento.

```
aws nimble get-launch-profile-details \  
  --studio-id "StudioID" \  
  --launch-profile-id "LaunchProfileID"
```

Salida:

```
{  
  "launchProfile": {  
    "arn": "arn:aws:nimble:us-west-2:123456789012:launch-profile/  
yeG7lDwNQEiwNTRT7DrV7Q",  
    "createdAt": "2022-01-27T21:18:59+00:00",  
    "createdBy": "AROA3002NEHCCYRNDDIFT:i-EXAMPLE11111",  
    "description": "The Launch Profile for the Render workers created by  
StudioBuilder.",  
    "ec2SubnetIds": [  
      "subnet-EXAMPLE11111"  
    ],  
    "launchProfileId": "yeG7lDwNQEiwNTRT7DrV7Q",  
    "launchProfileProtocolVersions": [  
      "2021-03-31"  
    ],  
    "name": "RenderWorker-Default",  
    "state": "READY",  
    "statusCode": "LAUNCH_PROFILE_CREATED",  
    "statusMessage": "Launch Profile has been created",  
    "streamConfiguration": {  
      "clipboardMode": "ENABLED",  
      "ec2InstanceTypes": [  
        "g4dn.4xlarge",  
        "g4dn.8xlarge"  
      ],  
      "maxSessionLengthInMinutes": 690,  
      "maxStoppedSessionLengthInMinutes": 0,  
      "streamingImageIds": [  
        "Cw_jXnp1QcSSXhE2hkNRoQ",  
        "YGXAqgoWTnCNSV8VP20sHQ"  
      ]  
    },  
    "studioComponentIds": [  
      "_hR_-RaAReS0jAnLakbX7Q",  
    ]  
  }  
}
```

```
    "vQ5w_TbIRayPkAZgcbYRA",
    "ZQuMxN99Qfa_Js6ma9TwdA",
    "45Kj0SPPrzK20yvpCuQ6qw"
  ],
  "tags": {
    "resourceArn": "arn:aws:nimble:us-west-2:123456789012:launch-profile/
yeG71DwNQEiwNTRT7DrV7Q"
  },
  "updatedAt": "2022-01-27T21:19:13+00:00",
  "updatedBy": "AROA3002NEHCCYRNDDIFT:i-00b98256b04d9e989",
  "validationResults": [
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_ACTIVE_DIRECTORY_STUDIO_COMPONENT"
    },
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_SUBNET_ASSOCIATION"
    },
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_NETWORK_ACL_ASSOCIATION"
    },
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_SECURITY_GROUP_ASSOCIATION"
    }
  ]
},
"streamingImages": [
  {
    "arn": "arn:aws:nimble:us-west-2:123456789012:streaming-image/
Cw_jXnp1QcSSXhE2hkNRoQ",
    "description": "Base windows image for NimbleStudio",
    "ec2ImageId": "ami-EXAMPLE11111",
    "eulaIds": [
```

```

        "gJZLygd-Srq_5NNbSfiaLg",
        "ggK2eIw6RQyt8PIee0lD3g",
        "a-D9Wc0VQCKUfxAinCDxaw",
        "RvoNmVXiSrS4LhLTb6ybkw",
        "wtp85BcSTa2NZeNRnMKdjw",
        "Rl-J0fM5S12hyIiwWIV6hw"
    ],
    "name": "NimbleStudioWindowsStreamImage",
    "owner": "amazon",
    "platform": "WINDOWS",
    "state": "READY",
    "streamingImageId": "Cw_jXnp1QcSSXhE2hkNRoQ",
    "tags": {
        "resourceArn": "arn:aws:nimble:us-west-2:123456789012:streaming-
image/Cw_jXnp1QcSSXhE2hkNRoQ"
    }
},
{
    "arn": "arn:aws:nimble:us-west-2:123456789012:streaming-image/
YGXAqgoWTnCNSV8VP20sHQ",
    "description": "Base linux image for NimbleStudio",
    "ec2ImageId": "ami-EXAMPLE11111",
    "eulaIds": [
        "gJZLygd-Srq_5NNbSfiaLg",
        "ggK2eIw6RQyt8PIee0lD3g",
        "a-D9Wc0VQCKUfxAinCDxaw",
        "RvoNmVXiSrS4LhLTb6ybkw",
        "wtp85BcSTa2NZeNRnMKdjw",
        "Rl-J0fM5S12hyIiwWIV6hw"
    ],
    "name": "NimbleStudioLinuxStreamImage",
    "owner": "amazon",
    "platform": "LINUX",
    "state": "READY",
    "streamingImageId": "YGXAqgoWTnCNSV8VP20sHQ",
    "tags": {
        "resourceArn": "arn:aws:nimble:us-west-2:123456789012:streaming-
image/YGXAqgoWTnCNSV8VP20sHQ"
    }
}
],
"studioComponentSummaries": [
    {
        "description": "FSx for Windows",

```

```

        "name": "FSxWindows",
        "studioComponentId": "ZQuMxN99Qfa Js6ma9TwdA",
        "subtype": "AMAZON_FSX_FOR_WINDOWS",
        "type": "SHARED_FILE_SYSTEM"
    },
    {
        "description": "Instance configuration studio component.",
        "name": "InstanceConfiguration",
        "studioComponentId": "vQ5w_TbIRayPkAZgcbYRA",
        "subtype": "CUSTOM",
        "type": "CUSTOM"
    },
    {
        "name": "ActiveDirectory",
        "studioComponentId": "_hR_-RaAReS0jAnLakbX7Q",
        "subtype": "AWS_MANAGED_MICROSOFT_AD",
        "type": "ACTIVE_DIRECTORY"
    },
    {
        "description": "Render farm running Deadline",
        "name": "RenderFarm",
        "studioComponentId": "45Kj0SPPrzK20yvpCuQ6qw",
        "subtype": "CUSTOM",
        "type": "COMPUTE_FARM"
    }
]
}

```

Para obtener más información, consulte [Creación de perfiles de lanzamiento](#) en la Guía del usuario de Amazon Nimble Studio.

- Para API obtener más información, consulte [GetLaunchProfileDetails](#) la Referencia de AWS CLI comandos.

## get-launch-profile

En el siguiente ejemplo de código, se muestra cómo usar `get-launch-profile`.

### AWS CLI

Para ver una lista de los widgets disponibles

En el siguiente `get-launch-profile` ejemplo, se muestra información sobre un perfil de lanzamiento.

```
aws nimble get-launch-profile \  
  --studio-id "StudioID" \  
  --launch-profile-id "LaunchProfileID"
```

Salida:

```
{  
  "launchProfile": {  
    "arn": "arn:aws:nimble:us-west-2:123456789012:launch-profile/  
yeG7lDwNQEiwNTRT7DrV7Q",  
    "createdAt": "2022-01-27T21:18:59+00:00",  
    "createdBy": "AROA3002NEHCCYRNDDIFT:i-EXAMPLE11111",  
    "description": "The Launch Profile for the Render workers created by  
StudioBuilder.",  
    "ec2SubnetIds": [  
      "subnet-EXAMPLE11111"  
    ],  
    "launchProfileId": "yeG7lDwNQEiwNTRT7DrV7Q",  
    "launchProfileProtocolVersions": [  
      "2021-03-31"  
    ],  
    "name": "RenderWorker-Default",  
    "state": "READY",  
    "statusCode": "LAUNCH_PROFILE_CREATED",  
    "statusMessage": "Launch Profile has been created",  
    "streamConfiguration": {  
      "clipboardMode": "ENABLED",  
      "ec2InstanceTypes": [  
        "g4dn.4xlarge",  
        "g4dn.8xlarge"  
      ],  
      "maxSessionLengthInMinutes": 690,  
      "maxStoppedSessionLengthInMinutes": 0,  
      "streamingImageIds": [  
        "Cw_jXnp1QcSSXhE2hkNRoQ",  
        "YGXAqgoWTnCNSV8VP20sHQ"  
      ]  
    },  
    "studioComponentIds": [  
      "_hR_-RaAReS0jAnLakbX7Q",  
    ]  
  }  
}
```

```

        "vQ5w_TbIRayPkAZgcbYRA",
        "ZQuMxN99Qfa_Js6ma9TwdA",
        "45Kj0SPPrzK20yvpCuQ6qw"
    ],
    "tags": {},
    "updatedAt": "2022-01-27T21:19:13+00:00",
    "updatedBy": "AR0A3002NEHCCYRNDDIFT:i-00b98256b04d9e989",
    "validationResults": [
        {
            "state": "VALIDATION_SUCCESS",
            "statusCode": "VALIDATION_SUCCESS",
            "statusMessage": "The validation succeeded.",
            "type": "VALIDATE_ACTIVE_DIRECTORY_STUDIO_COMPONENT"
        },
        {
            "state": "VALIDATION_SUCCESS",
            "statusCode": "VALIDATION_SUCCESS",
            "statusMessage": "The validation succeeded.",
            "type": "VALIDATE_SUBNET_ASSOCIATION"
        },
        {
            "state": "VALIDATION_SUCCESS",
            "statusCode": "VALIDATION_SUCCESS",
            "statusMessage": "The validation succeeded.",
            "type": "VALIDATE_NETWORK_ACL_ASSOCIATION"
        },
        {
            "state": "VALIDATION_SUCCESS",
            "statusCode": "VALIDATION_SUCCESS",
            "statusMessage": "The validation succeeded.",
            "type": "VALIDATE_SECURITY_GROUP_ASSOCIATION"
        }
    ]
}

```

Para obtener más información, consulte [Creación de perfiles de lanzamiento](#) en la Guía del usuario de Amazon Nimble Studio.

- Para API obtener más información, consulte [GetLaunchProfile](#) la Referencia de AWS CLI comandos.



## get-studio

En el siguiente ejemplo de código, se muestra cómo usar `get-studio`.

### AWS CLI

Para obtener información sobre tu estudio

En el siguiente `get-studio` ejemplo, se muestran los estudios de tu AWS cuenta.

```
aws nimble get-studio \  
  --studio-id "StudioID"
```

Salida:

```
{  
  "studio": {  
    "adminRoleArn": "arn:aws:iam::123456789012:role/studio-admin-role",  
    "arn": "arn:aws:nimble:us-west-2:123456789012:studio/stid-EXAMPLE11111",  
    "createdAt": "2022-01-27T20:29:35+00:00",  
    "displayName": "studio-name",  
    "homeRegion": "us-west-2",  
    "ssoClientId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "state": "READY",  
    "statusCode": "STUDIO_CREATED",  
    "statusMessage": "The studio has been created successfully ",  
    "studioEncryptionConfiguration": {  
      "keyType": "AWS_OWNED_KEY"  
    },  
    "studioId": "us-west-2:stid-EXAMPLE11111",  
    "studioName": "studio-name",  
    "studioUrl": "https://studio-name.nimblestudio.us-west-2.amazonaws.com",  
    "tags": {},  
    "updatedAt": "2022-01-27T20:29:37+00:00",  
    "userRoleArn": "arn:aws:iam::123456789012:role/studio-user-role"  
  }  
}
```

Para obtener más información, consulte [¿Qué es Amazon Nimble Studio?](#) en la Guía del usuario de Amazon Nimble Studio.

- Para API obtener más información, consulte [GetStudio](#) la Referencia de AWS CLI comandos.

## list-eula-acceptances

En el siguiente ejemplo de código, se muestra cómo usar `list-eula-acceptances`.

### AWS CLI

Para ver una lista de los widgets disponibles

En el siguiente `list-eula-acceptances` ejemplo, se enumeran los aceptados EULAs en su AWS cuenta.

```
aws nimble list-eula-acceptances \  
  --studio-id "StudioID"
```

Salida:

```
{  
  "eulaAcceptances": [  
    {  
      "acceptedAt": "2022-01-28T17:44:35+00:00",  
      "acceptedBy": "92677b4b19-e9fd012a-94ad-4f16-9866-c69a63ab6486",  
      "accepteeId": "us-west-2:stid-nyoqq12fteqy1x48",  
      "eulaAcceptanceId": "V0JlpZQaSx6yHcUuX0qfQw",  
      "eulaId": "Rl-J0fM5S12hyIiwWIV6hw"  
    },  
    {  
      "acceptedAt": "2022-01-28T17:44:35+00:00",  
      "acceptedBy": "92677b4b19-e9fd012a-94ad-4f16-9866-c69a63ab6486",  
      "accepteeId": "us-west-2:stid-nyoqq12fteqy1x48",  
      "eulaAcceptanceId": "YY_uDFW-SVibc627qbug0Q",  
      "eulaId": "RvoNmVXiSrS4LhLTb6ybkw"  
    },  
    {  
      "acceptedAt": "2022-01-28T17:44:35+00:00",  
      "acceptedBy": "92677b4b19-e9fd012a-94ad-4f16-9866-c69a63ab6486",  
      "accepteeId": "us-west-2:stid-nyoqq12fteqy1x48",  
      "eulaAcceptanceId": "ov087PnhQ4-MpttiL5uN6Q",  
      "eulaId": "a-D9Wc0VQCKUfxAinCDxaw"  
    },  
    {  
      "acceptedAt": "2022-01-28T17:44:35+00:00",  
      "acceptedBy": "92677b4b19-e9fd012a-94ad-4f16-9866-c69a63ab6486",  
      "accepteeId": "us-west-2:stid-nyoqq12fteqy1x48",  
      "eulaAcceptanceId": "ov087PnhQ4-MpttiL5uN6Q",  
      "eulaId": "a-D9Wc0VQCKUfxAinCDxaw"  
    }  
  ]  
}
```

```

        "eulaAcceptanceId": "5YeXje4yR0amuTESGvqIAQ",
        "eulaId": "gJZLygd-Srq_5NNbSfiaLg"
    },
    {
        "acceptedAt": "2022-01-28T17:44:35+00:00",
        "acceptedBy": "92677b4b19-e9fd012a-94ad-4f16-9866-c69a63ab6486",
        "accepteeId": "us-west-2:stid-nyoqq12fteqy1x48",
        "eulaAcceptanceId": "W1sIn8PtScqeJEn8sxxhgw",
        "eulaId": "ggK2eIw6RQyt8PIee01D3g"
    },
    {
        "acceptedAt": "2022-01-28T17:44:35+00:00",
        "acceptedBy": "92677b4b19-e9fd012a-94ad-4f16-9866-c69a63ab6486",
        "accepteeId": "us-west-2:stid-nyoqq12fteqy1x48",
        "eulaAcceptanceId": "Zq9KNEQPRMWJ7FolSoQgUA",
        "eulaId": "wtp85BcSTa2NZeNRnMKdjw"
    }
]
}

```

Para obtener más información, consulte [Aceptar el EULA en la](#) Guía del usuario de Amazon Nimble Studio.

- Para API obtener más información, consulte [ListEulaAcceptances](#) la Referencia de AWS CLI comandos.

## list-eulas

En el siguiente ejemplo de código, se muestra cómo usar `list-eulas`.

### AWS CLI

Para ver una lista de los widgets disponibles

En el siguiente `list-eulas` ejemplo, se muestran EULAs los de su AWS cuenta.

```
aws nimble list-eulas
```

Salida:

```

{
  "eulas": [
    {

```

```
    "content": "https://www.mozilla.org/en-US/MPL/2.0/",
    "createdAt": "2021-04-20T16:45:23+00:00",
    "eulaId": "gJZLygd-Srq_5NNbSfiaLg",
    "name": "Mozilla-FireFox",
    "updatedAt": "2021-04-20T16:45:23+00:00"
  },
  {
    "content": "https://www.awsthinkbox.com/end-user-license-agreement",
    "createdAt": "2021-04-20T16:45:24+00:00",
    "eulaId": "RvoNmVXiSrS4LhLTb6ybkw",
    "name": "Thinkbox-Deadline",
    "updatedAt": "2021-04-20T16:45:24+00:00"
  },
  {
    "content": "https://www.videolan.org/legal.html",
    "createdAt": "2021-04-20T16:45:24+00:00",
    "eulaId": "R1-J0fM5S12hyIiwWIV6hw",
    "name": "Videolan-VLC",
    "updatedAt": "2021-04-20T16:45:24+00:00"
  },
  {
    "content": "https://code.visualstudio.com/license",
    "createdAt": "2021-04-20T16:45:23+00:00",
    "eulaId": "ggK2eIw6RQyt8PIee0lD3g",
    "name": "Microsoft-VSCode",
    "updatedAt": "2021-04-20T16:45:23+00:00"
  },
  {
    "content": "https://darbyjohnston.github.io/DJV/legal.html#License",
    "createdAt": "2021-04-20T16:45:23+00:00",
    "eulaId": "wtp85BcSTa2NZeNRnMKdjw",
    "name": "DJV-DJV",
    "updatedAt": "2021-04-20T16:45:23+00:00"
  },
  {
    "content": "https://www.sidefx.com/legal/license-agreement/",
    "createdAt": "2021-04-20T16:45:24+00:00",
    "eulaId": "uu2VDLo-QJeIGWwLBae_UA",
    "name": "SideFX-Houdini",
    "updatedAt": "2021-04-20T16:45:24+00:00"
  },
  {
    "content": "https://www.chaosgroup.com/eula",
    "createdAt": "2021-04-20T16:45:23+00:00",
```

```

    "eulaId": "L0HS4P3CRYKVXc2J2L07Vw",
    "name": "ChaosGroup-Vray",
    "updatedAt": "2021-04-20T16:45:23+00:00"
  },
  {
    "content": "https://www.foundry.com/eula",
    "createdAt": "2021-04-20T16:45:23+00:00",
    "eulaId": "SAuhfHmMSAeUuq3wsMiMlw",
    "name": "Foundry-Nuke",
    "updatedAt": "2021-04-20T16:45:23+00:00"
  },
  {
    "content": "https://download.blender.org/release/GPL3-license.txt",
    "createdAt": "2021-04-20T16:45:23+00:00",
    "eulaId": "a-D9Wc0VQCKUfxAinCDxaw",
    "name": "BlenderFoundation-Blender",
    "updatedAt": "2021-04-20T16:45:23+00:00"
  }
]
}

```

Para obtener más información, consulte [Aceptar el EULA en la](#) Guía del usuario de Amazon Nimble Studio.

- Para API obtener más información, consulte [ListEulas](#) la Referencia de AWS CLI comandos.

## list-launch-profiles

En el siguiente ejemplo de código, se muestra cómo usar `list-launch-profiles`.

### AWS CLI

Para ver una lista de los widgets disponibles

En el siguiente `list-launch-profiles` ejemplo, se enumeran los perfiles de lanzamiento de su AWS cuenta.

```
aws nimble list-launch-profiles \
  --studio-id "StudioID"
```

Salida:

```
{
```

```
"launchProfiles": [
  {
    "arn": "arn:aws:nimble:us-west-2:123456789012:launch-profile/
yeG7lDwNQEiwNTRT7DrV7Q",
    "createdAt": "2022-01-27T21:18:59+00:00",
    "createdBy": "AROA3002NEHCCYRNDDIFT:i-EXAMPLE11111",
    "description": "The Launch Profile for the Render workers created by
StudioBuilder.",
    "ec2SubnetIds": [
      "subnet-EXAMPLE11111"
    ],
    "launchProfileId": "yeG7lDwNQEiwNTRT7DrV7Q",
    "launchProfileProtocolVersions": [
      "2021-03-31"
    ],
    "name": "RenderWorker-Default",
    "state": "READY",
    "statusCode": "LAUNCH_PROFILE_CREATED",
    "statusMessage": "Launch Profile has been created",
    "streamConfiguration": {
      "clipboardMode": "ENABLED",
      "ec2InstanceTypes": [
        "g4dn.4xlarge",
        "g4dn.8xlarge"
      ],
      "maxSessionLengthInMinutes": 690,
      "maxStoppedSessionLengthInMinutes": 0,
      "streamingImageIds": [
        "Cw_jXnp1QcSSXhE2hkNRoQ",
        "YGXAqgoWTnCNSV8VP20sHQ"
      ]
    },
    "studioComponentIds": [
      "_hR_-RaAReS0jAnLakbX7Q",
      "vQ5w_TbIRayPkAZgcbYRA",
      "ZQuMxN99Qfa_Js6ma9TwdA",
      "45Kj0SPPRzK20yvpCuQ6qw"
    ],
    "tags": {},
    "updatedAt": "2022-01-27T21:19:13+00:00",
    "updatedBy": "AROA3002NEHCCYRNDDIFT:i-EXAMPLE11111",
    "validationResults": [
      {
        "state": "VALIDATION_SUCCESS",
```

```

        "statusCode": "VALIDATION_SUCCESS",
        "statusMessage": "The validation succeeded.",
        "type": "VALIDATE_ACTIVE_DIRECTORY_STUDIO_COMPONENT"
    },
    {
        "state": "VALIDATION_SUCCESS",
        "statusCode": "VALIDATION_SUCCESS",
        "statusMessage": "The validation succeeded.",
        "type": "VALIDATE_SUBNET_ASSOCIATION"
    },
    {
        "state": "VALIDATION_SUCCESS",
        "statusCode": "VALIDATION_SUCCESS",
        "statusMessage": "The validation succeeded.",
        "type": "VALIDATE_NETWORK_ACL_ASSOCIATION"
    },
    {
        "state": "VALIDATION_SUCCESS",
        "statusCode": "VALIDATION_SUCCESS",
        "statusMessage": "The validation succeeded.",
        "type": "VALIDATE_SECURITY_GROUP_ASSOCIATION"
    }
]
},
{
    "arn": "arn:aws:nimble:us-west-2:123456789012:launch-profile/
jDCIm1jRSaa9e44PZ3w7gg",
    "createdAt": "2022-01-27T21:19:26+00:00",
    "createdBy": "AROA3002NEHCCYRNDDIFT:i-EXAMPLE11111",
    "description": "This Workstation Launch Profile was created by
StudioBuilder",
    "ec2SubnetIds": [
        "subnet-046f4205ae535b2cc"
    ],
    "launchProfileId": "jDCIm1jRSaa9e44PZ3w7gg",
    "launchProfileProtocolVersions": [
        "2021-03-31"
    ],
    "name": "Workstation-Default",
    "state": "READY",
    "statusCode": "LAUNCH_PROFILE_CREATED",
    "statusMessage": "Launch Profile has been created",
    "streamConfiguration": {
        "clipboardMode": "ENABLED",

```

```
    "ec2InstanceTypes": [
      "g4dn.4xlarge",
      "g4dn.8xlarge"
    ],
    "maxSessionLengthInMinutes": 690,
    "maxStoppedSessionLengthInMinutes": 0,
    "streamingImageIds": [
      "Cw_jXnp1QcSSXhE2hkNRoQ",
      "YGXAqgoWTnCNSV8VP20sHQ"
    ]
  },
  "studioComponentIds": [
    "_hR_-RaAReS0jAnLakbX7Q",
    "vQ5w_TbIRayPkAZgcbyYRA",
    "ZQuMxN99Qfa_Js6ma9TwdA",
    "yJSbsHXAQYwk9FXLNusX1Q",
    "45Kj0SPPRzK20yvpCuQ6qw"
  ],
  "tags": {},
  "updatedAt": "2022-01-27T21:19:40+00:00",
  "updatedBy": "AR0A3002NEHCCYRNDDIFT:i-EXAMPLE11111",
  "validationResults": [
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_ACTIVE_DIRECTORY_STUDIO_COMPONENT"
    },
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_SUBNET_ASSOCIATION"
    },
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",
      "type": "VALIDATE_NETWORK_ACL_ASSOCIATION"
    },
    {
      "state": "VALIDATION_SUCCESS",
      "statusCode": "VALIDATION_SUCCESS",
      "statusMessage": "The validation succeeded.",

```



```

    "type": "VALIDATE_SECURITY_GROUP_ASSOCIATION"
  }
]
}

```

Para obtener más información, consulte [Creación de perfiles de lanzamiento](#) en la Guía del usuario de Amazon Nimble Studio.

- Para API obtener más información, consulte [ListLaunchProfiles](#) la Referencia de AWS CLI comandos.

## list-studio-components

En el siguiente ejemplo de código, se muestra cómo usar `list-studio-components`.

### AWS CLI

Para ver una lista de los widgets disponibles

En el siguiente `list-studio-components` ejemplo, se muestran los componentes del estudio de tu AWS cuenta.

```

aws nimble list-studio-components \
  --studio-id "StudioID"

```

Salida:

```

{
  "studioComponents": [
    {
      "arn": "arn:aws:nimble:us-west-2:123456789012:studio-component/
ZQuMxN99Qfa Js6ma9TwdA",
      "configuration": {
        "sharedFileSystemConfiguration": {
          "fileSystemId": "fs-EXAMPLE11111",
          "linuxMountPoint": "/mnt/fsxshare",
          "shareName": "share",
          "windowsMountDrive": "Z"
        }
      }
    },
  ],
}

```

```

    "createdAt": "2022-01-27T21:15:34+00:00",
    "createdBy": "AROAZ002NEHCCYRNDDIFT:i-EXAMPLE11111",
    "description": "FSx for Windows",
    "ec2SecurityGroupIds": [
      "sg-EXAMPLE11111"
    ],
    "name": "FSxWindows",
    "state": "READY",
    "statusCode": "STUDIO_COMPONENT_CREATED",
    "statusMessage": "Studio Component has been created",
    "studioComponentId": "ZQuMxN99Qfa_Js6ma9TwdA",
    "subtype": "AMAZON_FSX_FOR_WINDOWS",
    "tags": {},
    "type": "SHARED_FILE_SYSTEM",
    "updatedAt": "2022-01-27T21:15:35+00:00",
    "updatedBy": "AROAZ002NEHCCYRNDDIFT:i-EXAMPLE11111"
  },
  ...
}

```

Para obtener más información, consulte [Cómo StudioBuilder funciona con Amazon Nimble Studio](#) en la Guía del usuario de Amazon Nimble Studio.

- Para API obtener más información, consulte la Referencia [ListStudioComponents](#) de AWS CLI comandos.

## list-studio-members

En el siguiente ejemplo de código, se muestra cómo usar `list-studio-members`.

### AWS CLI

Para ver una lista de los widgets disponibles

En el siguiente `list-studio-members` ejemplo, se muestran los miembros del estudio disponibles en tu AWS cuenta.

```
aws nimble list-studio-members \
  --studio-id "StudioID"
```

Salida:

```
{
```

```
"members": [  
  {  
    "identityStoreId": "d-EXAMPLE11111",  
    "persona": "ADMINISTRATOR",  
    "principalId": "EXAMPLE11111-e9fd012a-94ad-4f16-9866-c69a63ab6486"  
  }  
]  
}
```

Para obtener más información, consulte [Añadir usuarios de un estudio](#) en la Guía del usuario de Amazon Nimble Studio.

- Para API obtener más información, consulte [ListStudioMembers](#) la Referencia de AWS CLI comandos.

## list-studios

En el siguiente ejemplo de código, se muestra cómo usar `list-studios`.

### AWS CLI

Para enumerar tus estudios

En el siguiente `list-studios` ejemplo, se muestran los estudios de tu AWS cuenta.

```
aws nimble list-studios
```

Salida:

```
{  
  "studios": [  
    {  
      "adminRoleArn": "arn:aws:iam::123456789012:role/studio-admin-role",  
      "arn": "arn:aws:nimble:us-west-2:123456789012:studio/studio-id",  
      "createdAt": "2022-01-27T20:29:35+00:00",  
      "displayName": "studio-name",  
      "homeRegion": "us-west-2",  
      "ssoClientId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "state": "READY",  
      "statusCode": "STUDIO_CREATED",  
      "statusMessage": "The studio has been created successfully ",  
      "studioEncryptionConfiguration": {
```

```
        "keyType": "AWS_OWNED_KEY"
      },
      "studioId": "us-west-2:studio-id",
      "studioName": "studio-name",
      "studioUrl": "https://studio-name.nimblestudio.us-west-2.amazonaws.com",
      "tags": {},
      "updatedAt": "2022-01-27T20:29:37+00:00",
      "userRoleArn": "arn:aws:iam::123456789012:role/studio-user-role"
    }
  ]
}
```

Para obtener más información, consulte [¿Qué es Amazon Nimble Studio?](#) en la Guía del usuario de Amazon Nimble Studio.

- Para API obtener más información, consulte [ListStudios](#) la Referencia de AWS CLI comandos.

## OpenSearch Ejemplos de servicios que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante el uso del OpenSearch servicio AWS Command Line Interface with.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

#### **create-elasticsearch-domain**

En el siguiente ejemplo de código, se muestra cómo usar `create-elasticsearch-domain`.

#### AWS CLI

Para crear un dominio de Amazon Elasticsearch Service

El siguiente `create-elasticsearch-domain` comando crea un nuevo dominio de Amazon Elasticsearch Service dentro de un VPC y restringe el acceso a un único usuario. Amazon ES deduce el VPC ID de la subred y el grupo de seguridad especificados. IDs

```
aws es create-elasticsearch-domain \
  --domain-name vpc-cli-example \
  --elasticsearch-version 6.2 \
  --elasticsearch-cluster-
config InstanceType=m4.large.elasticsearch,InstanceCount=1 \
  --ebs-options EBSEnabled=true,VolumeType=standard,VolumeSize=10 \
  --access-policies '{"Version": "2012-10-17", "Statement": [ { "Effect":
"Allow", "Principal": {"AWS": "arn:aws:iam::123456789012:root" }, "Action": "es:*",
"Resource": "arn:aws:es:us-west-1:123456789012:domain/vpc-cli-example/*" } ] }' \
  --vpc-options SubnetIds=subnet-1a2a3a4a,SecurityGroupIds=sg-2a3a4a5a
```

Salida:

```
{
  "DomainStatus": {
    "ElasticsearchClusterConfig": {
      "DedicatedMasterEnabled": false,
      "InstanceCount": 1,
      "ZoneAwarenessEnabled": false,
      "InstanceType": "m4.large.elasticsearch"
    },
    "DomainId": "123456789012/vpc-cli-example",
    "CognitoOptions": {
      "Enabled": false
    },
    "VPCOptions": {
      "SubnetIds": [
        "subnet-1a2a3a4a"
      ],
      "VPCId": "vpc-3a4a5a6a",
      "SecurityGroupIds": [
        "sg-2a3a4a5a"
      ],
      "AvailabilityZones": [
        "us-west-1c"
      ]
    },
    "Created": true,
    "Deleted": false,
```

```

    "EBSOptions": {
      "VolumeSize": 10,
      "VolumeType": "standard",
      "EBSEnabled": true
    },
    "Processing": true,
    "DomainName": "vpc-cli-example",
    "SnapshotOptions": {
      "AutomatedSnapshotStartHour": 0
    },
    "ElasticsearchVersion": "6.2",
    "AccessPolicies": "{\n\"Version\": \"2012-10-17\", \"Statement\": [{\n\"Effect\":\n\n\"Allow\", \"Principal\": {\n\"AWS\": \"arn:aws:iam::123456789012:root\"}, \"Action\":\n\n\"es:*\", \"Resource\": \"arn:aws:es:us-west-1:123456789012:domain/vpc-cli-example/*\n\n\"}]}",
    "AdvancedOptions": {
      "rest.action.multi.allow_explicit_index": "true"
    },
    "EncryptionAtRestOptions": {
      "Enabled": false
    },
    "ARN": "arn:aws:es:us-west-1:123456789012:domain/vpc-cli-example"
  }
}

```

Para obtener más información, consulte [Creación y administración de dominios de Amazon Elasticsearch Service](#) en la Guía para desarrolladores de Amazon Elasticsearch Service.

- Para obtener API más información, consulte la Referencia de comandos [CreateElasticsearchDomain](#).AWS CLI

## describe-elasticsearch-domain-config

En el siguiente ejemplo de código, se muestra cómo usar `describe-elasticsearch-domain-config`.

### AWS CLI

Para obtener los detalles de configuración del dominio

El siguiente `describe-elasticsearch-domain-config` ejemplo proporciona los detalles de configuración de un dominio determinado, junto con la información de estado de cada componente del dominio individual.

```
aws es describe-elasticsearch-domain-config \  
--domain-name cli-example
```

Salida:

```
{  
  "DomainConfig": {  
    "ElasticsearchVersion": {  
      "Options": "7.4",  
      "Status": {  
        "CreationDate": 1589395034.946,  
        "UpdateDate": 1589395827.325,  
        "UpdateVersion": 8,  
        "State": "Active",  
        "PendingDeletion": false  
      }  
    },  
    "ElasticsearchClusterConfig": {  
      "Options": {  
        "InstanceType": "c5.large.elasticsearch",  
        "InstanceCount": 1,  
        "DedicatedMasterEnabled": true,  
        "ZoneAwarenessEnabled": false,  
        "DedicatedMasterType": "c5.large.elasticsearch",  
        "DedicatedMasterCount": 3,  
        "WarmEnabled": true,  
        "WarmType": "ultrawarm1.medium.elasticsearch",  
        "WarmCount": 2  
      },  
      "Status": {  
        "CreationDate": 1589395034.946,  
        "UpdateDate": 1589395827.325,  
        "UpdateVersion": 8,  
        "State": "Active",  
        "PendingDeletion": false  
      }  
    },  
    "EBSOptions": {  
      "Options": {  
        "EBSEnabled": true,  
        "VolumeType": "gp2",  
        "VolumeSize": 10  
      }  
    }  
  }  
}
```

```
    "Status": {
      "CreationDate": 1589395034.946,
      "UpdateDate": 1589395827.325,
      "UpdateVersion": 8,
      "State": "Active",
      "PendingDeletion": false
    }
  },
  "AccessPolicies": {
    "Options": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"*\"},\"Action\":\"es:*\",\"Resource\":\"arn:aws:es:us-east-1:123456789012:domain/cli-example/*\"}]}",
    "Status": {
      "CreationDate": 1589395034.946,
      "UpdateDate": 1589395827.325,
      "UpdateVersion": 8,
      "State": "Active",
      "PendingDeletion": false
    }
  },
  "SnapshotOptions": {
    "Options": {
      "AutomatedSnapshotStartHour": 0
    },
    "Status": {
      "CreationDate": 1589395034.946,
      "UpdateDate": 1589395827.325,
      "UpdateVersion": 8,
      "State": "Active",
      "PendingDeletion": false
    }
  },
  "VPCOptions": {
    "Options": {},
    "Status": {
      "CreationDate": 1591210426.162,
      "UpdateDate": 1591210426.162,
      "UpdateVersion": 18,
      "State": "Active",
      "PendingDeletion": false
    }
  },
  "CognitoOptions": {
    "Options": {
```



```
        "Enabled": false
    },
    "Status": {
        "CreationDate": 1591210426.163,
        "UpdateDate": 1591210426.163,
        "UpdateVersion": 18,
        "State": "Active",
        "PendingDeletion": false
    }
},
"EncryptionAtRestOptions": {
    "Options": {
        "Enabled": true,
        "KmsKeyId": "arn:aws:kms:us-
east-1:123456789012:key/1a2a3a4a-1a2a-1a2a-1a2a-1a2a3a4a5a6a"
    },
    "Status": {
        "CreationDate": 1589395034.946,
        "UpdateDate": 1589395827.325,
        "UpdateVersion": 8,
        "State": "Active",
        "PendingDeletion": false
    }
},
"NodeToNodeEncryptionOptions": {
    "Options": {
        "Enabled": true
    },
    "Status": {
        "CreationDate": 1589395034.946,
        "UpdateDate": 1589395827.325,
        "UpdateVersion": 8,
        "State": "Active",
        "PendingDeletion": false
    }
},
"AdvancedOptions": {
    "Options": {
        "rest.action.multi.allow_explicit_index": "true"
    },
    "Status": {
        "CreationDate": 1589395034.946,
        "UpdateDate": 1589395827.325,
        "UpdateVersion": 8,
```

```
        "State": "Active",
        "PendingDeletion": false
    }
},
"LogPublishingOptions": {
    "Options": {},
    "Status": {
        "CreationDate": 1591210426.164,
        "UpdateDate": 1591210426.164,
        "UpdateVersion": 18,
        "State": "Active",
        "PendingDeletion": false
    }
},
"DomainEndpointOptions": {
    "Options": {
        "EnforceHTTPS": true,
        "TLSSecurityPolicy": "Policy-Min-TLS-1-0-2019-07"
    },
    "Status": {
        "CreationDate": 1589395034.946,
        "UpdateDate": 1589395827.325,
        "UpdateVersion": 8,
        "State": "Active",
        "PendingDeletion": false
    }
},
"AdvancedSecurityOptions": {
    "Options": {
        "Enabled": true,
        "InternalUserDatabaseEnabled": true
    },
    "Status": {
        "CreationDate": 1589395034.946,
        "UpdateDate": 1589827485.577,
        "UpdateVersion": 14,
        "State": "Active",
        "PendingDeletion": false
    }
}
}
```

Para obtener más información, consulte [Creación y administración de dominios de Amazon Elasticsearch Service](#) en la Guía para desarrolladores de Amazon Elasticsearch Service.

- Para obtener API más información, consulte la Referencia de comandos [DescribeElasticsearchDomainConfig](#).AWS CLI

## **describe-elasticsearch-domain**

En el siguiente ejemplo de código, se muestra cómo usar `describe-elasticsearch-domain`.

### AWS CLI

Para obtener detalles de un solo dominio

En el siguiente `describe-elasticsearch-domain` ejemplo, se proporcionan los detalles de configuración de un dominio determinado.

```
aws es describe-elasticsearch-domain \  
  --domain-name cli-example
```

Salida:

```
{  
  "DomainStatus": {  
    "DomainId": "123456789012/cli-example",  
    "DomainName": "cli-example",  
    "ARN": "arn:aws:es:us-east-1:123456789012:domain/cli-example",  
    "Created": true,  
    "Deleted": false,  
    "Endpoint": "search-cli-example-1a2a3a4a5a6a7a8a9a0a.us-  
east-1.es.amazonaws.com",  
    "Processing": false,  
    "UpgradeProcessing": false,  
    "ElasticsearchVersion": "7.4",  
    "ElasticsearchClusterConfig": {  
      "InstanceType": "c5.large.elasticsearch",  
      "InstanceCount": 1,  
      "DedicatedMasterEnabled": true,  
      "ZoneAwarenessEnabled": false,  
      "DedicatedMasterType": "c5.large.elasticsearch",  
      "DedicatedMasterCount": 3,  
      "WarmEnabled": true,  
      "WarmType": "ultrawarm1.medium.elasticsearch",
```

```
    "WarmCount": 2
  },
  "EBSOptions": {
    "EBSEnabled": true,
    "VolumeType": "gp2",
    "VolumeSize": 10
  },
  "AccessPolicies": [{"Version": "2012-10-17", "Statement": [{"Effect": "Allow", "Principal": {"AWS": "*"}, "Action": "es:*", "Resource": "arn:aws:es:us-east-1:123456789012:domain/cli-example/*"}]}],
  "SnapshotOptions": {
    "AutomatedSnapshotStartHour": 0
  },
  "CognitoOptions": {
    "Enabled": false
  },
  "EncryptionAtRestOptions": {
    "Enabled": true,
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/1a2a3a4a-1a2a-1a2a-1a2a-1a2a3a4a5a6a"
  },
  "NodeToNodeEncryptionOptions": {
    "Enabled": true
  },
  "AdvancedOptions": {
    "rest.action.multi.allow_explicit_index": "true"
  },
  "ServiceSoftwareOptions": {
    "CurrentVersion": "R20200522",
    "NewVersion": "",
    "UpdateAvailable": false,
    "Cancellable": false,
    "UpdateStatus": "COMPLETED",
    "Description": "There is no software update available for this domain.",
    "AutomatedUpdateDate": 0.0
  },
  "DomainEndpointOptions": {
    "EnforceHTTPS": true,
    "TLSSecurityPolicy": "Policy-Min-TLS-1-0-2019-07"
  },
  "AdvancedSecurityOptions": {
    "Enabled": true,
    "InternalUserDatabaseEnabled": true
  }
}
```

```
}  
}
```

Para obtener más información, consulte [Creación y administración de dominios de Amazon Elasticsearch Service](#) en la Guía para desarrolladores de Amazon Elasticsearch Service.

- Para obtener API más información, consulte la Referencia de comandos [DescribeElasticsearchDomain](#).AWS CLI

## describe-elasticsearch-domains

En el siguiente ejemplo de código, se muestra cómo usar `describe-elasticsearch-domains`.

AWS CLI

Para obtener detalles de uno o más dominios

El siguiente `describe-elasticsearch-domains` ejemplo proporciona detalles de configuración para uno o más dominios.

```
aws es describe-elasticsearch-domains \  
  --domain-names cli-example-1 cli-example-2
```

Salida:

```
{  
  "DomainStatusList": [{  
    "DomainId": "123456789012/cli-example-1",  
    "DomainName": "cli-example-1",  
    "ARN": "arn:aws:es:us-east-1:123456789012:domain/cli-example-1",  
    "Created": true,  
    "Deleted": false,  
    "Endpoint": "search-cli-example-1-1a2a3a4a5a6a7a8a9a0a.us-  
east-1.es.amazonaws.com",  
    "Processing": false,  
    "UpgradeProcessing": false,  
    "ElasticsearchVersion": "7.4",  
    "ElasticsearchClusterConfig": {  
      "InstanceType": "c5.large.elasticsearch",  
      "InstanceCount": 1,  
      "DedicatedMasterEnabled": true,  
      "ZoneAwarenessEnabled": false,  
      "DedicatedMasterType": "c5.large.elasticsearch",
```

```

        "DedicatedMasterCount": 3,
        "WarmEnabled": true,
        "WarmType": "ultrawarm1.medium.elasticsearch",
        "WarmCount": 2
    },
    "EBSOptions": {
        "EBSEnabled": true,
        "VolumeType": "gp2",
        "VolumeSize": 10
    },
    "AccessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"*\"},\"Action\":\"es:*\",\"Resource\":\"arn:aws:es:us-east-1:123456789012:domain/cli-example-1/*\"}]}",
    "SnapshotOptions": {
        "AutomatedSnapshotStartHour": 0
    },
    "CognitoOptions": {
        "Enabled": false
    },
    "EncryptionAtRestOptions": {
        "Enabled": true,
        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/1a2a3a4a-1a2a-1a2a-1a2a-1a2a3a4a5a6a"
    },
    "NodeToNodeEncryptionOptions": {
        "Enabled": true
    },
    "AdvancedOptions": {
        "rest.action.multi.allow_explicit_index": "true"
    },
    "ServiceSoftwareOptions": {
        "CurrentVersion": "R20200522",
        "NewVersion": "",
        "UpdateAvailable": false,
        "Cancelable": false,
        "UpdateStatus": "COMPLETED",
        "Description": "There is no software update available for this domain.",
        "AutomatedUpdateDate": 0.0
    },
    "DomainEndpointOptions": {
        "EnforceHTTPS": true,
        "TLSSecurityPolicy": "Policy-Min-TLS-1-0-2019-07"
    },

```

```

    "AdvancedSecurityOptions": {
      "Enabled": true,
      "InternalUserDatabaseEnabled": true
    }
  },
  {
    "DomainId": "123456789012/cli-example-2",
    "DomainName": "cli-example-2",
    "ARN": "arn:aws:es:us-east-1:123456789012:domain/cli-example-2",
    "Created": true,
    "Deleted": false,
    "Processing": true,
    "UpgradeProcessing": false,
    "ElasticsearchVersion": "7.4",
    "ElasticsearchClusterConfig": {
      "InstanceType": "r5.large.elasticsearch",
      "InstanceCount": 1,
      "DedicatedMasterEnabled": false,
      "ZoneAwarenessEnabled": false,
      "WarmEnabled": false
    },
    "EBSOptions": {
      "EBSEnabled": true,
      "VolumeType": "gp2",
      "VolumeSize": 10
    },
    "AccessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Deny\",\"Principal\":{\"AWS\":\"*\"},\"Action\":\"es:*\",\"Resource\":\"arn:aws:es:us-east-1:123456789012:domain/cli-example-2/*\"}]}",
    "SnapshotOptions": {
      "AutomatedSnapshotStartHour": 0
    },
    "CognitoOptions": {
      "Enabled": false
    },
    "EncryptionAtRestOptions": {
      "Enabled": false
    },
    "NodeToNodeEncryptionOptions": {
      "Enabled": false
    },
    "AdvancedOptions": {
      "rest.action.multi.allow_explicit_index": "true"
    }
  },

```

```
    "ServiceSoftwareOptions": {
      "CurrentVersion": "",
      "NewVersion": "",
      "UpdateAvailable": false,
      "Cancellable": false,
      "UpdateStatus": "COMPLETED",
      "Description": "There is no software update available for this
domain.",
      "AutomatedUpdateDate": 0.0
    },
    "DomainEndpointOptions": {
      "EnforceHTTPS": false,
      "TLSSecurityPolicy": "Policy-Min-TLS-1-0-2019-07"
    },
    "AdvancedSecurityOptions": {
      "Enabled": false,
      "InternalUserDatabaseEnabled": false
    }
  }
]
}
```

Para obtener más información, consulte [Creación y administración de dominios de Amazon Elasticsearch Service](#) en la Guía para desarrolladores de Amazon Elasticsearch Service.

- Para obtener API más información, consulte la Referencia de comandos [DescribeElasticsearchDomains](#).AWS CLI

## **describe-reserved-elasticsearch-instances**

En el siguiente ejemplo de código, se muestra cómo usar `describe-reserved-elasticsearch-instances`.

AWS CLI

Para ver todas las instancias reservadas

El siguiente `describe-elasticsearch-domains` ejemplo proporciona un resumen de todas las instancias que ha reservado en una región.

```
aws es describe-reserved-elasticsearch-instances
```



**Salida:**

```
{
  "ReservedElasticsearchInstances": [{
    "FixedPrice": 100.0,
    "ReservedElasticsearchInstanceOfferingId":
"1a2a3a4a5-1a2a-3a4a-5a6a-1a2a3a4a5a6a",
    "ReservationName": "my-reservation",
    "PaymentOption": "PARTIAL_UPFRONT",
    "UsagePrice": 0.0,
    "ReservedElasticsearchInstanceId": "9a8a7a6a-5a4a-3a2a-1a0a-9a8a7a6a5a4a",
    "RecurringCharges": [{
      "RecurringChargeAmount": 0.603,
      "RecurringChargeFrequency": "Hourly"
    }],
    "State": "payment-pending",
    "StartTime": 1522872571.229,
    "ElasticsearchInstanceCount": 3,
    "Duration": 31536000,
    "ElasticsearchInstanceType": "m4.2xlarge.elasticsearch",
    "CurrencyCode": "USD"
  ]
}
```

Para obtener más información, consulte [Instancias reservadas](#) en la Guía para desarrolladores de Amazon Elasticsearch Service.

- Para API obtener más información, consulte la Referencia [DescribeReservedElasticsearchInstances](#) de AWS CLI comandos.

**list-domain-names**

En el siguiente ejemplo de código, se muestra cómo usar `list-domain-names`.

**AWS CLI**

Para enumerar todos los dominios

El siguiente `list-domain-names` ejemplo proporciona un resumen rápido de todos los dominios de la región.

```
aws es list-domain-names
```

Salida:

```
{
  "DomainNames": [{
    "DomainName": "cli-example-1"
  },
  {
    "DomainName": "cli-example-2"
  }
]
```

Para obtener más información, consulte [Creación y administración de dominios de Amazon Elasticsearch Service](#) en la Guía para desarrolladores de Amazon Elasticsearch Service.

- Para obtener API más información, consulte la Referencia de comandos [ListDomainNames](#).AWS CLI

## AWS OpsWorks ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS OpsWorks.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **assign-instance**

En el siguiente ejemplo de código, se muestra cómo usar `assign-instance`.

## AWS CLI

Para asignar una instancia registrada a una capa

El siguiente ejemplo asigna una instancia registrada a una capa personalizada.

```
aws opsworks --region us-east-1 assign-instance --instance-id 4d6d1710-ded9-42a1-b08e-b043ad7af1e2 --layer-ids 26cf1d32-6876-42fa-bbf1-9cad0bff938
```

Salida: ninguna.

### Más información

Para obtener más información, consulte [Asignación de una instancia registrada a una capa](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [AssignInstance](#) la Referencia de AWS CLI comandos.

## assign-volume

En el siguiente ejemplo de código, se muestra cómo usar `assign-volume`.

## AWS CLI

Para asignar un volumen registrado a una instancia

En el siguiente ejemplo, se asigna un volumen registrado de Amazon Elastic Block Store (AmazonEBS) a una instancia. El volumen se identifica por su ID de volumen, que es el GUID que se AWS OpsWorks asigna al registrar el volumen en una pila, no por el ID de volumen de Amazon Elastic Compute Cloud (AmazonEC2). Antes de ejecutar `assign-volume`, primero debe correr `update-volume` para asignar un punto de montaje al volumen.

```
aws opsworks --region us-east-1 assign-volume --instance-id 4d6d1710-ded9-42a1-b08e-b043ad7af1e2 --volume-id 26cf1d32-6876-42fa-bbf1-9cad0bff938
```

Salida: ninguna.

### Más información

Para obtener más información, consulte [Asignación de EBS volúmenes de Amazon a una instancia](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [AssignVolume](#) la Referencia de AWS CLI comandos.

## **associate-elastic-ip**

En el siguiente ejemplo de código, se muestra cómo usar `associate-elastic-ip`.

### AWS CLI

Para asociar una dirección IP elástica a una instancia

En el siguiente ejemplo, se asocia una dirección IP elástica a una instancia específica.

```
aws opsworks --region us-east-1 associate-elastic-ip --instance-id dfc18b02-5327-493d-91a4-c5c0c448927f --elastic-ip 54.148.130.96
```

Salida: ninguna.

### Más información

Para obtener más información, consulte Administración de recursos en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [AssociateElasticIp](#) la Referencia de AWS CLI comandos.

## **attach-elastic-load-balancer**

En el siguiente ejemplo de código, se muestra cómo usar `attach-elastic-load-balancer`.

### AWS CLI

Para adjuntar un balanceador de carga a una capa

En el siguiente ejemplo, se adjunta un equilibrador de cargas, identificado por su nombre, a una capa específica.

```
aws opsworks --region us-east-1 attach-elastic-load-balancer --elastic-load-balancer-name Java-LB --layer-id 888c5645-09a5-4d0e-95a8-812ef1db76a4
```

Salida: ninguna.

## Más información

Para obtener más información, consulte Elastic Load Balancing en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [AttachElasticLoadBalancer](#) la Referencia de AWS CLI comandos.

## create-app

En el siguiente ejemplo de código, se muestra cómo usar create-app.

### AWS CLI

Ejemplo 1: Para crear una aplicación

En el siguiente ejemplo, se crea una PHP aplicación denominada SimplePHPApp a partir del código almacenado en un GitHub repositorio. El comando usa la forma abreviada de la definición fuente de la aplicación.

```
aws opsworks create-app \  
  --region us-east-1 \  
  --stack-id f6673d70-32e6-4425-8999-265dd002fec7 \  
  --name SimplePHPApp \  
  --type php \  
  --app-source Type=git,Url=git://github.com/amazonwebservices/opsworks-demo-php-simple-app.git,Revision=version1
```

Salida:

```
{  
  "AppId": "6cf5163c-a951-444f-a8f7-3716be75f2a2"  
}
```

Ejemplo 2: Para crear una aplicación con una base de datos adjunta

El siguiente ejemplo crea una JSP aplicación a partir del código almacenado en un archivo.zip en un bucket público de S3. Adjunta una RDS instancia de base de datos que sirve como almacén de datos de la aplicación. Las fuentes de la aplicación y de la base de datos se definen en JSON archivos independientes que se encuentran en el directorio desde el que se ejecuta el comando.

```
aws opsworks create-app \  
  --region us-east-1 \  
  --stack-id 8c428b08-a1a1-46ce-a5f8-feddc43771b8 \  
  --name SimpleJSP \  
  --type java \  
  --app-source file://appsource.json \  
  --data-sources file://datasource.json
```

La información del origen de la aplicación se encuentra en `appsource.json` y contiene lo siguiente.

```
{  
  "Type": "archive",  
  "Url": "https://s3.amazonaws.com/opsworks-demo-assets/simplejsp.zip"  
}
```

La información de origen de la base de datos está incluida `datasource.json` y contiene lo siguiente.

```
[  
  {  
    "Type": "RdsDbInstance",  
    "Arn": "arn:aws:rds:us-west-2:123456789012:db:clitestdb",  
    "DatabaseName": "mydb"  
  }  
]
```

Nota: En el caso de una RDS instancia de base de datos, primero debe utilizarla `register-rds-db-instance` para registrar la instancia en la pila. Para las instancias de My SQL App Server, Type establézcalo en `OpsworksMySQLInstance`. Estas instancias las crea AWS OpsWorks, por lo que no es necesario registrarlas.

Salida:

```
{  
  "AppId": "26a61ead-d201-47e3-b55c-2a7c666942f8"  
}
```

Para obtener más información, consulte [Añadir aplicaciones](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [CreateApp](#) la Referencia de AWS CLI comandos.

## create-deployment

En el siguiente ejemplo de código, se muestra cómo usar `create-deployment`.

### AWS CLI

Ejemplo 1: Para implementar aplicaciones y ejecutar comandos de pila

Los siguientes ejemplos muestran cómo usar el `create-deployment` comando para implementar aplicaciones y ejecutar comandos de pila. Observe que los caracteres entre comillas (") del JSON objeto que especifica el comando van todos precedidos de caracteres de escape (\). Sin los caracteres de escape, es posible que el comando devuelva un JSON error no válido.

En el siguiente `create-deployment` ejemplo, se implementa una aplicación en una pila específica.

```
aws opsworks create-deployment \  
  --stack-id cfb7e082-ad1d-4599-8e81-de1c39ab45bf \  
  --app-id 307be5c8-d55d-47b5-bd6e-7bd417c6c7eb \  
  --command "{\"Name\":\"deploy\"}"
```

Salida:

```
{  
  "DeploymentId": "5746c781-df7f-4c87-84a7-65a119880560"  
}
```

Ejemplo 2: Para implementar una aplicación Rails y migrar la base de datos

El siguiente `create-deployment` comando implementa una aplicación de Ruby on Rails en una pila específica y migra la base de datos.

```
aws opsworks create-deployment \  
  --stack-id cfb7e082-ad1d-4599-8e81-de1c39ab45bf \  
  --app-id 307be5c8-d55d-47b5-bd6e-7bd417c6c7eb \  
  --command "{\"Name\":\"deploy\", \"Args\":{\"migrate\":[\"true\"]}\"}"
```

Salida:

```
{
  "DeploymentId": "5746c781-df7f-4c87-84a7-65a119880560"
}
```

Para obtener más información sobre la implementación, consulte [Implementación de aplicaciones](#) en la Guía del AWS OpsWorks usuario.

### Ejemplo 3: ejecutar una receta

El siguiente `create-deployment` comando ejecuta una receta personalizada en las instancias de una pila especificada. `phpapp::appsetup`

```
aws opsworks create-deployment \
  --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb \
  --command "{\"Name\":\"execute_recipes\", \"Args\":{\"recipes\":\
  [\"phpapp::appsetup\"]}}"
```

Salida:

```
{
  "DeploymentId": "5cbaa7b9-4e09-4e53-aa1b-314fbd106038"
}
```

Para obtener más información, consulte [Ejecutar comandos de pila](#) en la Guía del AWS OpsWorks usuario.

### Ejemplo 4: Instalar dependencias

El siguiente `create-deployment` comando instala las dependencias, como paquetes o gemas de Ruby, en las instancias de una pila específica.

```
aws opsworks create-deployment \
  --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb \
  --command "{\"Name\":\"install_dependencies\"}"
```

Salida:

```
{
  "DeploymentId": "aef5b255-8604-4928-81b3-9b0187f962ff"
}
```



```
}
```

Para obtener más información, consulta la sección [Ejecutar comandos de pila](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [CreateDeployment](#) la Referencia de AWS CLI comandos.

## create-instance

En el siguiente ejemplo de código, se muestra cómo usar `create-instance`.

### AWS CLI

Para crear una instancia

El siguiente `create-instance` comando crea una instancia `m1.large` de Amazon Linux denominada `myinstance1` en una pila específica. La instancia se asigna a una capa.

```
aws opsworks --region us-east-1 create-instance --stack-id 935450cc-61e0-4b03-  
a3e0-160ac817d2bb --layer-ids 5c8c272a-f2d5-42e3-8245-5bf3927cb65b --  
hostname myinstance1 --instance-type m1.large --os "Amazon Linux"
```

Para usar un nombre generado automáticamente, llame `get-hostname-suggestion`, que genera un nombre de host basado en el tema que especificó al crear la pila. A continuación, pasa ese nombre al argumento `hostname`.

Salida:

```
{  
  "InstanceId": "5f9adeaa-c94c-42c6-aeef-28a5376002cd"  
}
```

### Más información

Para obtener más información, consulte [Añadir una instancia a una capa](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [CreateInstance](#) la Referencia de AWS CLI comandos.

## create-layer

En el siguiente ejemplo de código, se muestra cómo usar `create-layer`.

### AWS CLI

Para crear una capa

El siguiente `create-layer` comando crea una capa de PHP App Server denominada MyPHPLayer en una pila especificada.

```
aws opsworks create-layer --region us-east-1 --stack-id f6673d70-32e6-4425-8999-265dd002fec7 --type php-app --name MyPHPLayer --shortname myphpLayer
```

Salida:

```
{
  "LayerId": "0b212672-6b4b-40e4-8a34-5a943cf2e07a"
}
```

### Más información

Para obtener más información, consulte [Cómo crear una capa](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [CreateLayer](#) la Referencia de AWS CLI comandos.

## create-server

En el siguiente ejemplo de código, se muestra cómo usar `create-server`.

### AWS CLI

Para crear un servidor

El siguiente `create-server` ejemplo crea un nuevo servidor de Chef Automate con `automate-06` el nombre de su región predeterminada. Tenga en cuenta que los valores predeterminados se utilizan para la mayoría de las demás configuraciones, como el número de copias de seguridad que se deben conservar y las horas de inicio y mantenimiento de las copias

de seguridad. Antes de ejecutar un `create-server` comando, complete los requisitos previos de la [Guía del usuario de Opsworks AWS OpsWorks for Chef Automate](#) en la Guía del usuario de AWS Opsworks for Chef Automate.

```
aws opsworks-cm create-server \  
  --engine "ChefAutomate" \  
  --instance-profile-arn "arn:aws:iam::012345678901:instance-profile/aws-opsworks-  
cm-ec2-role" \  
  --instance-type "t2.medium" \  
  --server-name "automate-06" \  
  --service-role-arn "arn:aws:iam::012345678901:role/aws-opsworks-cm-service-role"
```

Salida:

```
{  
  "Server": {  
    "AssociatePublicIpAddress": true,  
    "BackupRetentionCount": 10,  
    "CreatedAt": 2019-12-29T13:38:47.520Z,  
    "DisableAutomatedBackup": FALSE,  
    "Endpoint": "https://opsworks-cm.us-east-1.amazonaws.com",  
    "Engine": "ChefAutomate",  
    "EngineAttributes": [  
      {  
        "Name": "CHEF_AUTOMATE_ADMIN_PASSWORD",  
        "Value": "1Example1"  
      }  
    ],  
    "EngineModel": "Single",  
    "EngineVersion": "2019-08",  
    "InstanceProfileArn": "arn:aws:iam::012345678901:instance-profile/aws-  
opsworks-cm-ec2-role",  
    "InstanceType": "t2.medium",  
    "PreferredBackupWindow": "Sun:02:00",  
    "PreferredMaintenanceWindow": "00:00",  
    "SecurityGroupIds": [ "sg-12345678" ],  
    "ServerArn": "arn:aws:iam::012345678901:instance/automate-06-1010V4UU2WRM2",  
    "ServerName": "automate-06",  
    "ServiceRoleArn": "arn:aws:iam::012345678901:role/aws-opsworks-cm-service-  
role",  
    "Status": "CREATING",  
    "SubnetIds": [ "subnet-12345678" ]  
  }  
}
```

```
}
```

Para obtener más información, consulte [CreateServer](#) la referencia sobre Chef AWS OpsWorks Automate. API

- Para API obtener más información, consulte [CreateServer](#) la Referencia de AWS CLI comandos.

## create-stack

En el siguiente ejemplo de código, se muestra cómo usar create-stack.

### AWS CLI

Para crear una pila

El siguiente create-stack comando crea una pila denominada CLI Stack.

```
aws opsworks create-stack --name "CLI Stack" --stack-region "us-east-1" --service-  
role-arn arn:aws:iam::123456789012:role/aws-opsworks-service-role --default-  
instance-profile-arn arn:aws:iam::123456789012:instance-profile/aws-opsworks-ec2-  
role --region us-east-1
```

Los parámetros service-role-arn y default-instance-profile-arn son obligatorios. Por lo general, utilizas las que se AWS OpsWorks crean automáticamente cuando creas tu primera pila. Para obtener los nombres de recursos de Amazon (ARNs) de su cuenta, vaya a la IAM consola, elija Roles en el panel de navegación, elija el rol o el perfil y elija la Summary pestaña.

Salida:

```
{  
  "StackId": "f6673d70-32e6-4425-8999-265dd002fec7"  
}
```

### Más información

Para obtener más información, consulte Crear una pila nueva en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [CreateStack](#) la Referencia de AWS CLI comandos.

## create-user-profile

En el siguiente ejemplo de código, se muestra cómo usar `create-user-profile`.

### AWS CLI

Para crear un perfil de usuario

Para importar un usuario de AWS Identity and Access Manager (IAM), debe llamar `create-user-profile` para crear un perfil de usuario. AWS OpsWorks En el siguiente ejemplo, se crea un perfil de usuario para el `cli-user-test` IAM usuario, que se identifica mediante Amazon Resource Name (ARN). En el ejemplo se asigna al usuario un SSH nombre de usuario `myusername` y se habilita la autogestión, lo que permite al usuario especificar una clave SSH pública.

```
aws opsworks --region us-east-1 create-user-profile --iam-user-arn arn:aws:iam::123456789102:user/cli-user-test --ssh-username myusername --allow-self-management
```

Salida:

```
{
  "IamUserArn": "arn:aws:iam::123456789102:user/cli-user-test"
}
```

Consejo: Este comando importa un IAM usuario a AWS OpsWorks, pero solo con los permisos que otorgan las políticas adjuntas. Puede conceder AWS OpsWorks permisos por pila mediante el `set-permissions` comando.

Más información

Para obtener más información, consulte Importación de usuarios a AWS OpsWorks en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [CreateUserProfile](#) la Referencia de AWS CLI comandos.

## delete-app

En el siguiente ejemplo de código, se muestra cómo usar `delete-app`.

## AWS CLI

Para eliminar una aplicación

En el siguiente ejemplo, se elimina una aplicación específica, que se identifica por su ID de aplicación. Para obtener un ID de aplicación, ve a la página de detalles de la aplicación en la AWS OpsWorks consola o ejecuta el `describe-apps` comando.

```
aws opsworks delete-app --region us-east-1 --app-id 577943b9-2ec1-4baf-  
a7bf-1d347601edc5
```

Resultado: ninguno.

Más información

Para obtener más información, consulte Aplicaciones en la Guía AWS OpsWorks del usuario.

- Para API obtener más información, consulte [DeleteApp](#) la Referencia de AWS CLI comandos.

## **delete-instance**

En el siguiente ejemplo de código, se muestra cómo usar `delete-instance`.

## AWS CLI

Para eliminar una instancia

En el siguiente `delete-instance` ejemplo, se elimina una instancia específica, que se identifica por su ID de instancia. Para encontrar un ID de instancia, abre la página de detalles de la instancia en la AWS OpsWorks consola o ejecuta el `describe-instances` comando.

Si la instancia está en línea, primero debes detenerla mediante una llamada `ystop-instance`, después, esperar a que se detenga. Ejecuta `describe-instances` para comprobar el estado de la instancia.

Para eliminar los EBS volúmenes de Amazon o las direcciones IP elásticas de la instancia, agrega los `--delete-elastic-ip` argumentos `--delete-volumes` o, respectivamente.

```
aws opsworks delete-instance \  
  --region us-east-1 \  
  --instance-id 3a21cfac-4a1f-4ce2-a921-b2cfba6f7771
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar AWS OpsWorks instancias](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DeleteInstance](#) la Referencia de AWS CLI comandos.

## **delete-layer**

En el siguiente ejemplo de código, se muestra cómo usar `delete-layer`.

### AWS CLI

Para eliminar una capa

En el siguiente ejemplo, se elimina una capa específica, que se identifica por su ID de capa. Para obtener un ID de capa, vaya a la página de detalles de la capa en la AWS OpsWorks consola o ejecute el `describe-layers` comando.

Nota: Antes de eliminar una capa, debe utilizar esta `delete-instance` opción para eliminar todas las instancias de la capa.

```
aws opsworks delete-layer --region us-east-1 --layer-id a919454e-b816-4598-b29a-5796afb498ed
```

Salida: ninguna.

### Más información

Para obtener más información, consulte [Eliminar AWS OpsWorks instancias](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DeleteLayer](#) la Referencia de AWS CLI comandos.

## **delete-stack**

En el siguiente ejemplo de código, se muestra cómo usar `delete-stack`.

### AWS CLI

Eliminación de una pila

El siguiente ejemplo elimina una pila específica, que se identifica por su ID de pila. Para obtener un ID de pila, haga clic en Stack Settings en la AWS OpsWorks consola o ejecute el `describe-stacks` comando.

Nota: Antes de eliminar una capa `delete-appdelete-instance`, debes usar `delete-layer` eliminar todas las aplicaciones, instancias y capas de la pila.

```
aws opsworks delete-stack --region us-east-1 --stack-id 154a9d89-7e9e-433b-8de8-617e53756c84
```

Salida: ninguna.

Más información

Para obtener más información, consulte Shut Down a Stack en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DeleteStack](#) la Referencia de AWS CLI comandos.

## **delete-user-profile**

En el siguiente ejemplo de código, se muestra cómo usar `delete-user-profile`.

AWS CLI

Para eliminar un perfil de usuario y eliminar un IAM usuario de AWS OpsWorks

En el siguiente ejemplo, se elimina el perfil de usuario de un usuario específico de AWS Identity and Access Management (IAM), que se identifica mediante Amazon Resource Name (ARN). La operación elimina al usuario AWS OpsWorks, pero no lo elimina. IAM Debe usar la IAM consola o API para esa tarea. CLI

```
aws opsworks --region us-east-1 delete-user-profile --iam-user-arn arn:aws:iam::123456789102:user/cli-user-test
```

Resultado: ninguno.

Más información

Para obtener más información, consulte Importación de usuarios a AWS OpsWorks en la Guía del AWS OpsWorks usuario.



- Para API obtener más información, consulte [DeleteUserProfile](#) la Referencia de AWS CLI comandos.

## deregister-elastic-ip

En el siguiente ejemplo de código, se muestra cómo usar `deregister-elastic-ip`.

### AWS CLI

Para anular el registro de una dirección IP elástica de una pila

El siguiente ejemplo anula el registro de una dirección IP elástica, identificada por su dirección IP, de su pila.

```
aws opsworks deregister-elastic-ip --region us-east-1 --elastic-ip 54.148.130.96
```

Salida: ninguna.

### Más información

Para obtener más información, consulte Anular el registro de direcciones IP elásticas en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte la Referencia [DeregisterElasticIp](#) de AWS CLI comandos.

## deregister-instance

En el siguiente ejemplo de código, se muestra cómo usar `deregister-instance`.

### AWS CLI

Para anular el registro de una instancia registrada de una pila

El siguiente `deregister-instance` comando anula el registro de una instancia registrada de su pila.

```
aws opsworks --region us-east-1 deregister-instance --instance-id 4d6d1710-ded9-42a1-b08e-b043ad7af1e2
```

Resultado: ninguno.

## Más información

Para obtener más información, consulte Anular el registro de una instancia registrada en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte la Referencia [DeregisterInstance](#) de AWS CLI comandos.

## **deregister-rds-db-instance**

En el siguiente ejemplo de código, se muestra cómo usar `deregister-rds-db-instance`.

### AWS CLI

Para anular el registro de una RDS instancia de base de datos de Amazon de una pila

El siguiente ejemplo anula el registro de una RDS instancia de base de datos, identificada por su valorARN, de su pila.

```
aws opsworks deregister-rds-db-instance --region us-east-1 --rds-db-instance-arn arn:aws:rds:us-west-2:123456789012:db:clitestdb
```

Resultado: ninguno.

## Más información

Para obtener más información, consulte Anular el registro de Amazon RDS Instances en la Guía del ASW OpsWorks usuario.

ID de instancia: clitestdb Nombre de usuario principal: cliuser Master: some23! PWD Nombre de base de datos pwd: mydb  
aws opsworks deregister-rds-db-instance —region us-east-1 —arn:aws:rds:us-west- 2:645732743964:db:clitestdb rds-db-instance-arn

- API Para obtener [DeregisterRdsDbInstance](#) más AWS CLI información, consulte la Referencia de comandos.

## **deregister-volume**

En el siguiente ejemplo de código, se muestra cómo usar `deregister-volume`.

### AWS CLI

Para anular el registro de un volumen de Amazon EBS

El siguiente ejemplo anula el registro de un EBS volumen de su pila. El volumen se identifica por su identificador de volumen, que es el GUID que se AWS OpsWorks asignó al registrar el volumen en la pila, no por el identificador del EC2 volumen.

```
aws opsworks deregister-volume --region us-east-1 --volume-id 5c48ef52-3144-4bf5-beaa-fda4deb23d4d
```

Salida: ninguna.

Más información

Para obtener más información, consulte Anular el registro de Amazon EBS Volumes en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte la Referencia [DeregisterVolume](#) de AWS CLI comandos.

## describe-apps

En el siguiente ejemplo de código, se muestra cómo usar `describe-apps`.

AWS CLI

Para describir las aplicaciones

El siguiente `describe-apps` comando describe las aplicaciones de una pila específica.

```
aws opsworks describe-apps \  
  --stack-id 38ee91e2-abdc-4208-a107-0b7168b3cc7a \  
  --region us-east-1
```

Salida:

```
{  
  "Apps": [  
    {  
      "StackId": "38ee91e2-abdc-4208-a107-0b7168b3cc7a",  
      "AppSource": {  
        "Url": "https://s3-us-west-2.amazonaws.com/opsworks-demo-assets/  
simplejsp.zip",  
        "Type": "archive"  
      },  
    },  
  ],  
}
```

```

        "Name": "SimpleJSP",
        "EnableSsl": false,
        "SslConfiguration": {},
        "AppId": "da1decc1-0dff-43ea-ad7c-bb667cd87c8b",
        "Attributes": {
          "RailsEnv": null,
          "AutoBundleOnDeploy": "true",
          "DocumentRoot": "ROOT"
        },
        "Shortname": "simplejsp",
        "Type": "other",
        "CreatedAt": "2013-08-01T21:46:54+00:00"
      }
    ]
  }
}

```

Para obtener más información, consulte Aplicaciones en la Guía AWS OpsWorks del usuario.

- Para API obtener más información, consulte [DescribeApps](#) la Referencia de AWS CLI comandos.

## describe-commands

En el siguiente ejemplo de código, se muestra cómo usar `describe-commands`.

### AWS CLI

Para describir los comandos

El siguiente `describe-commands` comando describe los comandos de una instancia específica.

```

aws opsworks describe-commands \
  --instance-id 8c2673b9-3fe5-420d-9cfa-78d875ee7687 \
  --region us-east-1

```

Salida:

```

{
  "Commands": [
    {
      "Status": "successful",
      "CompletedAt": "2013-07-25T18:57:47+00:00",

```

```

    "InstanceId": "8c2673b9-3fe5-420d-9cfa-78d875ee7687",
    "DeploymentId": "6ed0df4c-9ef7-4812-8dac-d54a05be1029",
    "AcknowledgedAt": "2013-07-25T18:57:41+00:00",
    "LogUrl": "https://s3.amazonaws.com/<bucket-name>/logs/008c1a91-
ec59-4d51-971d-3adff54b00cc?AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE
&Expires=1375394373&Signature=HkXil6UuNfxTCC37EPQAa462E1E%3D&response-cache-
control=private&response-content-encoding=gzip&response-content-type=text%2Fplain",
    "Type": "undeploy",
    "CommandId": "008c1a91-ec59-4d51-971d-3adff54b00cc",
    "CreatedAt": "2013-07-25T18:57:34+00:00",
    "ExitCode": 0
  },
  {
    "Status": "successful",
    "CompletedAt": "2013-07-25T18:55:40+00:00",
    "InstanceId": "8c2673b9-3fe5-420d-9cfa-78d875ee7687",
    "DeploymentId": "19d3121e-d949-4ff2-9f9d-94eac087862a",
    "AcknowledgedAt": "2013-07-25T18:55:32+00:00",
    "LogUrl": "https://s3.amazonaws.com/<bucket-name>/
logs/899d3d64-0384-47b6-a586-33433aad117c?AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE
&Expires=1375394373&Signature=xMsJvtLuUqWmsr8s%2FAjVru0BtRs%3D&response-cache-
control=private&response-content-encoding=gzip&response-content-type=text%2Fplain",
    "Type": "deploy",
    "CommandId": "899d3d64-0384-47b6-a586-33433aad117c",
    "CreatedAt": "2013-07-25T18:55:29+00:00",
    "ExitCode": 0
  }
]
}

```

Para obtener más información, consulte los eventos AWS OpsWorks del ciclo de vida en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribeCommands](#) la Referencia de AWS CLI comandos.

## describe-deployments

En el siguiente ejemplo de código, se muestra cómo usar `describe-deployments`.

### AWS CLI

Para describir las implementaciones

El siguiente `describe-deployments` comando describe las implementaciones de una pila específica.

```
aws opsworks --region us-east-1 describe-deployments --stack-id 38ee91e2-abdc-4208-a107-0b7168b3cc7a
```

Salida:

```
{
  "Deployments": [
    {
      "StackId": "38ee91e2-abdc-4208-a107-0b7168b3cc7a",
      "Status": "successful",
      "CompletedAt": "2013-07-25T18:57:49+00:00",
      "DeploymentId": "6ed0df4c-9ef7-4812-8dac-d54a05be1029",
      "Command": {
        "Args": {},
        "Name": "undeploy"
      },
    },
    "CreatedAt": "2013-07-25T18:57:34+00:00",
    "Duration": 15,
    "InstanceIds": [
      "8c2673b9-3fe5-420d-9cfa-78d875ee7687",
      "9e588a25-35b2-4804-bd43-488f85ebe5b7"
    ]
  },
  {
    "StackId": "38ee91e2-abdc-4208-a107-0b7168b3cc7a",
    "Status": "successful",
    "CompletedAt": "2013-07-25T18:56:41+00:00",
    "IamUserArn": "arn:aws:iam::123456789012:user/someuser",
    "DeploymentId": "19d3121e-d949-4ff2-9f9d-94eac087862a",
    "Command": {
      "Args": {},
      "Name": "deploy"
    },
  },
  "InstanceIds": [
    "8c2673b9-3fe5-420d-9cfa-78d875ee7687",
    "9e588a25-35b2-4804-bd43-488f85ebe5b7"
  ],
  "Duration": 72,
  "CreatedAt": "2013-07-25T18:55:29+00:00"
}
```

```
]
}
```

## Más información

Para obtener más información, consulte Implementación de aplicaciones en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribeDeployments](#) la Referencia de AWS CLI comandos.

## describe-elastic-ips

En el siguiente ejemplo de código, se muestra cómo usar `describe-elastic-ips`.

### AWS CLI

Para describir las instancias de Elastic IP

El siguiente `describe-elastic-ips` comando describe las direcciones IP elásticas de una instancia específica.

```
aws opsworks --region us-east-1 describe-elastic-ips --instance-id b62f3e04-  
e9eb-436c-a91f-d9e9a396b7b0
```

Salida:

```
{
  "ElasticIps": [
    {
      "Ip": "192.0.2.0",
      "Domain": "standard",
      "Region": "us-west-2"
    }
  ]
}
```

## Más información

Para obtener más información, consulte Instancias en la Guía AWS OpsWorks del usuario.

- Para API obtener más información, consulte [DescribeElasticLoadBalancers](#) la Referencia de AWS CLI comandos.

## describe-elastic-load-balancers

En el siguiente ejemplo de código, se muestra cómo usar `describe-elastic-load-balancers`.

### AWS CLI

Para describir los balanceadores de carga elásticos de una pila

El siguiente `describe-elastic-load-balancers` comando describe los balanceadores de carga de una pila específica.

```
aws opsworks --region us-west-2 describe-elastic-load-balancers --stack-id 6f4660e5-37a6-4e42-bfa0-1358ebd9c182
```

Resultado: esta pila en particular tiene un balanceador de carga.

```
{
  "ElasticLoadBalancers": [
    {
      "SubnetIds": [
        "subnet-60e4ea04",
        "subnet-66e1c110"
      ],
      "Ec2InstanceIds": [],
      "ElasticLoadBalancerName": "my-balancer",
      "Region": "us-west-2",
      "LayerId": "344973cb-bf2b-4cd0-8d93-51cd819bab04",
      "AvailabilityZones": [
        "us-west-2a",
        "us-west-2b"
      ],
      "VpcId": "vpc-b319f9d4",
      "StackId": "6f4660e5-37a6-4e42-bfa0-1358ebd9c182",
      "DnsName": "my-balancer-2094040179.us-west-2.elb.amazonaws.com"
    }
  ]
}
```

### Más información



Para obtener más información, consulte Aplicaciones en la Guía AWS OpsWorks del usuario.

- Para API obtener más información, consulte [DescribeElasticLoadBalancers](#) la Referencia de AWS CLI comandos.

## describe-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-instances`.

### AWS CLI

Para describir las instancias

El siguiente `describe-instances` comando describe las instancias de una pila específica:

```
aws opsworks --region us-east-1 describe-instances --stack-id 8c428b08-a1a1-46ce-a5f8-feddc43771b8
```

Salida: el siguiente ejemplo de salida es para una pila con dos instancias. La primera es una EC2 instancia registrada y la segunda la creó AWS OpsWorks.

```
{
  "Instances": [
    {
      "StackId": "71c7ca72-55ae-4b6a-8ee1-a8dcdded3fa0f",
      "PrivateDns": "ip-10-31-39-66.us-west-2.compute.internal",
      "LayerIds": [
        "26cf1d32-6876-42fa-bbf1-9cadcd0bff938"
      ],
      "EbsOptimized": false,
      "ReportedOs": {
        "Version": "14.04",
        "Name": "ubuntu",
        "Family": "debian"
      },
      "Status": "online",
      "InstanceId": "4d6d1710-ded9-42a1-b08e-b043ad7af1e2",
      "SshKeyName": "US-West-2",
      "InfrastructureClass": "ec2",
      "RootDeviceVolumeId": "vol-d08ec6c1",
      "SubnetId": "subnet-b8de0ddd",
      "InstanceType": "t1.micro",
      "CreatedAt": "2015-02-24T20:52:49+00:00",
```

```
"AmiId": "ami-35501205",
"Hostname": "ip-192-0-2-0",
"Ec2InstanceId": "i-5cd23551",
"PublicDns": "ec2-192-0-2-0.us-west-2.compute.amazonaws.com",
"SecurityGroupIds": [
  "sg-c4d3f0a1"
],
"Architecture": "x86_64",
"RootDeviceType": "ebs",
"InstallUpdatesOnBoot": true,
"Os": "Custom",
"VirtualizationType": "paravirtual",
"AvailabilityZone": "us-west-2a",
"PrivateIp": "10.31.39.66",
"PublicIp": "192.0.2.06",
"RegisteredBy": "arn:aws:iam::123456789102:user/AWS/OpsWorks/OpsWorks-
EC2Register-i-5cd23551"
},
{
  "StackId": "71c7ca72-55ae-4b6a-8ee1-a8dcdded3fa0f",
  "PrivateDns": "ip-10-31-39-158.us-west-2.compute.internal",
  "SshHostRsaKeyFingerprint": "69:6b:7b:8b:72:f3:ed:23:01:00:05:bc:9f:a4:60:c1",
  "LayerIds": [
    "26cf1d32-6876-42fa-bbf1-9cad0bfff938"
  ],
  "EbsOptimized": false,
  "ReportedOs": {},
  "Status": "booting",
  "InstanceId": "9b137a0d-2f5d-4cc0-9704-13da4b31fdcb",
  "SshKeyName": "US-West-2",
  "InfrastructureClass": "ec2",
  "RootDeviceVolumeId": "vol-e09dd5f1",
  "SubnetId": "subnet-b8de0ddd",
  "InstanceProfileArn": "arn:aws:iam::123456789102:instance-profile/aws-
opsworks-ec2-role",
  "InstanceType": "c3.large",
  "CreatedAt": "2015-02-24T21:29:33+00:00",
  "AmiId": "ami-9fc29baf",
  "SshHostDsaKeyFingerprint": "fc:87:95:c3:f5:e1:3b:9f:d2:06:6e:62:9a:35:27:e8",
  "Ec2InstanceId": "i-8d2dca80",
  "PublicDns": "ec2-192-0-2-1.us-west-2.compute.amazonaws.com",
  "SecurityGroupIds": [
    "sg-b022add5",
    "sg-b122add4"
```

```
    ],  
    "Architecture": "x86_64",  
    "RootDeviceType": "ebs",  
    "InstallUpdatesOnBoot": true,  
    "Os": "Amazon Linux 2014.09",  
    "VirtualizationType": "paravirtual",  
    "AvailabilityZone": "us-west-2a",  
    "Hostname": "custom11",  
    "PrivateIp": "10.31.39.158",  
    "PublicIp": "192.0.2.0"  
  }  
]  
}
```

## Más información

Para obtener más información, consulte Instancias en la Guía AWS OpsWorks del usuario.

- Para API obtener más información, consulte [DescribeInstances](#) la Referencia de AWS CLI comandos.

## describe-layers

En el siguiente ejemplo de código, se muestra cómo usar `describe-layers`.

### AWS CLI

Para describir las capas de una pila

El siguiente `describe-layers` comando describe las capas de una pila específica:

```
aws opsworks --region us-east-1 describe-layers --stack-id 38ee91e2-abdc-4208-a107-0b7168b3cc7a
```

Salida:

```
{  
  "Layers": [  
    {  
      "StackId": "38ee91e2-abdc-4208-a107-0b7168b3cc7a",  
      "Type": "db-master",  
      "DefaultSecurityGroupNames": [  
        "AWS-OpsWorks-DB-Master-Server"  
      ]  
    }  
  ]  
}
```

```
],
  "Name": "MySQL",
  "Packages": [],
  "DefaultRecipes": {
    "Undeploy": [],
    "Setup": [
      "opsworks_initial_setup",
      "ssh_host_keys",
      "ssh_users",
      "mysql::client",
      "dependencies",
      "ebs",
      "opsworks_ganglia::client",
      "mysql::server",
      "dependencies",
      "deploy:mysql"
    ],
    "Configure": [
      "opsworks_ganglia::configure-client",
      "ssh_users",
      "agent_version",
      "deploy:mysql"
    ],
    "Shutdown": [
      "opsworks_shutdown::default",
      "mysql::stop"
    ],
    "Deploy": [
      "deploy::default",
      "deploy:mysql"
    ]
  },
  "CustomRecipes": {
    "Undeploy": [],
    "Setup": [],
    "Configure": [],
    "Shutdown": [],
    "Deploy": []
  },
  "EnableAutoHealing": false,
  "LayerId": "41a20847-d594-4325-8447-171821916b73",
  "Attributes": {
    "MysqlRootPasswordUbiquitous": "true",
    "RubygemsVersion": null,
```

```

    "RailsStack": null,
    "HaproxyHealthCheckMethod": null,
    "RubyVersion": null,
    "BundlerVersion": null,
    "HaproxyStatsPassword": null,
    "PassengerVersion": null,
    "MemcachedMemory": null,
    "EnableHaproxyStats": null,
    "ManageBundler": null,
    "NodejsVersion": null,
    "HaproxyHealthCheckUrl": null,
    "MysqlRootPassword": "*****FILTERED*****",
    "GangliaPassword": null,
    "GangliaUser": null,
    "HaproxyStatsUrl": null,
    "GangliaUrl": null,
    "HaproxyStatsUser": null
  },
  "Shortname": "db-master",
  "AutoAssignElasticIps": false,
  "CustomSecurityGroupIds": [],
  "CreatedAt": "2013-07-25T18:11:19+00:00",
  "VolumeConfigurations": [
    {
      "MountPoint": "/vol/mysql",
      "Size": 10,
      "NumberOfDisks": 1
    }
  ]
},
{
  "StackId": "38ee91e2-abdc-4208-a107-0b7168b3cc7a",
  "Type": "custom",
  "DefaultSecurityGroupNames": [
    "AWS-OpsWorks-Custom-Server"
  ],
  "Name": "TomCustom",
  "Packages": [],
  "DefaultRecipes": {
    "Undeploy": [],
    "Setup": [
      "opsworks_initial_setup",
      "ssh_host_keys",
      "ssh_users",

```

```
        "mysql::client",
        "dependencies",
        "ebs",
        "opsworks_ganglia::client"
    ],
    "Configure": [
        "opsworks_ganglia::configure-client",
        "ssh_users",
        "agent_version"
    ],
    "Shutdown": [
        "opsworks_shutdown::default"
    ],
    "Deploy": [
        "deploy::default"
    ]
},
"CustomRecipes": {
    "Undeploy": [],
    "Setup": [
        "tomcat::setup"
    ],
    "Configure": [
        "tomcat::configure"
    ],
    "Shutdown": [],
    "Deploy": [
        "tomcat::deploy"
    ]
},
"EnableAutoHealing": true,
"LayerId": "e6cbcd29-d223-40fc-8243-2eb213377440",
"Attributes": {
    "MysqlRootPasswordUbiquitous": null,
    "RubygemsVersion": null,
    "RailsStack": null,
    "HaproxyHealthCheckMethod": null,
    "RubyVersion": null,
    "BundlerVersion": null,
    "HaproxyStatsPassword": null,
    "PassengerVersion": null,
    "MemcachedMemory": null,
    "EnableHaproxyStats": null,
    "ManageBundler": null,
```

```
        "NodejsVersion": null,  
        "HaproxyHealthCheckUrl": null,  
        "MysqlRootPassword": null,  
        "GangliaPassword": null,  
        "GangliaUser": null,  
        "HaproxyStatsUrl": null,  
        "GangliaUrl": null,  
        "HaproxyStatsUser": null  
    },  
    "Shortname": "tomcustom",  
    "AutoAssignElasticIps": false,  
    "CustomSecurityGroupIds": [],  
    "CreatedAt": "2013-07-25T18:12:53+00:00",  
    "VolumeConfigurations": []  
  }  
]  
}
```

## Más información

Para obtener más información, consulte Capas en la Guía AWS OpsWorks del usuario.

- Para API obtener más información, consulte [DescribeLayers](#) la Referencia de AWS CLI comandos.

## describe-load-based-auto-scaling

En el siguiente ejemplo de código, se muestra cómo usar `describe-load-based-auto-scaling`.

### AWS CLI

Para describir la configuración de escalado basada en la carga de una capa

El siguiente ejemplo describe la configuración de escalado basada en la carga de una capa específica. La capa se identifica por su identificador de capa, que puede encontrar en la página de detalles de la capa o `describe-layers` ejecutándolo.

```
aws opsworks describe-load-based-auto-scaling --region us-east-1 --layer-  
ids 6bec29c9-c866-41a0-aba5-fa3e374ce2a1
```

Resultado: la capa de ejemplo tiene una única instancia basada en la carga.

```
{
  "LoadBasedAutoScalingConfigurations": [
    {
      "DownScaling": {
        "IgnoreMetricsTime": 10,
        "ThresholdsWaitTime": 10,
        "InstanceCount": 1,
        "CpuThreshold": 30.0
      },
      "Enable": true,
      "UpScaling": {
        "IgnoreMetricsTime": 5,
        "ThresholdsWaitTime": 5,
        "InstanceCount": 1,
        "CpuThreshold": 80.0
      },
      "LayerId": "6bec29c9-c866-41a0-aba5-fa3e374ce2a1"
    }
  ]
}
```

## Más información

Para obtener más información, consulte [Cómo funciona el escalado automático basado en la carga](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribeLoadBasedAutoScaling](#) la Referencia de AWS CLI comandos.

## **describe-my-user-profile**

En el siguiente ejemplo de código, se muestra cómo usar `describe-my-user-profile`.

### AWS CLI

Para obtener el perfil de un usuario

El siguiente ejemplo muestra cómo obtener el perfil del usuario de AWS Identity and Access Management (IAM) que ejecuta el comando.

```
aws opsworks --region us-east-1 describe-my-user-profile
```



Resultado: por motivos de brevedad, la mayor parte de la clave SSH pública del usuario se sustituye por puntos suspensivos (...).

```
{
  "UserProfile": {
    "IamUserArn": "arn:aws:iam::123456789012:user/myusername",
    "SshPublicKey": "ssh-rsa AAAAB3NzaC1yc2EAAAABJQ...3LQ4aX9jpxQw== rsa-
key-20141104",
    "Name": "myusername",
    "SshUsername": "myusername"
  }
}
```

Más información

Para obtener más información, consulte Importación de usuarios a AWS OpsWorks en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribeMyUserProfile](#) la Referencia de AWS CLI comandos.

## describe-permissions

En el siguiente ejemplo de código, se muestra cómo usar `describe-permissions`.

### AWS CLI

Para obtener el nivel de AWS OpsWorks permisos por pila de un usuario

El siguiente ejemplo muestra cómo obtener el nivel de permiso de un usuario de AWS Identity and Access Management (IAM) en una pila específica.

```
aws opsworks --region us-east-1 describe-permissions --iam-user-
arn arn:aws:iam::123456789012:user/cli-user-test --stack-id d72553d4-8727-448c-9b00-
f024f0ba1b06
```

Salida:

```
{
  "Permissions": [
    {
      "StackId": "d72553d4-8727-448c-9b00-f024f0ba1b06",
```

```
    "IamUserArn": "arn:aws:iam::123456789012:user/cli-user-test",
    "Level": "manage",
    "AllowSudo": true,
    "AllowSsh": true
  }
]
```

## Más información

Para obtener más información, consulte [Concesión de niveles de permisos por pila](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribePermissions](#) la Referencia de AWS CLI comandos.

## describe-raid-arrays

En el siguiente ejemplo de código, se muestra cómo usar `describe-raid-arrays`.

### AWS CLI

Para describir las RAID matrices

En el siguiente ejemplo, se describen las RAID matrices asociadas a las instancias de una pila específica.

```
aws opsworks --region us-east-1 describe-raid-arrays --stack-id d72553d4-8727-448c-9b00-f024f0ba1b06
```

Resultado: el siguiente es el resultado de una pila con una RAID matriz.

```
{
  "RaidArrays": [
    {
      "StackId": "d72553d4-8727-448c-9b00-f024f0ba1b06",
      "AvailabilityZone": "us-west-2a",
      "Name": "Created for php-app1",
      "NumberOfDisks": 2,
      "InstanceId": "9f14adbc-ced5-43b6-bf01-e7d0db6cf2f7",
      "RaidLevel": 0,
      "VolumeType": "standard",
```

```

    "RaidArrayId": "f2d4e470-5972-4676-b1b8-bae41ec3e51c",
    "Device": "/dev/md0",
    "MountPoint": "/mnt/workspace",
    "CreatedAt": "2015-02-26T23:53:09+00:00",
    "Size": 100
  }
]
}

```

Para obtener más información, consulte EBS Volúmenes en la Guía AWS OpsWorks del usuario.

- Para API obtener más información, consulte [DescribeRaidArrays](#) la Referencia de AWS CLI comandos.

## describe-rds-db-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-rds-db-instances`.

### AWS CLI

Para describir las RDS instancias de Amazon registradas de una pila

En el siguiente ejemplo, se describen las RDS instancias de Amazon registradas con una pila específica.

```

aws opsworks --region us-east-1 describe-rds-db-instances --stack-id d72553d4-8727-448c-9b00-f024f0ba1b06

```

Resultado: el siguiente es el resultado de una pila con una RDS instancia registrada.

```

{
  "RdsDbInstances": [
    {
      "Engine": "mysql",
      "StackId": "d72553d4-8727-448c-9b00-f024f0ba1b06",
      "MissingOnRds": false,
      "Region": "us-west-2",
      "RdsDbInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:clitestdb",
      "DbPassword": "*****FILTERED*****",
      "Address": "clitestdb.cd1qlk5uwd0k.us-west-2.rds.amazonaws.com",
      "DbUser": "cliuser",
      "DbInstanceIdentifier": "clitestdb"
    }
  ]
}

```

```
]
}
```

Para obtener más información, consulte Administración de recursos en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribeRdsDbInstances](#) la Referencia de AWS CLI comandos.

## describe-stack-provisioning-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-stack-provisioning-parameters`.

### AWS CLI

Para devolver los parámetros de aprovisionamiento de una pila

El siguiente `describe-stack-provisioning-parameters` ejemplo devuelve los parámetros de aprovisionamiento de una pila específica. Los parámetros de aprovisionamiento incluyen ajustes como la ubicación de instalación del agente y la clave pública que se OpsWorks utiliza para administrar el agente en las instancias de una pila.

```
aws opsworks describe-stack-provisioning-parameters \
  --stack-id 62744d97-6faf-4ecb-969b-a086fEXAMPLE
```

Salida:

```
{
  "AgentInstallerUrl": "https://opsworks-instance-agent-us-
west-2.s3.amazonaws.com/ID_number/opsworks-agent-installer.tgz",
  "Parameters": {
    "agent_installer_base_url": "https://opsworks-instance-agent-us-
west-2.s3.amazonaws.com",
    "agent_installer_tgz": "opsworks-agent-installer.tgz",
    "assets_download_bucket": "opsworks-instance-assets-us-
west-2.s3.amazonaws.com",
    "charlie_public_key": "-----BEGIN PUBLIC KEY-----PUBLIC_KEY_EXAMPLE\n-----
END PUBLIC KEY-----",
    "instance_service_endpoint": "opsworks-instance-service.us-
west-2.amazonaws.com",
    "instance_service_port": "443",
```

```
    "instance_service_region": "us-west-2",
    "instance_service_ssl_verify_peer": "true",
    "instance_service_use_ssl": "true",
    "ops_works_endpoint": "opsworks.us-west-2.amazonaws.com",
    "ops_works_port": "443",
    "ops_works_region": "us-west-2",
    "ops_works_ssl_verify_peer": "true",
    "ops_works_use_ssl": "true",
    "verbose": "false",
    "wait_between_runs": "30"
  }
}
```

Para obtener más información, consulte [Ejecutar comandos de pila](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribeStackProvisioningParameters](#) la Referencia de AWS CLI comandos.

## describe-stack-summary

En el siguiente ejemplo de código, se muestra cómo usar `describe-stack-summary`.

### AWS CLI

Para describir la configuración de una pila

El siguiente `describe-stack-summary` comando devuelve un resumen de la configuración de la pila especificada.

```
aws opsworks --region us-east-1 describe-stack-summary --stack-id 8c428b08-a1a1-46ce-a5f8-feddc43771b8
```

Salida:

```
{
  "StackSummary": {
    "StackId": "8c428b08-a1a1-46ce-a5f8-feddc43771b8",
    "InstancesCount": {
      "Booting": 1
    },
    "Name": "CLITest",
```

```

    "AppsCount": 1,
    "LayersCount": 1,
    "Arn": "arn:aws:opsworks:us-west-2:123456789012:stack/8c428b08-a1a1-46ce-a5f8-
feddc43771b8/"
  }
}

```

## Más información

Para obtener más información, consulte [Stacks](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribeStackSummary](#) la Referencia de AWS CLI comandos.

## describe-stacks

En el siguiente ejemplo de código, se muestra cómo usar `describe-stacks`.

### AWS CLI

Para describir las pilas

El siguiente `describe-stacks` comando describe las pilas de una cuenta.

```
aws opsworks --region us-east-1 describe-stacks
```

Salida:

```

{
  "Stacks": [
    {
      "ServiceRoleArn": "arn:aws:iam::444455556666:role/aws-opsworks-service-role",
      "StackId": "aeb7523e-7c8b-49d4-b866-03aae9d4fbcf",
      "DefaultRootDeviceType": "instance-store",
      "Name": "TomStack-sd",
      "ConfigurationManager": {
        "Version": "11.4",
        "Name": "Chef"
      },
      "UseCustomCookbooks": true,
      "CustomJson": "{\n  \"tomcat\": {\n    \"base_version\": 7,\n    \"java_opts\n\": \"-Djava.awt.headless=true -Xmx256m\"\n  },\n  \"datasources\": {\n    \"R00T\":\n  \"jdbc/mydb\"\n  }\n}",

```

```
    "Region": "us-east-1",
    "DefaultInstanceProfileArn": "arn:aws:iam::444455556666:instance-profile/aws-opsworks-ec2-role",
    "CustomCookbooksSource": {
      "Url": "git://github.com/example-repo/tomcustom.git",
      "Type": "git"
    },
    "DefaultAvailabilityZone": "us-east-1a",
    "HostnameTheme": "Layer_Dependent",
    "Attributes": {
      "Color": "rgb(45, 114, 184)"
    },
    "DefaultOs": "Amazon Linux",
    "CreatedAt": "2013-08-01T22:53:42+00:00"
  },
  {
    "ServiceRoleArn": "arn:aws:iam::444455556666:role/aws-opsworks-service-role",
    "StackId": "40738975-da59-4c5b-9789-3e422f2cf099",
    "DefaultRootDeviceType": "instance-store",
    "Name": "MyStack",
    "ConfigurationManager": {
      "Version": "11.4",
      "Name": "Chef"
    },
    "UseCustomCookbooks": false,
    "Region": "us-east-1",
    "DefaultInstanceProfileArn": "arn:aws:iam::444455556666:instance-profile/aws-opsworks-ec2-role",
    "CustomCookbooksSource": {},
    "DefaultAvailabilityZone": "us-east-1a",
    "HostnameTheme": "Layer_Dependent",
    "Attributes": {
      "Color": "rgb(45, 114, 184)"
    },
    "DefaultOs": "Amazon Linux",
    "CreatedAt": "2013-10-25T19:24:30+00:00"
  }
]
}
```

## Más información

Para obtener más información, consulte [Stacks](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribeStacks](#) la Referencia de AWS CLI comandos.

## describe-timebased-auto-scaling

En el siguiente ejemplo de código, se muestra cómo usar `describe-timebased-auto-scaling`.

### AWS CLI

Para describir la configuración de escalado basada en el tiempo de una instancia

En el siguiente ejemplo, se describe la configuración de escalado basada en el tiempo de una instancia específica. La instancia se identifica por su ID de instancia, que puedes encontrar en la página de detalles de la instancia o `describe-instances` ejecutándola.

```
aws opsworks describe-time-based-auto-scaling --region us-east-1 --instance-ids 701f2ffe-5d8e-4187-b140-77b75f55de8d
```

Resultado: el ejemplo tiene una única instancia basada en el tiempo.

```
{
  "TimeBasedAutoScalingConfigurations": [
    {
      "InstanceId": "701f2ffe-5d8e-4187-b140-77b75f55de8d",
      "AutoScalingSchedule": {
        "Monday": {
          "11": "on",
          "10": "on",
          "13": "on",
          "12": "on"
        },
        "Tuesday": {
          "11": "on",
          "10": "on",
          "13": "on",
          "12": "on"
        }
      }
    }
  ]
}
```



## Más información

Para obtener más información, consulte [Cómo funciona el escalado automático basado en el tiempo en la Guía del AWS OpsWorks usuario](#).

- Para API obtener más información, consulte [DescribeTimebasedAutoScaling](#) la Referencia de AWS CLI comandos.

## describe-user-profiles

En el siguiente ejemplo de código, se muestra cómo usar `describe-user-profiles`.

### AWS CLI

Para describir los perfiles de usuario

El siguiente `describe-user-profiles` comando describe los perfiles de usuario de la cuenta.

```
aws opsworks --region us-east-1 describe-user-profiles
```

Salida:

```
{
  "UserProfiles": [
    {
      "IamUserArn": "arn:aws:iam::123456789012:user/someuser",
      "SshPublicKey": "ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAQEak0uP7i80q3Cko...",
      "AllowSelfManagement": true,
      "Name": "someuser",
      "SshUsername": "someuser"
    },
    {
      "IamUserArn": "arn:aws:iam::123456789012:user/cli-user-test",
      "AllowSelfManagement": true,
      "Name": "cli-user-test",
      "SshUsername": "myusername"
    }
  ]
}
```

## Más información

Para obtener más información, consulte Administración de AWS OpsWorks usuarios en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribeUserProfiles](#) la Referencia de AWS CLI comandos.

## describe-volumes

En el siguiente ejemplo de código, se muestra cómo usar `describe-volumes`.

### AWS CLI

Para describir los volúmenes de una pila

El siguiente ejemplo describe los EBS volúmenes de una pila.

```
aws opsworks --region us-east-1 describe-volumes --stack-id 8c428b08-a1a1-46ce-a5f8-feddc43771b8
```

Salida:

```
{
  "Volumes": [
    {
      "Status": "in-use",
      "AvailabilityZone": "us-west-2a",
      "Name": "CLITest",
      "InstanceId": "dfe18b02-5327-493d-91a4-c5c0c448927f",
      "VolumeType": "standard",
      "VolumeId": "56b66fbd-e1a1-4aff-9227-70f77118d4c5",
      "Device": "/dev/sdi",
      "Ec2VolumeId": "vol-295c1638",
      "MountPoint": "/mnt/myvolume",
      "Size": 1
    }
  ]
}
```

### Más información

Para obtener más información, consulte Administración de recursos en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribeVolumes](#) la Referencia de AWS CLI comandos.

## **detach-elastic-load-balancer**

En el siguiente ejemplo de código, se muestra cómo usar `detach-elastic-load-balancer`.

### AWS CLI

Para separar un balanceador de cargas de su capa

En el siguiente ejemplo, se separa un balanceador de cargas, identificado por su nombre, de su capa.

```
aws opsworks --region us-east-1 detach-elastic-load-balancer --elastic-load-balancer-name Java-LB --layer-id 888c5645-09a5-4d0e-95a8-812ef1db76a4
```

Salida: ninguna.

### Más información

Para obtener más información, consulte Elastic Load Balancing en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DetachElasticLoadBalancer](#) la Referencia de AWS CLI comandos.

## **disassociate-elastic-ip**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-elastic-ip`.

### AWS CLI

Para desasociar una dirección IP elástica de una instancia

El siguiente ejemplo desasocia una dirección IP elástica de una instancia específica.

```
aws opsworks --region us-east-1 disassociate-elastic-ip --elastic-ip 54.148.130.96
```

Salida: ninguna.

## Más información

Para obtener más información, consulte Administración de recursos en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DisassociateElasticIp](#) la Referencia de AWS CLI comandos.

## get-hostname-suggestion

En el siguiente ejemplo de código, se muestra cómo usar `get-hostname-suggestion`.

### AWS CLI

Para obtener el siguiente nombre de host de una capa

El siguiente ejemplo obtiene el siguiente nombre de host generado para una capa especificada. La capa utilizada en este ejemplo es una capa de servidor de aplicaciones Java con una instancia. El tema del nombre de host de la pila es el predeterminado, `Layer_Dependent`.

```
aws opsworks --region us-east-1 get-hostname-suggestion --layer-id 888c5645-09a5-4d0e-95a8-812ef1db76a4
```

Salida:

```
{
  "Hostname": "java-app2",
  "LayerId": "888c5645-09a5-4d0e-95a8-812ef1db76a4"
}
```

## Más información

Para obtener más información, consulte Crear una pila nueva en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [GetHostnameSuggestion](#) la Referencia de AWS CLI comandos.

## reboot-instance

En el siguiente ejemplo de código, se muestra cómo usar `reboot-instance`.

## AWS CLI

Para reiniciar una instancia

En el siguiente ejemplo, se reinicia una instancia.

```
aws opsworks --region us-east-1 reboot-instance --instance-  
id dfe18b02-5327-493d-91a4-c5c0c448927f
```

Salida: ninguna.

### Más información

Para obtener más información, consulte Reiniciar una instancia en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte la Referencia [RebootInstance](#) de AWS CLI comandos.

## register-elastic-ip

En el siguiente ejemplo de código, se muestra cómo usar `register-elastic-ip`.

## AWS CLI

Para registrar una dirección IP elástica con una pila

El siguiente ejemplo registra una dirección IP elástica, identificada por su dirección IP, con una pila específica.

Nota: La dirección IP elástica debe estar en la misma región que la pila.

```
aws opsworks register-elastic-ip --region us-east-1 --stack-  
id d72553d4-8727-448c-9b00-f024f0ba1b06 --elastic-ip 54.148.130.96
```

### Salida

```
{  
  "ElasticIp": "54.148.130.96"  
}
```

### Más información

Para obtener más información, consulte Registrar direcciones IP elásticas con una pila en la Guía del OpsWorks usuario.

- Para API obtener más información, consulte [RegisterElasticIp](#) la Referencia de AWS CLI comandos.

## **register-rds-db-instance**

En el siguiente ejemplo de código, se muestra cómo usar `register-rds-db-instance`.

### AWS CLI

Para registrar una RDS instancia de Amazon con una pila

El siguiente ejemplo registra una RDS instancia de base de datos de Amazon, identificada por su nombre de recurso de Amazon (ARN), con una pila específica. También especifica el nombre de usuario y la contraseña maestros de la instancia. Tenga en cuenta que AWS OpsWorks esto no valida ninguno de estos valores. Si alguno de ellos es incorrecto, la aplicación no podrá conectarse a la base de datos.

```
aws opsworks register-rds-db-instance --region us-east-1 --stack-  
id d72553d4-8727-448c-9b00-f024f0ba1b06 --rds-db-instance-arn arn:aws:rds:us-  
west-2:123456789012:db:clitestdb --db-user cliuser --db-password some23!pwd
```

Salida: ninguna.

### Más información

Para obtener más información, consulte Registrar RDS instancias de Amazon con una pila en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [RegisterRdsDbInstance](#) la Referencia de AWS CLI comandos.

## **register-volume**

En el siguiente ejemplo de código, se muestra cómo usar `register-volume`.

### AWS CLI

Para registrar un EBS volumen de Amazon con una pila

El siguiente ejemplo registra un EBS volumen de Amazon, identificado por su ID de volumen, con una pila específica.

```
aws opsworks register-volume --region us-east-1 --stack-id d72553d4-8727-448c-9b00-f024f0ba1b06 --ec-2-volume-id vol-295c1638
```

Salida:

```
{
  "VolumeId": "ee08039c-7cb7-469f-be10-40fb7f0c05e8"
}
```

Más información

Para obtener más información, consulte Registrar EBS volúmenes de Amazon con una pila en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [RegisterVolume](#) la Referencia de AWS CLI comandos.

## register

En el siguiente ejemplo de código, se muestra cómo usar `register`.

### AWS CLI

Para registrar instancias con una pila

Los siguientes ejemplos muestran diversas formas de registrar instancias con una pila que se crearon fuera de AWS Opsworks. Puede ejecutar `register` desde la instancia que desea registrar o desde una estación de trabajo independiente. Para obtener más información, consulte Registrar instancias de Amazon EC2 y locales en la Guía del AWS OpsWorks usuario.

Nota: Por motivos de brevedad, los ejemplos omiten el argumento. `region`

Para registrar una EC2 instancia de Amazon

Para indicar que está registrando una EC2 instancia, defina el `--infrastructure-class` argumento `enec2`.

En el siguiente ejemplo, se registra una EC2 instancia con la pila especificada desde una estación de trabajo independiente. La instancia se identifica por su EC2 ID, `i-12345678`. El ejemplo usa

el nombre de SSH usuario predeterminado de la estación de trabajo e intenta iniciar sesión en la instancia mediante técnicas de autenticación que no requieren una contraseña, como una SSH clave privada predeterminada. Si esto no funciona, `register` busca la contraseña.

```
aws opsworks register --infrastructure-class=ec2 --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb i-12345678
```

En el siguiente ejemplo, se registra una EC2 instancia con la pila especificada desde una estación de trabajo independiente. Utiliza los `--ssh-private-key` argumentos `--ssh-username` y para especificar de forma explícita el SSH nombre de usuario y el archivo de clave privada que el comando utiliza para iniciar sesión en la instancia. `ec2-user` es el nombre de usuario estándar para las instancias de Amazon Linux. Úselo `ubuntu` para instancias de Ubuntu.

```
aws opsworks register --infrastructure-class=ec2 --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb --ssh-username ec2-user --ssh-private-key ssh_private_key i-12345678
```

En el siguiente ejemplo, se registra la EC2 instancia que ejecuta el `register` comando. Inicia sesión en la instancia con el `--local` argumento SSH y `register` ejecútalo con él en lugar de con un identificador de instancia o un nombre de host.

```
aws opsworks register --infrastructure-class ec2 --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb --local
```

Para registrar una instancia local

Para indicar que está registrando una instancia local, defina el `--infrastructure-class` argumento en `on-premises`

En el siguiente ejemplo, se registra una instancia local existente con una pila específica de una estación de trabajo independiente. La instancia se identifica por su dirección IP, `192.0.2.3`. El ejemplo usa el nombre de SSH usuario predeterminado de la estación de trabajo e intenta iniciar sesión en la instancia mediante técnicas de autenticación que no requieren una contraseña, como una SSH clave privada predeterminada. Si esto no funciona, `register` busca la contraseña.

```
aws opsworks register --infrastructure-class on-premises --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb 192.0.2.3
```



En el siguiente ejemplo, se registra una instancia local con una pila específica de una estación de trabajo independiente. La instancia se identifica por su nombre de host, `host1`. Los `--override-...` argumentos indican a AWS OpsWorks que se muestran `webserver1` como nombre de host, `192.0.2.3` y `10.0.0.2` como direcciones IP públicas y privadas de la instancia, respectivamente.

```
aws opsworks register --infrastructure-class on-premises --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb --override-hostname webserver1 --override-public-ip 192.0.2.3 --override-private-ip 10.0.0.2 host1
```

En el siguiente ejemplo, se registra una instancia local con una pila específica desde una estación de trabajo independiente. La instancia se identifica por su dirección IP. `register` inicia sesión en la instancia con el SSH nombre de usuario y el archivo de clave privada especificados.

```
aws opsworks register --infrastructure-class on-premises --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb --ssh-username admin --ssh-private-key ssh_private_key 192.0.2.3
```

En el siguiente ejemplo, se registra una instancia local existente con una pila específica desde una estación de trabajo independiente. El comando inicia sesión en la instancia mediante una cadena de SSH comandos personalizada que especifica la SSH contraseña y la dirección IP de la instancia.

```
aws opsworks register --infrastructure-class on-premises --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb --override-ssh "sshpass -p 'mypassword' ssh your-user@192.0.2.3"
```

En el siguiente ejemplo, se registra la instancia local que ejecuta el `register` comando. Inicia sesión en la instancia con el `--local` argumento SSH y ejecútalo `register` con él en lugar de con un identificador de instancia o un nombre de host.

```
aws opsworks register --infrastructure-class on-premises --stack-id 935450cc-61e0-4b03-a3e0-160ac817d2bb --local
```

Resultado: el siguiente es el resultado típico para registrar una EC2 instancia.

```
Warning: Permanently added '52.11.41.206' (ECDSA) to the list of known hosts.
% Total      % Received % Xferd  Average Speed   Time    Time     Time  Current
```

```

          Dload Upload  Total  Spent  Left  Speed
100 6403k  100 6403k   0    0 2121k    0 0:00:03 0:00:03 --:--:-- 2121k
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Initializing AWS OpsWorks
environment
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Running on Ubuntu
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Checking if OS is supported
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Running on supported OS
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Setup motd
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Executing: ln -sf --backup /etc/
motd.opsworks-static /etc/motd
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Enabling multiverse repositories
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Customizing APT environment
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Installing system packages
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Executing: dpkg --configure -a
[Tue, 24 Feb 2015 20:48:37 +0000] opsworks-init: Executing with retry: apt-get
update
[Tue, 24 Feb 2015 20:49:13 +0000] opsworks-init: Executing: apt-get install -y ruby
ruby-dev libicu-dev libssl-dev libxslt-dev libxml2-dev libyaml-dev monit
[Tue, 24 Feb 2015 20:50:13 +0000] opsworks-init: Using assets bucket from
environment: 'opsworks-instance-assets-us-east-1.s3.amazonaws.com'.
[Tue, 24 Feb 2015 20:50:13 +0000] opsworks-init: Installing Ruby for the agent
[Tue, 24 Feb 2015 20:50:13 +0000] opsworks-init: Executing: /tmp/opsworks-
agent-installer.YgGq8wF3UUre6yDy/opsworks-agent-installer/opsworks-agent/bin/
installer_wrapper.sh -r -R opsworks-instance-assets-us-east-1.s3.amazonaws.com
[Tue, 24 Feb 2015 20:50:44 +0000] opsworks-init: Starting the installer
Instance successfully registered. Instance ID: 4d6d1710-ded9-42a1-b08e-b043ad7af1e2
Connection to 52.11.41.206 closed.

```

## Más información

Para obtener más información, consulte [Registrar una instancia con una AWS OpsWorks pila en la Guía del AWS OpsWorks usuario](#).

- Para API obtener más información, consulte [Register](#) in AWS CLI Command Reference.

## set-load-based-auto-scaling

En el siguiente ejemplo de código, se muestra cómo usar `set-load-based-auto-scaling`.

### AWS CLI

Para establecer la configuración de escalado basada en la carga de una capa

El siguiente ejemplo habilita el escalado basado en la carga para una capa específica y establece la configuración de esa capa. Debe utilizarse `create-instance` para añadir instancias basadas en la carga a la capa.

```
aws opsworks --region us-east-1 set-load-based-auto-scaling --layer-id 523569ae-2faf-47ac-b39e-f4c4b381f36d --enable --up-scaling file://upscale.json --down-scaling file://downscale.json
```

El ejemplo coloca la configuración del umbral de escalado en un archivo independiente del directorio de trabajo denominado `upscale.json`, que contiene lo siguiente.

```
{
  "InstanceCount": 2,
  "ThresholdsWaitTime": 3,
  "IgnoreMetricsTime": 3,
  "CpuThreshold": 85,
  "MemoryThreshold": 85,
  "LoadThreshold": 85
}
```

El ejemplo coloca la configuración del umbral de reducción de escala en un archivo independiente del directorio de trabajo denominado `downscale.json`, que contiene lo siguiente.

```
{
  "InstanceCount": 2,
  "ThresholdsWaitTime": 3,
  "IgnoreMetricsTime": 3,
  "CpuThreshold": 35,
  "MemoryThreshold": 30,
  "LoadThreshold": 30
}
```

Resultado: ninguno.

## Más información

Para obtener más información, consulte [Uso del escalado automático basado en la carga](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [SetLoadBasedAutoScaling](#) la Referencia de AWS CLI comandos.

## set-permission

En el siguiente ejemplo de código, se muestra cómo usar `set-permission`.

### AWS CLI

Para conceder niveles de AWS OpsWorks permisos por pila

Al importar un usuario de AWS Identity and Access Management (IAM) a AWS OpsWorks mediante una llamada `create-user-profile`, el usuario solo tiene los permisos que otorgan IAM las políticas adjuntas. Puede conceder AWS OpsWorks permisos modificando las políticas de un usuario. Sin embargo, suele ser más fácil importar un usuario y, a continuación, utilizar el `set-permission` comando para conceder al usuario uno de los niveles de permisos estándar para cada pila a la que el usuario necesite acceder.

El siguiente ejemplo concede permiso para la pila especificada a un usuario, que se identifica mediante Amazon Resource Name (ARN). El ejemplo otorga al usuario un nivel de administración de permisos, con `sudo` y SSH privilegios en las instancias de la pila.

```
aws opsworks set-permission --region us-east-1 --stack-id 71c7ca72-55ae-4b6a-8ee1-a8dcded3fa0f --level manage --iam-user-arn arn:aws:iam::123456789102:user/cli-user-test --allow-ssh --allow-sudo
```

Resultado: ninguno.

### Más información

Para obtener más información, consulte [Concesión de permisos por pila a los AWS OpsWorks usuarios](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [SetPermission](#) la Referencia de AWS CLI comandos.

## set-time-based-auto-scaling

En el siguiente ejemplo de código, se muestra cómo usar `set-time-based-auto-scaling`.

### AWS CLI

Para establecer la configuración de escalado basada en el tiempo para una capa

El siguiente ejemplo establece la configuración basada en el tiempo para una instancia específica. Primero debe usar `create-instance` para agregar la instancia a la capa.

```
aws opsworks --region us-east-1 set-time-based-auto-scaling --instance-id 69b6237c-08c0-4edb-a6af-78f3d01cedf2 --auto-scaling-schedule file://schedule.json
```

En el ejemplo, se coloca la programación en un archivo independiente del directorio de trabajo denominadoschedule.json. En este ejemplo, la instancia estará activa durante unas horas alrededor del mediodía UTC (hora universal coordinada) del lunes y el martes.

```
{
  "Monday": {
    "10": "on",
    "11": "on",
    "12": "on",
    "13": "on"
  },
  "Tuesday": {
    "10": "on",
    "11": "on",
    "12": "on",
    "13": "on"
  }
}
```

Salida: ninguna.

### Más información

Para obtener más información, consulte [Uso del escalado automático basado en el tiempo](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [SetTimeBasedAutoScaling](#) la Referencia de AWS CLI comandos.

## start-instance

En el siguiente ejemplo de código, se muestra cómo usar `start-instance`.

### AWS CLI

Para iniciar una instancia

El siguiente `start-instance` comando inicia una instancia específica de 24 horas al día, 7 días a la semana.

```
aws opsworks start-instance --instance-id f705ee48-9000-4890-8bd3-20eb05825aaf
```

Salida: ninguna. Usa `describe-instances` para comprobar el estado de la instancia.

Consejo: puedes iniciar todas las instancias sin conexión de una pila con un comando llamando a `start-stack`.

### Más información

Para obtener más información, consulte Inicio, detención y reinicio manuales de instancias ininterrumpidas en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte la Referencia [StartInstance](#) de AWS CLI comandos.

## start-stack

En el siguiente ejemplo de código, se muestra cómo usar `start-stack`.

### AWS CLI

Para iniciar las instancias de una pila

En el siguiente ejemplo, se inician todas las instancias ininterrumpidas de una pila. Para iniciar una instancia concreta, usa `start-instance`.

```
aws opsworks --region us-east-1 start-stack --stack-id 8c428b08-a1a1-46ce-a5f8-feddc43771b8
```

Salida: ninguna.

### Más información

Para obtener más información, consulte Cómo iniciar una instancia en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [StartStack](#) la Referencia de AWS CLI comandos.

## stop-instance

En el siguiente ejemplo de código, se muestra cómo usar `stop-instance`.

### AWS CLI

Para detener una instancia

El siguiente ejemplo detiene una instancia específica, que se identifica por su ID de instancia. Para obtener un ID de instancia, ve a la página de detalles de la instancia en la AWS OpsWorks consola o ejecuta el `describe-instances` comando.

```
aws opsworks stop-instance --region us-east-1 --instance-id 3a21cfac-4a1f-4ce2-a921-b2cfba6f7771
```

Puedes reiniciar una instancia detenida llamando `start-instance` o eliminándola mediante una llamada `delete-instance`.

Resultado: ninguno.

Más información

Para obtener más información, consulte Detener una instancia en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [StopInstance](#) la Referencia de AWS CLI comandos.

## stop-stack

En el siguiente ejemplo de código, se muestra cómo usar `stop-stack`.

### AWS CLI

Para detener las instancias de una pila

En el siguiente ejemplo, se detienen todas las instancias ininterrumpidas de una pila. Para detener una instancia concreta, utilice `stop-instance`.

```
aws opsworks --region us-east-1 stop-stack --stack-id 8c428b08-a1a1-46ce-a5f8-feddc43771b8
```

Salida: Sin salida.

## Más información

Para obtener más información, consulte Detener una instancia en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [StopStack](#) la Referencia de AWS CLI comandos.

## **unassign-instance**

En el siguiente ejemplo de código, se muestra cómo usar `unassign-instance`.

### AWS CLI

Para anular la asignación de una instancia registrada de sus capas

El siguiente `unassign-instance` comando anula la asignación de una instancia de sus capas adjuntas.

```
aws opsworks --region us-east-1 unassign-instance --instance-id 4d6d1710-ded9-42a1-b08e-b043ad7af1e2
```

Salida: ninguna.

## Más información

Para obtener más información, consulte Anular la asignación de una instancia registrada en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte la Referencia [UnassignInstance](#) de AWS CLI comandos.

## **unassign-volume**

En el siguiente ejemplo de código, se muestra cómo usar `unassign-volume`.

### AWS CLI

Para anular la asignación de un volumen de su instancia

En el siguiente ejemplo, se anula la asignación de un volumen registrado de Amazon Elastic Block Store EBS (Amazon) de su instancia. El volumen se identifica por su ID de volumen, que



es el GUID que se AWS OpsWorks asigna al registrar el volumen en una pila, no por el ID de volumen de Amazon Elastic Compute Cloud (AmazonEC2).

```
aws opsworks --region us-east-1 unassign-volume --volume-id 8430177d-52b7-4948-9c62-e195af4703df
```

Salida: ninguna.

Más información

Para obtener más información, consulta [Cómo anular la asignación de Amazon EBS Volumes](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte la Referencia [UnassignVolume](#) de AWS CLI comandos.

## update-app

En el siguiente ejemplo de código, se muestra cómo usar `update-app`.

AWS CLI

Para actualizar una aplicación

En el siguiente ejemplo, se actualiza una aplicación específica para cambiarle el nombre.

```
aws opsworks --region us-east-1 update-app --app-id 26a61ead-d201-47e3-b55c-2a7c666942f8 --name NewAppName
```

Resultado: ninguno.

Más información

Para obtener más información, consulte [Edición de aplicaciones](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [UpdateApp](#) la Referencia de AWS CLI comandos.

## update-elastic-ip

En el siguiente ejemplo de código, se muestra cómo usar `update-elastic-ip`.

## AWS CLI

Para actualizar el nombre de una dirección IP elástica

El siguiente ejemplo actualiza el nombre de una dirección IP elástica especificada.

```
aws opsworks --region us-east-1 update-elastic-ip --elastic-ip 54.148.130.96 --  
name NewIPName
```

Resultado: ninguno.

Más información

Para obtener más información, consulte Administración de recursos en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [UpdateElasticIp](#) la Referencia de AWS CLI comandos.

## update-instance

En el siguiente ejemplo de código, se muestra cómo usar `update-instance`.

### AWS CLI

Para actualizar una instancia

En el siguiente ejemplo, se actualiza el tipo de instancia especificado.

```
aws opsworks --region us-east-1 update-instance --instance-  
id dfe18b02-5327-493d-91a4-c5c0c448927f --instance-type c3.xlarge
```

Salida: ninguna.

Más información

Para obtener más información, consulte Edición de la configuración de la instancia en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [UpdateInstance](#) la Referencia de AWS CLI comandos.

## update-layer

En el siguiente ejemplo de código, se muestra cómo usar `update-layer`.

### AWS CLI

Para actualizar una capa

En el siguiente ejemplo, se actualiza una capa específica para usar instancias EBS optimizadas para Amazon.

```
aws opsworks --region us-east-1 update-layer --layer-id 888c5645-09a5-4d0e-95a8-812ef1db76a4 --use-efs-optimized-instances
```

Salida: ninguna.

### Más información

Para obtener más información, consulte Edición de la configuración de una OpsWorks capa en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [UpdateLayer](#) la Referencia de AWS CLI comandos.

## update-my-user-profile

En el siguiente ejemplo de código, se muestra cómo usar `update-my-user-profile`.

### AWS CLI

Para actualizar el perfil de un usuario

En el siguiente ejemplo, se actualiza el perfil del `development` usuario para que utilice una clave SSH pública específica. Las AWS credenciales del usuario se representan mediante el `development` perfil del `credentials` archivo (`~\.aws\credentials`) y la clave se encuentra en un `.pem` archivo del directorio de trabajo.

```
aws opsworks --region us-east-1 --profile development update-my-user-profile --ssh-public-key file://development_key.pem
```

Resultado: ninguno.

## Más información

Para obtener más información, consulte Edición de la configuración del AWS OpsWorks usuario en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [UpdateMyUserProfile](#) la Referencia de AWS CLI comandos.

## **update-rds-db-instance**

En el siguiente ejemplo de código, se muestra cómo usar `update-rds-db-instance`.

### AWS CLI

Para actualizar una RDS instancia de base de datos de Amazon registrada

El siguiente ejemplo actualiza el valor de la contraseña maestra de una RDS instancia de Amazon. Tenga en cuenta que este comando no cambia la contraseña maestra de la RDS instancia, solo la contraseña que usted proporciona AWS OpsWorks. Si esta contraseña no coincide con la contraseña de la RDS instancia, la aplicación no podrá conectarse a la base de datos.

```
aws opsworks --region us-east-1 update-rds-db-instance --db-password 123456789
```

Salida: ninguna.

## Más información

Para obtener más información, consulte Registrar RDS instancias de Amazon con una pila en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [UpdateRdsDbInstance](#) la Referencia de AWS CLI comandos.

## **update-volume**

En el siguiente ejemplo de código, se muestra cómo usar `update-volume`.

### AWS CLI

Para actualizar un volumen registrado

El siguiente ejemplo actualiza el punto de montaje de un volumen registrado de Amazon Elastic Block Store (AmazonEBS). El volumen se identifica por su ID de volumen, que es el GUID que se AWS OpsWorks asigna al volumen cuando lo registras en una pila, no por el ID de volumen de Amazon Elastic Compute Cloud (AmazonEC2). :

```
aws opsworks --region us-east-1 update-volume --volume-id 8430177d-52b7-4948-9c62-e195af4703df --mount-point /mnt/myvol
```

Salida: ninguna.

### Más información

Para obtener más información, consulte Asignación de EBS volúmenes de Amazon a una instancia en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [UpdateVolume](#) la Referencia de AWS CLI comandos.

## AWS OpsWorks CM ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS OpsWorks CM.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

#### **associate-node**

En el siguiente ejemplo de código, se muestra cómo usar `associate-node`.

## AWS CLI

Para asociar nodos

El siguiente `associate-node` comando asocia un nodo `i-44de882p` con un nombre de servidor de Chef Automate `automate-06`, lo que significa que el `automate-06` servidor administra el nodo y comunica los comandos de receta al nodo a través del software de `chef-client` agente que se instala en el nodo mediante el comando `associate-node`. Los nombres de nodo válidos son EC2 instancia. IDs :

```
aws opsworks-cm associate-node --server-name "automate-06" --node-name "i-43de882p"
--engine-attributes "Name=CHEF_ORGANIZATION,Value='MyOrganization'
Name=CHEF_NODE_PUBLIC_KEY,Value='Public_key_contents'"
```

El resultado devuelto por el comando es similar al siguiente. Salida:

```
{
  "NodeAssociationStatusToken": "AHUY8wFe4pdXtZC5DiJa5S0Lp5o14DH//
rHRqHDWXxwVoNBxcEy4V7R0N0Fymh7E/1Hum0BPsemPQFE6dcGaiFk"
}
```

Más información

Para obtener más información, consulte [Añadir nodos automáticamente a Chef Automate](#) en la Guía del AWS OpsWorks usuario. AWS OpsWorks

- Para API obtener más información, consulte [AssociateNode](#) la Referencia de AWS CLI comandos.

## create-backup

En el siguiente ejemplo de código, se muestra cómo usar `create-backup`.

## AWS CLI

Para crear copias de seguridad

El siguiente `create-backup` comando inicia una copia de seguridad manual de un servidor de Chef Automate nombrado `automate-06` en la `us-east-1` región. El comando añade un mensaje descriptivo a la copia de seguridad del `--description` parámetro.

```
aws opsworks-cm create-backup \
  --server-name 'automate-06' \
  --description "state of my infrastructure at launch"
```

El resultado muestra información similar a la siguiente acerca de la nueva copia de seguridad.

Salida:

```
{
  "Backups": [
    {
      "BackupArn": "string",
      "BackupId": "automate-06-20160729133847520",
      "BackupType": "MANUAL",
      "CreatedAt": 2016-07-29T13:38:47.520Z,
      "Description": "state of my infrastructure at launch",
      "Engine": "Chef",
      "EngineModel": "Single",
      "EngineVersion": "12",
      "InstanceProfileArn": "arn:aws:iam::1019881987024:instance-profile/
automate-06-1010V4UU2WRM2",
      "InstanceType": "m4.large",
      "KeyPair": "",
      "PreferredBackupWindow": "",
      "PreferredMaintenanceWindow": "",
      "S3LogUrl": "https://s3.amazonaws.com/<bucket-name>/
automate-06-20160729133847520",
      "SecurityGroupIds": [ "sg-1a24c270" ],
      "ServerName": "automate-06",
      "ServiceRoleArn": "arn:aws:iam::1019881987024:role/aws-opsworks-cm-
service-role.1114810729735",
      "Status": "OK",
      "StatusDescription": "",
      "SubnetIds": [ "subnet-49436a18" ],
      "ToolsVersion": "string",
      "UserArn": "arn:aws:iam::1019881987024:user/opsworks-user"
    }
  ],
}
```

Para obtener más información, consulte [Hacer una copia de seguridad y restaurar un servidor de AWS OpsWorks For Chef Automate](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [CreateBackup](#) la Referencia de AWS CLI comandos.

## create-server

En el siguiente ejemplo de código, se muestra cómo usar `create-server`.

### AWS CLI

Para crear un servidor

El siguiente `create-server` ejemplo crea un nuevo servidor de Chef Automate con `automate-06` el nombre de su región predeterminada. Tenga en cuenta que los valores predeterminados se utilizan para la mayoría de las demás configuraciones, como el número de copias de seguridad que se deben conservar y las horas de inicio y mantenimiento de las copias de seguridad. Antes de ejecutar un `create-server` comando, complete los requisitos previos de la [Guía del usuario de Opsworks AWS OpsWorks for Chef Automate](#) en la Guía del usuario de AWS Opsworks for Chef Automate.

```
aws opsworks-cm create-server \  
  --engine "Chef" \  
  --engine-model "Single" \  
  --engine-version "12" \  
  --server-name "automate-06" \  
  --instance-profile-arn "arn:aws:iam::1019881987024:instance-profile/aws-opsworks-cm-ec2-role" \  
  --instance-type "t2.medium" \  
  --key-pair "amazon-test" \  
  --service-role-arn "arn:aws:iam::044726508045:role/aws-opsworks-cm-service-role"
```

El resultado muestra información similar a la siguiente sobre el nuevo servidor:

```
{  
  "Server": {  
    "BackupRetentionCount": 10,  
    "CreatedAt": 2016-07-29T13:38:47.520Z,  
    "DisableAutomatedBackup": FALSE,  
    "Endpoint": "https://opsworks-cm.us-east-1.amazonaws.com",  
    "Engine": "Chef",  
    "EngineAttributes": [  
      {
```



```

        "Name": "CHEF_DELIVERY_ADMIN_PASSWORD",
        "Value": "1Password1"
    }
],
"EngineModel": "Single",
"EngineVersion": "12",
"InstanceProfileArn": "arn:aws:iam::1019881987024:instance-profile/aws-opsworks-cm-ec2-role",
"InstanceType": "t2.medium",
"KeyPair": "amazon-test",
"MaintenanceStatus": "",
"PreferredBackupWindow": "Sun:02:00",
"PreferredMaintenanceWindow": "00:00",
"SecurityGroupIds": [ "sg-1a24c270" ],
"ServerArn": "arn:aws:iam::1019881987024:instance/automate-06-1010V4UU2WRM2",
"ServerName": "automate-06",
"ServiceRoleArn": "arn:aws:iam::1019881987024:role/aws-opsworks-cm-service-role",
>Status": "CREATING",
>StatusReason": "",
"SubnetIds": [ "subnet-49436a18" ]
}
}

```

Para obtener más información, consulte [UpdateServer](#) la API referencia AWS OpsWorks de Chef Automate.

- Para API obtener más información, consulte [CreateServer](#) la Referencia de AWS CLI comandos.

## delete-backup

En el siguiente ejemplo de código, se muestra cómo usar `delete-backup`.

### AWS CLI

Para eliminar copias de seguridad

El siguiente `delete-backup` comando elimina una copia de seguridad manual o automática de un servidor Chef Automate, identificada por el ID de la copia de seguridad. Este comando resulta útil cuando se acerca al número máximo de copias de seguridad que puede guardar o si desea minimizar los costes de almacenamiento en Amazon S3. :

```
aws opsworks-cm delete-backup --backup-id "automate-06-2016-11-19T23:42:40.240Z"
```

El resultado muestra si la eliminación de la copia de seguridad se ha realizado correctamente.

### Más información

Para obtener más información, consulte [Hacer una copia de seguridad y restaurar un servidor AWS OpsWorks para Chef Automate](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DeleteBackup](#) la Referencia de AWS CLI comandos.

## delete-server

En el siguiente ejemplo de código, se muestra cómo usar `delete-server`.

### AWS CLI

Para eliminar servidores

El siguiente `delete-server` comando elimina un servidor de Chef Automate, identificado por el nombre del servidor. Una vez eliminado el servidor, las `DescribeServer` solicitudes ya no lo devuelven. :

```
aws opsworks-cm delete-server --server-name "automate-06"
```

El resultado muestra si la eliminación del servidor se ha realizado correctamente.

### Más información

Para obtener más información, consulte [Eliminar un servidor AWS OpsWorks de Chef Automate](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DeleteServer](#) la Referencia de AWS CLI comandos.

## describe-account-attributes

En el siguiente ejemplo de código, se muestra cómo usar `describe-account-attributes`.

### AWS CLI

Descripción de atributos de cuenta

El siguiente `describe-account-attributes` comando devuelve información sobre el uso que su cuenta hace de los recursos AWS OpsWorks de Chef Automate. :

```
aws opsworks-cm describe-account-attributes
```

El resultado de cada entrada de atributo de cuenta devuelto por el comando es similar al siguiente. Salida:

```
{
  "Attributes": [
    {
      "Maximum": 5,
      "Name": "ServerLimit",
      "Used": 2
    }
  ]
}
```

### Más información

Para obtener más información, consulte `DescribeAccountAttributes` la APIreferencia AWS OpsWorks de Chef Automate.

- Para API obtener más información, consulte [DescribeAccountAttributes](#) la Referencia de AWS CLI comandos.

## describe-backups

En el siguiente ejemplo de código, se muestra cómo usar `describe-backups`.

### AWS CLI

Para describir las copias de seguridad

El siguiente `describe-backups` comando devuelve información sobre todas las copias de seguridad asociadas a su cuenta en la región predeterminada.

```
aws opsworks-cm describe-backups
```

El resultado de cada entrada de copia de seguridad devuelta por el comando es similar al siguiente.

**Salida:**

```
{
  "Backups": [
    {
      "BackupArn": "string",
      "BackupId": "automate-06-20160729133847520",
      "BackupType": "MANUAL",
      "CreatedAt": 2016-07-29T13:38:47.520Z,
      "Description": "state of my infrastructure at launch",
      "Engine": "Chef",
      "EngineModel": "Single",
      "EngineVersion": "12",
      "InstanceProfileArn": "arn:aws:iam::1019881987024:instance-profile/
automate-06-1010V4UU2WRM2",
      "InstanceType": "m4.large",
      "KeyPair": "",
      "PreferredBackupWindow": "",
      "PreferredMaintenanceWindow": "",
      "S3LogUrl": "https://s3.amazonaws.com/<bucket-name>/
automate-06-20160729133847520",
      "SecurityGroupIds": [ "sg-1a24c270" ],
      "ServerName": "automate-06",
      "ServiceRoleArn": "arn:aws:iam::1019881987024:role/aws-opsworks-cm-
service-role.1114810729735",
      "Status": "Successful",
      "StatusDescription": "",
      "SubnetIds": [ "subnet-49436a18" ],
      "ToolsVersion": "string",
      "UserArn": "arn:aws:iam::1019881987024:user/opsworks-user"
    }
  ],
}
```

Para obtener más información, consulte [Hacer una copia de seguridad y restaurar un servidor de AWS OpsWorks For Chef Automate](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribeBackups](#) la Referencia de AWS CLI comandos.

**describe-events**

En el siguiente ejemplo de código, se muestra cómo usar `describe-events`.

## AWS CLI

Para describir los eventos

El siguiente `describe-events` ejemplo devuelve información sobre todos los eventos asociados al servidor Chef Automate especificado.

```
aws opsworks-cm describe-events \  
  --server-name 'automate-06'
```

El resultado de cada entrada de evento devuelto por el comando es similar al siguiente ejemplo:

```
{  
  "ServerEvents": [  
    {  
      "CreatedAt": "2016-07-29T13:38:47.520Z",  
      "LogUrl": "https://s3.amazonaws.com/<bucket-name>/  
automate-06-20160729133847520",  
      "Message": "Updates successfully installed.",  
      "ServerName": "automate-06"  
    }  
  ]  
}
```

Para obtener más información, consulte los [consejos generales de solución de problemas](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DescribeEvents](#) la Referencia de AWS CLI comandos.

## **describe-node-association-status**

En el siguiente ejemplo de código, se muestra cómo usar `describe-node-association-status`.

## AWS CLI

Para describir el estado de la asociación de nodos

El siguiente `describe-node-association-status` comando devuelve el estado de una solicitud para asociar un nodo a un servidor de Chef Automate denominado `automate-06`. :

```
aws opsworks-cm describe-node-association-status --server-  
name "automate-06" --node-association-status-token "AflJKl+/  
GoKLZJBdDQEx0065CDi57b1Qe9nKM8joSok0pQ9xr8DqApBN9/106sLdSvlfDEKkEx+eoCHvrjoWHa0s="
```

El resultado de cada entrada de atributo de cuenta devuelto por el comando es similar al siguiente. Salida:

```
{  
  "NodeAssociationStatus": "IN_PROGRESS"  
}
```

### Más información

Para obtener más información, consulte `DescribeNodeAssociationStatus` la API referencia AWS OpsWorks de Chef Automate.

- Para API obtener más información, consulte [DescribeNodeAssociationStatus](#) la Referencia de AWS CLI comandos.

## describe-servers

En el siguiente ejemplo de código, se muestra cómo usar `describe-servers`.

### AWS CLI

Para describir los servidores

El siguiente `describe-servers` comando devuelve información sobre todos los servidores que están asociados a su cuenta y en su región predeterminada. :

```
aws opsworks-cm describe-servers
```

El resultado de cada entrada del servidor devuelto por el comando es similar al siguiente. Salida:

```
{  
  "Servers": [  
    {  
      "BackupRetentionCount": 8,  
      "CreatedAt": 2016-07-29T13:38:47.520Z,  
      "DisableAutomatedBackup": FALSE,  
      "Endpoint": "https://opsworks-cm.us-east-1.amazonaws.com",
```

```

    "Engine": "Chef",
    "EngineAttributes": [
      {
        "Name": "CHEF_DELIVERY_ADMIN_PASSWORD",
        "Value": "1Password1"
      }
    ],
    "EngineModel": "Single",
    "EngineVersion": "12",
    "InstanceProfileArn": "arn:aws:iam::1019881987024:instance-profile/
automate-06-1010V4UU2WRM2",
    "InstanceType": "m4.large",
    "KeyPair": "",
    "MaintenanceStatus": "SUCCESS",
    "PreferredBackupWindow": "03:00",
    "PreferredMaintenanceWindow": "Mon:09:00",
    "SecurityGroupIds": [ "sg-1a24c270" ],
    "ServerArn": "arn:aws:iam::1019881987024:instance/automate-06-1010V4UU2WRM2",
    "ServerName": "automate-06",
    "ServiceRoleArn": "arn:aws:iam::1019881987024:role/aws-opsworks-cm-service-
role.1114810729735",
    "Status": "HEALTHY",
    "StatusReason": "",
    "SubnetIds": [ "subnet-49436a18" ]
  }
]
}

```

## Más información

Para obtener más información, consulte `DescribeServers` la APIguía AWS OpsWorks de Chef Automate.

- Para API obtener más información, consulte [DescribeServers](#) la Referencia de AWS CLI comandos.

## **disassociate-node**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-node`.

### AWS CLI

Para desasociar nodos

El siguiente `disassociate-node` comando desasocia un nodo denominado `i-44de882p` y elimina el nodo de la administración por parte de un servidor de Chef Automate denominado `automate-06`. Los nombres de nodo válidos son EC2 instanciaIDs. :

```
aws opsworks-cm disassociate-node --server-name "automate-06" --node-
name "i-43de882p" --engine-attributes "Name=CHEF_ORGANIZATION,Value='MyOrganization'
Name=CHEF_NODE_PUBLIC_KEY,Value='Public_key_contents'"
```

El resultado devuelto por el comando es similar al siguiente. Salida:

```
{
  "NodeAssociationStatusToken": "AHUY8wFe4pdXtZC5DiJa5S0Lp5o14DH//
rHRqHDWxwVoNBxcEy4V7R0NOFymh7E/1Hum0BPsemPQFE6dcGaiFk"
}
```

### Más información

Para obtener más información, consulte [Eliminar un servidor AWS OpsWorks de Chef Automate](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [DisassociateNode](#) la Referencia de AWS CLI comandos.

## restore-server

En el siguiente ejemplo de código, se muestra cómo usar `restore-server`.

### AWS CLI

Para restaurar un servidor

El siguiente `restore-server` comando realiza una restauración in situ de un servidor de Chef Automate nombrado `automate-06` en su región predeterminada a partir de una copia de seguridad con un ID de `automate-06-2016-11-22T16:13:27.998Z`. Al restaurar un servidor, se restauran las conexiones a los nodos que el servidor de Chef Automate administraba en el momento en que se realizó la copia de seguridad especificada.

```
aws opsworks-cm restore-server --backup-id «Automate-06-2016-11-22T 16:13:27 .998 Z» --
nombre de servidor «automate-06»
```

El resultado es solo el ID del comando. Salida:



(None)

## Más información

Para obtener más información, consulte Restaurar un servidor de Chef Automate que ha fallado AWS OpsWorks en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [RestoreServer](#) la Referencia de AWS CLI comandos.

## start-maintenance

En el siguiente ejemplo de código, se muestra cómo usar `start-maintenance`.

### AWS CLI

Para iniciar el mantenimiento

El siguiente `start-maintenance` ejemplo inicia manualmente el mantenimiento en el servidor Chef Automate o Puppet Enterprise especificado en su región predeterminada. Este comando resulta útil si se ha producido un error en un intento de mantenimiento automatizado anterior y se ha resuelto la causa subyacente del fallo de mantenimiento.

```
aws opsworks-cm start-maintenance \  
  --server-name 'automate-06'
```

Salida:

```
{  
  "Server": {  
    "AssociatePublicIpAddress": true,  
    "BackupRetentionCount": 10,  
    "ServerName": "automate-06",  
    "CreatedAt": 1569229584.842,  
    "CloudFormationStackArn": "arn:aws:cloudformation:us-  
west-2:123456789012:stack/aws-opsworks-cm-instance-automate-06-1606611794746/  
EXAMPLE0-31de-11eb-bdb0-0a5b0a1353b8",  
    "DisableAutomatedBackup": false,  
    "Endpoint": "automate-06-EXAMPLEv8gjf5f.us-west-2.opsworks-cm.io",  
    "Engine": "ChefAutomate",  
    "EngineModel": "Single",  
    "EngineAttributes": [],
```

```

    "EngineVersion": "2020-07",
    "InstanceProfileArn": "arn:aws:iam::123456789012:instance-profile/aws-opsworks-cm-ec2-role",
    "InstanceType": "m5.large",
    "PreferredMaintenanceWindow": "Sun:01:00",
    "PreferredBackupWindow": "Sun:15:00",
    "SecurityGroupIds": [
        "sg-EXAMPLE"
    ],
    "ServiceRoleArn": "arn:aws:iam::123456789012:role/service-role/aws-opsworks-cm-service-role",
    "Status": "UNDER_MAINTENANCE",
    "SubnetIds": [
        "subnet-EXAMPLE"
    ],
    "ServerArn": "arn:aws:opsworks-cm:us-west-2:123456789012:server/automate-06/0148382d-66b0-4196-8274-d1a2b6dff8d1"
}
}

```

Para obtener más información, consulte [Mantenimiento del sistema \(servidores Puppet Enterprise\)](#) o [Mantenimiento del sistema \(servidores Chef Automate\)](#) en la Guía del AWS OpsWorks usuario.

- Para API obtener más información, consulte [StartMaintenance](#) la Referencia de AWS CLI comandos.

## update-server-engine-attributes

En el siguiente ejemplo de código, se muestra cómo usar `update-server-engine-attributes`.

### AWS CLI

Para actualizar los atributos del motor del servidor

El siguiente `update-server-engine-attributes` comando actualiza el valor del atributo del `CHEF_PIVOTAL_KEY` motor para un servidor de Chef Automate denominado `automate-06`. Actualmente no es posible cambiar el valor de otros atributos del motor.

```

aws opsworks-cm update-server-engine-attributes \
  --attribute-name CHEF_PIVOTAL_KEY \
  --attribute-value "new key value" \

```

```
--server-name "automate-06"
```

El resultado muestra información similar a la siguiente sobre el servidor actualizado.

```
{
  "Server": {
    "BackupRetentionCount": 2,
    "CreatedAt": "2016-07-29T13:38:47.520Z",
    "DisableAutomatedBackup": FALSE,
    "Endpoint": "https://opsworks-cm.us-east-1.amazonaws.com",
    "Engine": "Chef",
    "EngineAttributes": [
      {
        "Name": "CHEF_PIVOTAL_KEY",
        "Value": "new key value"
      }
    ],
    "EngineModel": "Single",
    "EngineVersion": "12",
    "InstanceProfileArn": "arn:aws:iam::1019881987024:instance-profile/
automate-06-1010V4UU2WRM2",
    "InstanceType": "m4.large",
    "KeyPair": "",
    "MaintenanceStatus": "SUCCESS",
    "PreferredBackupWindow": "Mon:09:15",
    "PreferredMaintenanceWindow": "03:00",
    "SecurityGroupIds": [ "sg-1a24c270" ],
    "ServerArn": "arn:aws:iam::1019881987024:instance/
automate-06-1010V4UU2WRM2",
    "ServerName": "automate-06",
    "ServiceRoleArn": "arn:aws:iam::1019881987024:role/aws-opsworks-cm-service-
role.1114810729735",
    "Status": "HEALTHY",
    "StatusReason": "",
    "SubnetIds": [ "subnet-49436a18" ]
  }
}
```

Para obtener más información, consulte [UpdateServerEngineAttributes](#) la API referencia AWS OpsWorks de Chef Automate.

- Para API obtener más información, consulte [UpdateServerEngineAttributes](#) la Referencia de AWS CLI comandos.

## update-server

En el siguiente ejemplo de código, se muestra cómo usar `update-server`.

### AWS CLI

Para actualizar un servidor

El siguiente `update-server` comando actualiza la hora de inicio del mantenimiento del servidor Chef Automate especificado en su región predeterminada. El `--preferred-maintenance-window` parámetro se agrega para cambiar el día y la hora de inicio del mantenimiento del servidor a los lunes a las 9:15 a.m. UTC. :

```
aws opsworks-cm update-server \  
  --server-name "automate-06" \  
  --preferred-maintenance-window "Mon:09:15"
```

El resultado muestra información similar a la siguiente sobre el servidor actualizado.

```
{  
  "Server": {  
    "BackupRetentionCount": 8,  
    "CreatedAt": 2016-07-29T13:38:47.520Z,  
    "DisableAutomatedBackup": TRUE,  
    "Endpoint": "https://opsworks-cm.us-east-1.amazonaws.com",  
    "Engine": "Chef",  
    "EngineAttributes": [  
      {  
        "Name": "CHEF_DELIVERY_ADMIN_PASSWORD",  
        "Value": "1Password1"  
      }  
    ],  
    "EngineModel": "Single",  
    "EngineVersion": "12",  
    "InstanceProfileArn": "arn:aws:iam::1019881987024:instance-profile/  
automate-06-1010V4UU2WRM2",  
    "InstanceType": "m4.large",  
    "KeyPair": "",  
    "MaintenanceStatus": "OK",  
    "PreferredBackupWindow": "Mon:09:15",  
    "PreferredMaintenanceWindow": "03:00",  
    "SecurityGroupIds": [ "sg-1a24c270" ],
```

```
    "ServerArn": "arn:aws:iam::1019881987024:instance/
automate-06-1010V4UU2WRM2",
    "ServerName": "automate-06",
    "ServiceRoleArn": "arn:aws:iam::1019881987024:role/aws-opsworks-cm-service-
role.1114810729735",
    "Status": "HEALTHY",
    "StatusReason": "",
    "SubnetIds": [ "subnet-49436a18" ]
  }
}
```

Para obtener más información, consulte [UpdateServer](#) la API referencia AWS OpsWorks de Chef Automate.

- Para API obtener más información, consulte [UpdateServer](#) la Referencia de AWS CLI comandos.

## Ejemplos de organizaciones que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de with AWS Command Line Interface Organizations.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **accept-handshake**

En el siguiente ejemplo de código, se muestra cómo usar accept-handshake.

AWS CLI

Para aceptar un apretón de manos de otra cuenta

Bill, el propietario de una organización, invitó previamente a la cuenta de Juan a unirse a su organización. En el siguiente ejemplo, se muestra la cuenta de Juan aceptando el apretón de manos y, por lo tanto, aceptando la invitación.

```
aws organizations accept-handshake --handshake-id h-examplehandshakeid111
```

En el resultado se observa lo siguiente:

```
{
  "Handshake": {
    "Action": "INVITE",
    "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
    "RequestedTimestamp": 1481656459.257,
    "ExpirationTimestamp": 1482952459.257,
    "Id": "h-examplehandshakeid111",
    "Parties": [
      {
        "Id": "o-exampleorgid",
        "Type": "ORGANIZATION"
      },
      {
        "Id": "juan@example.com",
        "Type": "EMAIL"
      }
    ],
    "Resources": [
      {
        "Resources": [
          {
            "Type": "MASTER_EMAIL",
            "Value": "bill@amazon.com"
          },
          {
            "Type": "MASTER_NAME",
            "Value": "Org Master Account"
          },
          {
            "Type": "ORGANIZATION_FEATURE_SET",
            "Value": "ALL"
          }
        ],
        "Type": "ORGANIZATION",

```

```

        "Value": "o-exampleorgid"
      },
      {
        "Type": "EMAIL",
        "Value": "juan@example.com"
      }
    ],
    "State": "ACCEPTED"
  }
}

```

- Para API obtener más información, consulte [AcceptHandshake](#) la Referencia de AWS CLI comandos.

## attach-policy

En el siguiente ejemplo de código, se muestra cómo usar `attach-policy`.

### AWS CLI

Asociación de una política a un nodo raíz, unidad organizativa o cuenta

#### Ejemplo 1

El siguiente ejemplo muestra cómo adjuntar una política de control de servicios (SCP) a una unidad organizativa:

```

aws organizations attach-policy
    --policy-id p-examplepolicyid111
    --target-id ou-examplerootid111-exampleoid111

```

#### Ejemplo 2

El siguiente ejemplo de código muestra cómo adjuntar una política de control de servicio directamente a una cuenta:

```

aws organizations attach-policy
    --policy-id p-examplepolicyid111
    --target-id 333333333333

```

- Para API obtener más información, consulte [AttachPolicy](#) la Referencia de AWS CLI comandos.

## cancel-handshake

En el siguiente ejemplo de código, se muestra cómo usar `cancel-handshake`.

### AWS CLI

Para cancelar un apretón de manos enviado desde otra cuenta

Anteriormente, Bill envió una invitación a la cuenta de Susan para unirse a su organización. Cambia de opinión y decide cancelar la invitación antes de que Susan la acepte. El siguiente ejemplo muestra la cancelación de Bill:

```
aws organizations cancel-handshake --handshake-id h-examplehandshakeid111
```

El resultado incluye un objeto de apretón de manos que muestra que el estado actual CANCELED es:

```
{
  "Handshake": {
    "Id": "h-examplehandshakeid111",
    "State": "CANCELED",
    "Action": "INVITE",
    "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
    "Parties": [
      {
        "Id": "o-exampleorgid",
        "Type": "ORGANIZATION"
      },
      {
        "Id": "susan@example.com",
        "Type": "EMAIL"
      }
    ],
    "Resources": [
      {
        "Type": "ORGANIZATION",
        "Value": "o-exampleorgid",
        "Resources": [
          {
            "Type": "MASTER_EMAIL",
            "Value": "bill@example.com"
          }
        ]
      }
    ]
  }
}
```



```

        {
            "Type": "MASTER_NAME",
            "Value": "Master Account"
        },
        {
            "Type": "ORGANIZATION_FEATURE_SET",
            "Value": "CONSOLIDATED_BILLING"
        }
    ]
},
{
    "Type": "EMAIL",
    "Value": "anika@example.com"
},
{
    "Type": "NOTES",
    "Value": "This is a request for Susan's account to
join Bob's organization."
}
],
"RequestedTimestamp": 1.47008383521E9,
"ExpirationTimestamp": 1.47137983521E9
}
}

```

- Para API obtener más información, consulte [CancelHandshake](#) la Referencia de AWS CLI comandos.

## create-account

En el siguiente ejemplo de código, se muestra cómo usar create-account.

### AWS CLI

Creación de una cuenta miembro que forme parte automáticamente de la organización

En el siguiente ejemplo se muestra cómo crear una cuenta miembro en una organización. La cuenta miembro se configura con el nombre Production Account y la dirección de correo electrónico susan@example.com. Organizations crea automáticamente un IAM rol con el nombre predeterminado de OrganizationAccountAccessRole porque el roleName parámetro no está especificado. Además, la configuración que permite a IAM los usuarios o roles con permisos suficientes acceder a los datos de facturación de la cuenta se establece en el valor

predeterminado de `ALLOW` porque el `iamUserAccessToBilling` parámetro no está especificado. Organizations envía automáticamente a Susan un correo electrónico de AWS bienvenida a:

```
aws organizations create-account --email susan@example.com --account-  
name "Production Account"
```

La salida incluye un objeto de solicitud que muestra que el estado ahora es `IN_PROGRESS`:

```
{  
  "CreateAccountStatus": {  
    "State": "IN_PROGRESS",  
    "Id": "car-examplecreateaccountrequestid111"  
  }  
}
```

Más adelante, puede consultar el estado actual de la solicitud proporcionando el valor de respuesta `Id` al `describe-create-account-status` comando como valor del `create-account-request-id` parámetro.

Para obtener más información, consulte [Crear una AWS cuenta en su organización](#) en la Guía del AWS usuario de Organizations.

- Para API obtener más información, consulte [CreateAccount](#) la Referencia de AWS CLI comandos.

## **create-organization**

En el siguiente ejemplo de código, se muestra cómo usar `create-organization`.

### AWS CLI

#### Ejemplo 1: Creación de una nueva organización

Bill quiere crear una organización con las credenciales de la cuenta 111111111111. En el siguiente ejemplo se muestra que la cuenta se convierte en la cuenta maestra de la nueva organización. Puesto que no especifica un conjunto de características, la nueva organización tiene habilitadas todas las características de forma predeterminada y las políticas de control de servicios están habilitadas en la raíz.

```
aws organizations create-organization
```

El resultado incluye un objeto de organización con detalles sobre la nueva organización:

```
{
  "Organization": {
    "AvailablePolicyTypes": [
      {
        "Status": "ENABLED",
        "Type": "SERVICE_CONTROL_POLICY"
      }
    ],
    "MasterAccountId": "111111111111",
    "MasterAccountArn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/111111111111",
    "MasterAccountEmail": "bill@example.com",
    "FeatureSet": "ALL",
    "Id": "o-exampleorgid",
    "Arn": "arn:aws:organizations::111111111111:organization/o-
exampleorgid"
  }
}
```

Ejemplo 2: Creación de una nueva organización con solo las características de facturación consolidada habilitadas

En el siguiente ejemplo se crea una organización que solo admite las características de facturación consolidada:

```
aws organizations create-organization --feature-set CONSOLIDATED_BILLING
```

El resultado incluye un objeto de organización con detalles sobre la nueva organización:

```
{
  "Organization": {
    "Arn": "arn:aws:organizations::111111111111:organization/o-
exampleorgid",
    "AvailablePolicyTypes": [],
    "Id": "o-exampleorgid",
    "MasterAccountArn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/111111111111",
    "MasterAccountEmail": "bill@example.com",
    "MasterAccountId": "111111111111",
    "FeatureSet": "CONSOLIDATED_BILLING"
  }
}
```

```
}  
}
```

Para obtener más información, consulte [Creación de una organización](#) en la Guía del usuario de AWS Organizations.

- Para API obtener más información, consulte [CreateOrganization](#) la Referencia de AWS CLI comandos.

## **create-organizational-unit**

En el siguiente ejemplo de código, se muestra cómo usar `create-organizational-unit`.

### AWS CLI

Creación de una unidad organizativa en raíz o UO principal

En el siguiente ejemplo se muestra cómo crear una unidad organizativa que se denomina AccountingOU:

```
aws organizations create-organizational-unit --parent-id r-examplerootid111 --  
name AccountingOU
```

El resultado incluye un `organizationalUnit` objeto con detalles sobre la nueva unidad organizativa:

```
{  
  "OrganizationalUnit": {  
    "Id": "ou-examplerootid111-exampleoid111",  
    "Arn": "arn:aws:organizations::111111111111:ou/o-exampleorgid/ou-  
examplerootid111-exampleoid111",  
    "Name": "AccountingOU"  
  }  
}
```

- Para API obtener más información, consulte [CreateOrganizationalUnit](#) la Referencia de AWS CLI comandos.

## **create-policy**

En el siguiente ejemplo de código, se muestra cómo usar `create-policy`.

## AWS CLI

Ejemplo 1: Para crear una política con un archivo fuente de texto para la JSON política

El siguiente ejemplo muestra cómo crear una política de control de servicios (SCP) denominada `AllowAllS3Actions`. El contenido de la política se extrae de un archivo del equipo local denominado `policy.json`.

```
aws organizations create-policy --content file://policy.json --
name AllowAllS3Actions, --type SERVICE_CONTROL_POLICY --description "Allows
delegation of all S3 actions"
```

La salida incluye un objeto de política con detalles sobre la nueva política:

```
{
  "Policy": {
    "Content": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
\\\"Allow\\\",\\\"Action\":[\\\"s3:*\\\"],\\\"Resource\":[\\\"*\\\"]}]}",
    "PolicySummary": {
      "Arn": "arn:aws:organizations::o-exampleorgid:policy/
service_control_policy/p-examplepolicyid111",
      "Description": "Allows delegation of all S3 actions",
      "Name": "AllowAllS3Actions",
      "Type": "SERVICE_CONTROL_POLICY"
    }
  }
}
```

Ejemplo 2: Para crear una política con una JSON política como parámetro

El siguiente ejemplo muestra cómo crearla SCP, esta vez incrustando el contenido de la política en forma de JSON cadena en el parámetro. La cadena debe aplicar una secuencia de escape con barras diagonales antes de las comillas dobles para garantizar que se traten como literales en el parámetro, que a su vez queda rodeado de comillas dobles:

```
aws organizations create-policy --content "{\"Version\":\"2012-10-17\",\"Statement
\":[{\"Effect\":\"Allow\",\"Action\":[\"s3:*\"],\"Resource\":[\"*\"]}]}" --
name AllowAllS3Actions --type SERVICE_CONTROL_POLICY --description "Allows
delegation of all S3 actions"
```

Para obtener más información sobre la creación y el uso de políticas en su organización, consulte Administración de políticas de la organización en la Guía del usuario de AWS Organizations.

- Para API obtener más información, consulte [CreatePolicy](#) la Referencia de AWS CLI comandos.

## decline-handshake

En el siguiente ejemplo de código, se muestra cómo usar `decline-handshake`.

### AWS CLI

Para rechazar un apretón de manos enviado desde otra cuenta

El siguiente ejemplo muestra que Susan, una administradora propietaria de la cuenta 222222222222, rechaza una invitación para unirse a la organización de Bill. La `DeclineHandshake` operación devuelve un objeto de apretón de manos que muestra que el estado actual es: `DECLINED`

```
aws organizations decline-handshake --handshake-id h-examplehandshakeid111
```

El resultado incluye un objeto de apretón de manos que muestra el nuevo estado de: `DECLINED`

```
{
  "Handshake": {
    "Id": "h-examplehandshakeid111",
    "State": "DECLINED",
    "Resources": [
      {
        "Type": "ORGANIZATION",
        "Value": "o-exampleorgid",
        "Resources": [
          {
            "Type": "MASTER_EMAIL",
            "Value": "bill@example.com"
          },
          {
            "Type": "MASTER_NAME",
            "Value": "Master Account"
          }
        ]
      }
    ],
    {
      "Type": "EMAIL",
      "Value": "susan@example.com"
    }
  }
}
```

```

        },
        {
            "Type": "NOTES",
            "Value": "This is an invitation to Susan's account
to join the Bill's organization."
        }
    ],
    "Parties": [
        {
            "Type": "EMAIL",
            "Id": "susan@example.com"
        },
        {
            "Type": "ORGANIZATION",
            "Id": "o-exampleorgid"
        }
    ],
    "Action": "INVITE",
    "RequestedTimestamp": 1470684478.687,
    "ExpirationTimestamp": 1471980478.687,
    "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111"
    }
}

```

- Para API obtener más información, consulte [DeclineHandshake](#) la Referencia de AWS CLI comandos.

## delete-organization

En el siguiente ejemplo de código, se muestra cómo usar `delete-organization`.

### AWS CLI

#### Eliminación de una organización

En el siguiente ejemplo se muestra cómo eliminar una organización. Para realizar esta operación, debe ser administrador de la cuenta maestra de la organización. En el ejemplo se supone que previamente eliminaste todas las cuentas y políticas de los miembros de la organización: OUs

```
aws organizations delete-organization
```

- Para API obtener más información, consulte [DeleteOrganization](#) la Referencia de AWS CLI comandos.

## **delete-organizational-unit**

En el siguiente ejemplo de código, se muestra cómo usar `delete-organizational-unit`.

AWS CLI

Eliminación de una unidad organizativa

En el ejemplo siguiente se muestra cómo se elimina una OU. En el ejemplo se supone que anteriormente se eliminaron todas las cuentas y otras OUs de la OU:

```
aws organizations delete-organizational-unit --organizational-unit-id ou-examplerootid111-exampleoid111
```

- Para API obtener más información, consulte [DeleteOrganizationalUnit](#) la Referencia de AWS CLI comandos.

## **delete-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-policy`.

AWS CLI

Eliminación de una política

En el siguiente ejemplo se muestra cómo eliminar una política de una organización. En el ejemplo se asume que anteriormente se ha desvinculado la política de todas las entidades:

```
aws organizations delete-policy --policy-id p-examplepolicyid111
```

- Para API obtener más información, consulte [DeletePolicy](#) la Referencia de AWS CLI comandos.

## **describe-account**

En el siguiente ejemplo de código, se muestra cómo usar `describe-account`.



## AWS CLI

Para obtener los detalles de una cuenta

En el siguiente ejemplo, se muestra cómo solicitar los detalles de una cuenta:

```
aws organizations describe-account --account-id 555555555555
```

El resultado muestra un objeto de cuenta con los detalles de la cuenta:

```
{
  "Account": {
    "Id": "555555555555",
    "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/555555555555",
    "Name": "Beta account",
    "Email": "anika@example.com",
    "JoinedMethod": "INVITED",
    "JoinedTimeStamp": 1481756563.134,
    "Status": "ACTIVE"
  }
}
```

- Para API obtener más información, consulte [DescribeAccount](#) la Referencia de AWS CLI comandos.

### **describe-create-account-status**

En el siguiente ejemplo de código, se muestra cómo usar `describe-create-account-status`.

## AWS CLI

Para obtener el estado más reciente de una solicitud de creación de una cuenta

El siguiente ejemplo muestra cómo solicitar el estado más reciente de una solicitud anterior para crear una cuenta en una organización. El `--request-id` especificado proviene de la respuesta de la llamada original a `create-account`. La solicitud de creación de cuenta muestra en el campo de estado que Organizations completó correctamente la creación de la cuenta.

Comando:

```
aws organizations describe-create-account-status --create-account-request-id car-examplecreateaccountrequestid111
```

Salida:

```
{
  "CreateAccountStatus": {
    "State": "SUCCEEDED",
    "AccountId": "555555555555",
    "AccountName": "Beta account",
    "RequestedTimestamp": 1470684478.687,
    "CompletedTimestamp": 1470684532.472,
    "Id": "car-examplecreateaccountrequestid111"
  }
}
```

- Para API obtener más información, consulte [DescribeCreateAccountStatus](#) la Referencia de AWS CLI comandos.

## describe-handshake

En el siguiente ejemplo de código, se muestra cómo usar `describe-handshake`.

AWS CLI

Para obtener información sobre un apretón de manos

En el siguiente ejemplo, se muestra cómo solicitar detalles sobre un apretón de manos. El identificador del apretón de manos proviene de la llamada original a `InviteAccountToOrganization` o de una llamada a `ListHandshakesForAccount` o: `ListHandshakesForOrganization`

```
aws organizations describe-handshake --handshake-id h-examplehandshakeid111
```

El resultado incluye un objeto de apretón de manos que contiene todos los detalles sobre el apretón de manos solicitado:

```
{
  "Handshake": {
```

```

    "Id": "h-examplehandshakeid111",
    "State": "OPEN",
    "Resources": [
      {
        "Type": "ORGANIZATION",
        "Value": "o-exampleorgid",
        "Resources": [
          {
            "Type": "MASTER_EMAIL",
            "Value": "bill@example.com"
          },
          {
            "Type": "MASTER_NAME",
            "Value": "Master Account"
          }
        ]
      },
      {
        "Type": "EMAIL",
        "Value": "anika@example.com"
      }
    ],
    "Parties": [
      {
        "Type": "ORGANIZATION",
        "Id": "o-exampleorgid"
      },
      {
        "Type": "EMAIL",
        "Id": "anika@example.com"
      }
    ],
    "Action": "INVITE",
    "RequestedTimestamp": 1470158698.046,
    "ExpirationTimestamp": 1471454698.046,
    "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111"
  }
}

```

- Para API obtener más información, consulte la Referencia [DescribeHandshake](#) de AWS CLI comandos.

## describe-organization

En el siguiente ejemplo de código, se muestra cómo usar `describe-organization`.

### AWS CLI

Para obtener información sobre la organización actual

El siguiente ejemplo muestra cómo solicitar detalles sobre una organización:

```
aws organizations describe-organization
```

El resultado incluye un objeto de organización que contiene los detalles de la organización:

```
{
  "Organization": {
    "MasterAccountArn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/111111111111",
    "MasterAccountEmail": "bill@example.com",
    "MasterAccountId": "111111111111",
    "Id": "o-exampleorgid",
    "FeatureSet": "ALL",
    "Arn": "arn:aws:organizations::111111111111:organization/o-
exampleorgid",
    "AvailablePolicyTypes": [
      {
        "Status": "ENABLED",
        "Type": "SERVICE_CONTROL_POLICY"
      }
    ]
  }
}
```

- Para API obtener más información, consulte [DescribeOrganization](#) la Referencia de AWS CLI comandos.

## describe-organizational-unit

En el siguiente ejemplo de código, se muestra cómo usar `describe-organizational-unit`.

### AWS CLI

Para obtener información sobre una unidad organizativa

En el siguiente `describe-organizational-unit` ejemplo se solicitan detalles sobre una OU.

```
aws organizations describe-organizational-unit \
  --organizational-unit-id ou-examplerootid111-exampleoid111
```

Salida:

```
{
  "OrganizationalUnit": {
    "Name": "Accounting Group",
    "Arn": "arn:aws:organizations::123456789012:ou/o-exampleorgid/ou-
  exemplerooid111-exampleoid111",
    "Id": "ou-examplerootid111-exampleoid111"
  }
}
```

- Para API obtener más información, consulte [DescribeOrganizationalUnit](#) la Referencia de AWS CLI comandos.

## describe-policy

En el siguiente ejemplo de código, se muestra cómo usar `describe-policy`.

### AWS CLI

Obtención de información acerca de una política

En el siguiente ejemplo se muestra cómo solicitar información acerca de una política:

```
aws organizations describe-policy --policy-id p-examplepolicyid111
```

El resultado incluye un objeto de política que contiene detalles acerca de la política:

```
{
  "Policy": {
    "Content": "{\n  \"Version\": \"2012-10-17\",\n  \"Statement\": [\n
    {\n      \"Effect\": \"Allow\",\n      \"Action\": \"*\",\n      \"Resource\":
    \"*\">\n    ]\n  }",
    "PolicySummary": {
      "Arn": "arn:aws:organizations::111111111111:policy/o-
    exampleorgid/service_control_policy/p-examplepolicyid111",
```

```

        "Type": "SERVICE_CONTROL_POLICY",
        "Id": "p-examplepolicyid111",
        "AwsManaged": false,
        "Name": "AllowAllS3Actions",
        "Description": "Enables admins to delegate S3 permissions"
    }
}
}

```

- Para API obtener más información, consulte [DescribePolicy](#) la Referencia de AWS CLI comandos.

## detach-policy

En el siguiente ejemplo de código, se muestra cómo usar `detach-policy`.

### AWS CLI

Desasociación de una política de un nodo raíz, unidad organizativa o cuenta

En el siguiente ejemplo se muestra cómo desasociar una política de una unidad organizativa:

```
aws organizations detach-policy --target-id ou-examplerootid111-exampleouid111 --policy-id p-examplepolicyid111
```

- Para API obtener más información, consulte [DetachPolicy](#) la Referencia de AWS CLI comandos.

## disable-policy-type

En el siguiente ejemplo de código, se muestra cómo usar `disable-policy-type`.

### AWS CLI

Para deshabilitar un tipo de política en una raíz

El siguiente ejemplo muestra cómo deshabilitar el tipo de política de control de servicios (SCP) en una raíz:

```
aws organizations disable-policy-type --root-id r-examplerootid111 --policy-type SERVICE_CONTROL_POLICY
```

El resultado muestra que el elemento de PolicyTypes respuesta ya no incluye SERVICE \_CONTROL\_POLICY:

```
{
  "Root": {
    "PolicyTypes": [],
    "Name": "Root",
    "Id": "r-examplerootid111",
    "Arn": "arn:aws:organizations::111111111111:root/o-exampleorgid/r-examplerootid111"
  }
}
```

- Para API obtener más información, consulte [DisablePolicyType](#) la Referencia de AWS CLI comandos.

## enable-all-features

En el siguiente ejemplo de código, se muestra cómo usar enable-all-features.

### AWS CLI

Para habilitar todas las funciones de una organización

En este ejemplo, se muestra al administrador pidiendo a todas las cuentas invitadas de la organización que aprueben la activación de todas las funciones de la organización. AWS Organizations envía un correo electrónico a la dirección registrada en cada cuenta de miembro invitado pidiéndole al propietario que apruebe el cambio en todas las funciones mediante la aceptación del apretón de manos enviado. Una vez que todas las cuentas de los miembros invitados hayan aceptado el apretón de manos, el administrador de la organización podrá finalizar el cambio en todas las funciones y, quienes dispongan de los permisos adecuados, OUs podrán crear políticas y aplicarlas a las raíces y las cuentas:

```
aws organizations enable-all-features
```

El resultado es un objeto de apretón de manos que se envía a todas las cuentas de los miembros invitados para su aprobación:

```
{
```

```

    "Handshake": {
      "Action": "ENABLE_ALL_FEATURES",
      "Arn": "arn:aws:organizations::111111111111:handshake/o-exampleorgid/enable_all_features/h-examplehandshakeid111",
      "ExpirationTimestamp": 1.483127868609E9,
      "Id": "h-examplehandshakeid111",
      "Parties": [
        {
          "id": "o-exampleorgid",
          "type": "ORGANIZATION"
        }
      ],
      "requestedTimestamp": 1.481831868609E9,
      "resources": [
        {
          "type": "ORGANIZATION",
          "value": "o-exampleorgid"
        }
      ],
      "state": "REQUESTED"
    }
  }
}

```

- Para API obtener más información, consulte [EnableAllFeatures](#) la Referencia de AWS CLI comandos.

## enable-policy-type

En el siguiente ejemplo de código, se muestra cómo usar `enable-policy-type`.

### AWS CLI

Para habilitar el uso de un tipo de política en una raíz

El siguiente ejemplo muestra cómo habilitar el tipo de política de control de servicios (SCP) en una raíz:

```
aws organizations enable-policy-type --root-id r-examplerootid111 --policy-type SERVICE_CONTROL_POLICY
```

El resultado muestra un objeto raíz con un elemento de `policyTypes` respuesta que indica que ahora SCPs están habilitados:



```
{
  "Root": {
    "PolicyTypes": [
      {
        "Status": "ENABLED",
        "Type": "SERVICE_CONTROL_POLICY"
      }
    ],
    "Id": "r-examplerootid111",
    "Name": "Root",
    "Arn": "arn:aws:organizations::111111111111:root/o-exampleorgid/r-examplerootid111"
  }
}
```

- Para API obtener más información, consulte [EnablePolicyType](#) la Referencia de AWS CLI comandos.

## invite-account-to-organization

En el siguiente ejemplo de código, se muestra cómo usar `invite-account-to-organization`.

### AWS CLI

Para invitar a una cuenta a unirse a una organización

El siguiente ejemplo muestra la cuenta maestra propiedad de `bill@example.com` que invita a la cuenta propiedad de `juan@example.com` a unirse a una organización:

```
aws organizations invite-account-to-organization --target '{"Type": "EMAIL", "Id": "juan@example.com"}' --notes "This is a request for Juan's account to join Bill's organization."
```

El resultado incluye una estructura de apretón de manos que muestra lo que se envía a la cuenta invitada:

```
{
  "Handshake": {
    "Action": "INVITE",
    "Arn": "arn:aws:organizations::111111111111:handshake/o-exampleorgid/invite/h-examplehandshakeid111",
  }
}
```

```
"ExpirationTimestamp": 1482952459.257,
"Id": "h-examplehandshakeid111",
"Parties": [
  {
    "Id": "o-exampleorgid",
    "Type": "ORGANIZATION"
  },
  {
    "Id": "juan@example.com",
    "Type": "EMAIL"
  }
],
"RequestedTimestamp": 1481656459.257,
"Resources": [
  {
    "Resources": [
      {
        "Type": "MASTER_EMAIL",
        "Value": "bill@amazon.com"
      },
      {
        "Type": "MASTER_NAME",
        "Value": "Org Master Account"
      },
      {
        "Type": "ORGANIZATION_FEATURE_SET",
        "Value": "FULL"
      }
    ],
    "Type": "ORGANIZATION",
    "Value": "o-exampleorgid"
  },
  {
    "Type": "EMAIL",
    "Value": "juan@example.com"
  }
],
"State": "OPEN"
}
```

- Para API obtener más información, consulte [InviteAccountToOrganization](#) la Referencia de AWS CLI comandos.

## leave-organization

En el siguiente ejemplo de código, se muestra cómo usar `leave-organization`.

### AWS CLI

Para dejar una organización como cuenta de miembro

En el siguiente ejemplo, se muestra al administrador de una cuenta de miembro que solicita abandonar la organización de la que es miembro actualmente:

```
aws organizations leave-organization
```

- Para API obtener más información, consulte [LeaveOrganization](#) la Referencia de AWS CLI comandos.

## list-accounts-for-parent

En el siguiente ejemplo de código, se muestra cómo usar `list-accounts-for-parent`.

### AWS CLI

Para recuperar una lista de todas las cuentas de una raíz o unidad organizativa principal especificada

El siguiente ejemplo muestra cómo solicitar una lista de las cuentas de una OU:

```
aws organizations list-accounts-for-parent --parent-id ou-examplerootid111-exampleouid111
```

La salida incluye una lista de objetos de resumen de cuenta.

```
{
  "Accounts": [
    {
      "Arn": "arn:aws:organizations::111111111111:account/o-exampleorgid/333333333333",
      "JoinedMethod": "INVITED",
      "JoinedTimestamp": 1481835795.536,
      "Id": "333333333333",
    }
  ]
}
```

```

        "Name": "Development Account",
        "Email": "juan@example.com",
        "Status": "ACTIVE"
    },
    {
        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/44444444444444",
        "JoinedMethod": "INVITED",
        "JoinedTimestamp": 1481835812.143,
        "Id": "44444444444444",
        "Name": "Test Account",
        "Email": "anika@example.com",
        "Status": "ACTIVE"
    }
]
}

```

- Para API obtener más información, consulte [ListAccountsForParent](#) la Referencia de AWS CLI comandos.

## list-accounts

En el siguiente ejemplo de código, se muestra cómo usar `list-accounts`.

### AWS CLI

Recuperación de una lista de todas las cuentas de una organización

En el siguiente ejemplo se muestra cómo solicitar una lista de las cuentas de una organización:

```
aws organizations list-accounts
```

La salida incluye una lista de objetos de resumen de cuenta.

```

{
    "Accounts": [
        {
            "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/111111111111",
            "JoinedMethod": "INVITED",
            "JoinedTimestamp": 1481830215.45,

```

```

        "Id": "111111111111",
        "Name": "Master Account",
        "Email": "bill@example.com",
        "Status": "ACTIVE"
    },
    {
        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/222222222222",
        "JoinedMethod": "INVITED",
        "JoinedTimestamp": 1481835741.044,
        "Id": "222222222222",
        "Name": "Production Account",
        "Email": "alice@example.com",
        "Status": "ACTIVE"
    },
    {
        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/333333333333",
        "JoinedMethod": "INVITED",
        "JoinedTimestamp": 1481835795.536,
        "Id": "333333333333",
        "Name": "Development Account",
        "Email": "juan@example.com",
        "Status": "ACTIVE"
    },
    {
        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/444444444444",
        "JoinedMethod": "INVITED",
        "JoinedTimestamp": 1481835812.143,
        "Id": "444444444444",
        "Name": "Test Account",
        "Email": "anika@example.com",
        "Status": "ACTIVE"
    }
]
}

```

- Para API obtener más información, consulte [ListAccounts](#) la Referencia de AWS CLI comandos.

## list-children

En el siguiente ejemplo de código, se muestra cómo usar `list-children`.

## AWS CLI

Para recuperar las cuentas secundarias y OUs de una unidad organizativa o raíz principal

En el siguiente ejemplo, se muestra cómo enumerar la raíz o la unidad organizativa que contiene esa cuenta 444444444444:

```
aws organizations list-children --child-type ORGANIZATIONAL_UNIT --parent-id ou-examplerootid111-exampleoid111
```

El resultado muestra los dos elementos secundarios OUs que contiene el padre:

```
{
  "Children": [
    {
      "Id": "ou-examplerootid111-exampleoid111",
      "Type": "ORGANIZATIONAL_UNIT"
    },
    {
      "Id": "ou-examplerootid111-exampleoid222",
      "Type": "ORGANIZATIONAL_UNIT"
    }
  ]
}
```

- Para API obtener más información, consulte [ListChildren](#) la Referencia de AWS CLI comandos.

## list-create-account-status

En el siguiente ejemplo de código, se muestra cómo usar `list-create-account-status`.

### AWS CLI

Ejemplo 1: Para recuperar una lista de las solicitudes de creación de cuentas realizadas en la organización actual

El siguiente ejemplo muestra cómo solicitar una lista de las solicitudes de creación de cuentas para una organización que se han completado correctamente:

```
aws organizations list-create-account-status --states SUCCEEDED
```

El resultado incluye una matriz de objetos con información sobre cada solicitud.

```
{
  "CreateAccountStatuses": [
    {
      "AccountId": "4444444444444444",
      "AccountName": "Developer Test Account",
      "CompletedTimeStamp": 1481835812.143,
      "Id": "car-examplecreateaccountrequestid111",
      "RequestedTimeStamp": 1481829432.531,
      "State": "SUCCEEDED"
    }
  ]
}
```

Ejemplo 2: Para recuperar una lista de las solicitudes de creación de cuentas en curso realizadas en la organización actual

En el siguiente ejemplo, se obtiene una lista de las solicitudes de creación de cuentas en curso para una organización:

```
aws organizations list-create-account-status --states IN_PROGRESS
```

El resultado incluye un conjunto de objetos con información sobre cada solicitud.

```
{
  "CreateAccountStatuses": [
    {
      "State": "IN_PROGRESS",
      "Id": "car-examplecreateaccountrequestid111",
      "RequestedTimeStamp": 1481829432.531,
      "AccountName": "Production Account"
    }
  ]
}
```

- Para API obtener más información, consulte [ListCreateAccountStatus](#) la Referencia de AWS CLI comandos.

## list-handshakes-for-account

En el siguiente ejemplo de código, se muestra cómo usar `list-handshakes-for-account`.

## AWS CLI

Para recuperar una lista de los apretones de manos enviados a una cuenta

El siguiente ejemplo muestra cómo obtener una lista de todos los apretones de manos asociados a la cuenta de las credenciales que se utilizaron para llamar a la operación:

```
aws organizations list-handshakes-for-account
```

El resultado incluye una lista de estructuras de apretón de manos con información sobre cada apretón de manos, incluido su estado actual:

```
{
  "Handshake": {
    "Action": "INVITE",
    "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
    "ExpirationTimestamp": 1482952459.257,
    "Id": "h-examplehandshakeid111",
    "Parties": [
      {
        "Id": "o-exampleorgid",
        "Type": "ORGANIZATION"
      },
      {
        "Id": "juan@example.com",
        "Type": "EMAIL"
      }
    ],
    "RequestedTimestamp": 1481656459.257,
    "Resources": [
      {
        "Resources": [
          {
            "Type": "MASTER_EMAIL",
            "Value": "bill@amazon.com"
          },
          {
            "Type": "MASTER_NAME",
            "Value": "Org Master Account"
          },
          {
            "Type": "ORGANIZATION_FEATURE_SET",
```



```

        "Value": "FULL"
      }
    ],
    "Type": "ORGANIZATION",
    "Value": "o-exampleorgid"
  },
  {
    "Type": "EMAIL",
    "Value": "juan@example.com"
  }
],
"State": "OPEN"
}
}

```

- Para API obtener más información, consulte la Referencia [ListHandshakesForAccount](#) de AWS CLI comandos.

## list-handshakes-for-organization

En el siguiente ejemplo de código, se muestra cómo usar `list-handshakes-for-organization`.

### AWS CLI

Para recuperar una lista de los apretones de manos asociados a una organización

El siguiente ejemplo muestra cómo obtener una lista de los apretones de manos asociados a la organización actual:

```
aws organizations list-handshakes-for-organization
```

El resultado muestra dos apretones de manos. El primero es una invitación a la cuenta de Juan y muestra un estado de OPEN La segunda es una invitación a la cuenta de Anika y muestra un estado deACCEPTED:

```

{
  "Handshakes": [
    {
      "Action": "INVITE",
      "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",

```

```

"ExpirationTimestamp": 1482952459.257,
"Id": "h-examplehandshakeid111",
"Parties": [
  {
    "Id": "o-exampleorgid",
    "Type": "ORGANIZATION"
  },
  {
    "Id": "juan@example.com",
    "Type": "EMAIL"
  }
],
"RequestedTimestamp": 1481656459.257,
"Resources": [
  {
    "Resources": [
      {
        "Type": "MASTER_EMAIL",
        "Value": "bill@amazon.com"
      },
      {
        "Type": "MASTER_NAME",
        "Value": "Org Master"
      }
    ],
    "Type": "ORGANIZATION_FEATURE_SET",
    "Value": "FULL"
  },
  {
    "Type": "ORGANIZATION",
    "Value": "o-exampleorgid"
  },
  {
    "Type": "EMAIL",
    "Value": "juan@example.com"
  },
  {
    "Type": "NOTES",
    "Value": "This is an invitation to Juan's
account to join Bill's organization."
  }
],

```

```

        "State": "OPEN"
    },
    {
        "Action": "INVITE",
        "State": "ACCEPTED",
        "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
        "ExpirationTimestamp": 1.471797437427E9,
        "Id": "h-examplehandshakeid222",
        "Parties": [
            {
                "Id": "o-exampleorgid",
                "Type": "ORGANIZATION"
            },
            {
                "Id": "anika@example.com",
                "Type": "EMAIL"
            }
        ],
        "RequestedTimestamp": 1.469205437427E9,
        "Resources": [
            {
                "Resources": [
                    {
                        "Type": "MASTER_EMAIL",
                        "Value": "bill@example.com"
                    },
                    {
                        "Type": "MASTER_NAME",
                        "Value": "Master Account"
                    }
                ],
                "Type": "ORGANIZATION",
                "Value": "o-exampleorgid"
            },
            {
                "Type": "EMAIL",
                "Value": "anika@example.com"
            },
            {
                "Type": "NOTES",
                "Value": "This is an invitation to Anika's
account to join Bill's organization."
            }
        ]
    }
}

```

```

    ]
  }
]
}

```

- Para API obtener más información, consulte [ListHandshakesForOrganization](#) la Referencia de AWS CLI comandos.

## list-organizational-units-for-parent

En el siguiente ejemplo de código, se muestra cómo usar `list-organizational-units-for-parent`.

### AWS CLI

Para recuperar una lista de OUs las unidades organizativas o raíz principales

El siguiente ejemplo muestra cómo obtener una lista de OUs en una raíz específica:

```
aws organizations list-organizational-units-for-parent --parent-id r-examplerootid111
```

El resultado muestra que la raíz especificada contiene dos OUs y muestra los detalles de cada una:

```

{
  "OrganizationalUnits": [
    {
      "Name": "AccountingDepartment",
      "Arn": "arn:aws:organizations::o-exampleorgid:ou/r-examplerootid111/ou-examplerootid111-exampleoid111"
    },
    {
      "Name": "ProductionDepartment",
      "Arn": "arn:aws:organizations::o-exampleorgid:ou/r-examplerootid111/ou-examplerootid111-exampleoid222"
    }
  ]
}

```

- Para API obtener más información, consulte [ListOrganizationalUnitsForParent](#) la Referencia de AWS CLI comandos.

## list-parents

En el siguiente ejemplo de código, se muestra cómo usar `list-parents`.

### AWS CLI

Para enumerar la matriz OUs o las raíces de una cuenta o unidad organizativa secundaria

En el siguiente ejemplo, se muestra cómo enumerar la unidad organizativa raíz o principal que contiene esa cuenta 444444444444:

```
aws organizations list-parents --child-id 444444444444
```

El resultado muestra que la cuenta especificada está en la OU con un ID especificado:

```
{
  "Parents": [
    {
      "Id": "ou-examplerootid111-exampleouid111",
      "Type": "ORGANIZATIONAL_UNIT"
    }
  ]
}
```

- Para API obtener más información, consulte [ListParents](#) la Referencia de AWS CLI comandos.

## list-policies-for-target

En el siguiente ejemplo de código, se muestra cómo usar `list-policies-for-target`.

### AWS CLI

Para recuperar una lista de los archivos SCPs adjuntos directamente a una cuenta

El siguiente ejemplo muestra cómo obtener una lista de todas las políticas de control de servicios (SCPs), tal como se especifica en el parámetro `Filter`, que están directamente asociadas a una cuenta:

```
aws organizations list-policies-for-target --filter SERVICE_CONTROL_POLICY --target-id 444444444444
```

El resultado incluye una lista de estructuras de políticas con información resumida sobre las políticas. La lista no incluye las políticas que se aplican a la cuenta porque se heredan de su ubicación en una jerarquía de unidades organizativas:

```
{
  "Policies": [
    {
      "Type": "SERVICE_CONTROL_POLICY",
      "Name": "AllowAllEC2Actions",
      "AwsManaged": false,
      "Id": "p-examplepolicyid222",
      "Arn": "arn:aws:organizations::o-exampleorgid:policy/service_control_policy/p-examplepolicyid222",
      "Description": "Enables account admins to delegate permissions for any EC2 actions to users and roles in their accounts."
    }
  ]
}
```

- Para API obtener más información, consulte [ListPoliciesForTarget](#) la Referencia de AWS CLI comandos.

## list-policies

En el siguiente ejemplo de código, se muestra cómo usar `list-policies`.

### AWS CLI

Recuperación de una lista de todas las políticas de una organización de un tipo determinado

El siguiente ejemplo muestra cómo obtener una lista de SCPs, tal como se especifica en el parámetro de filtro:

```
aws organizations list-policies --filter SERVICE_CONTROL_POLICY
```

La salida incluye una lista de políticas con información resumida:

```
{
  "Policies": [
    {
      "Type": "SERVICE_CONTROL_POLICY",
```

```

        "Name": "AllowAllS3Actions",
        "AwsManaged": false,
        "Id": "p-examplepolicyid111",
        "Arn": "arn:aws:organizations::111111111111:policy/
service_control_policy/p-examplepolicyid111",
        "Description": "Enables account admins to delegate
permissions for any S3 actions to users and roles in their accounts."
    },
    {
        "Type": "SERVICE_CONTROL_POLICY",
        "Name": "AllowAllEC2Actions",
        "AwsManaged": false,
        "Id": "p-examplepolicyid222",
        "Arn": "arn:aws:organizations::111111111111:policy/
service_control_policy/p-examplepolicyid222",
        "Description": "Enables account admins to delegate
permissions for any EC2 actions to users and roles in their accounts."
    },
    {
        "AwsManaged": true,
        "Description": "Allows access to every operation",
        "Type": "SERVICE_CONTROL_POLICY",
        "Id": "p-FullAWSAccess",
        "Arn": "arn:aws:organizations::aws:policy/
service_control_policy/p-FullAWSAccess",
        "Name": "FullAWSAccess"
    }
]
}

```

- Para API obtener más información, consulte [ListPolicies](#) la Referencia de AWS CLI comandos.

## list-roots

En el siguiente ejemplo de código, se muestra cómo usar `list-roots`.

### AWS CLI

Para recuperar una lista de las raíces de una organización

En este ejemplo, se muestra cómo obtener la lista de raíces de una organización:

```
aws organizations list-roots
```

El resultado incluye una lista de estructuras raíz con información resumida:

```
{
  "Roots": [
    {
      "Name": "Root",
      "Arn": "arn:aws:organizations::111111111111:root/o-
exampleorgid/r-examplerootid111",
      "Id": "r-examplerootid111",
      "PolicyTypes": [
        {
          "Status": "ENABLED",
          "Type": "SERVICE_CONTROL_POLICY"
        }
      ]
    }
  ]
}
```

- Para API obtener más información, consulte [ListRoots](#) la Referencia de AWS CLI comandos.

## list-targets-for-policy

En el siguiente ejemplo de código, se muestra cómo usar `list-targets-for-policy`.

### AWS CLI

Para recuperar una lista de las raíces y OUs las cuentas a las que está asociada una política

El siguiente ejemplo muestra cómo obtener una lista de las raíces y las cuentas a las que está asociada la política especificada: OUs

```
aws organizations list-targets-for-policy --policy-id p-FULLAWSAccess
```

El resultado incluye una lista de objetos adjuntos con información resumida sobre las raíces y las cuentas a las que está asociada la política: OUs

```
{
  "Targets": [
    {
      "Arn": "arn:aws:organizations::111111111111:root/o-
exampleorgid/r-examplerootid111",
```



```

        "Name": "Root",
        "TargetId": "r-examplerootid111",
        "Type": "ROOT"
    },
    {
        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/333333333333;",
        "Name": "Developer Test Account",
        "TargetId": "333333333333",
        "Type": "ACCOUNT"
    },
    {
        "Arn": "arn:aws:organizations::111111111111:ou/o-
exampleorgid/ou-examplerootid111-exampleouid111",
        "Name": "Accounting",
        "TargetId": "ou-examplerootid111-exampleouid111",
        "Type": "ORGANIZATIONAL_UNIT"
    }
]
}

```

- Para API obtener más información, consulte [ListTargetsForPolicy](#) la Referencia de AWS CLI comandos.

## move-account

En el siguiente ejemplo de código, se muestra cómo usar `move-account`.

### AWS CLI

Para mover una cuenta entre raíces o OUs

El siguiente ejemplo muestra cómo mover la cuenta maestra de la organización de la raíz a una unidad organizativa:

```
aws organizations move-account --account-id 333333333333 --source-parent-id r-
examplerootid111 --destination-parent-id ou-examplerootid111-exampleouid111
```

- Para API obtener más información, consulte [MoveAccount](#) la Referencia de AWS CLI comandos.

## remove-account-from-organization

En el siguiente ejemplo de código, se muestra cómo usar `remove-account-from-organization`.

### AWS CLI

Para eliminar una cuenta de una organización como cuenta maestra

El siguiente ejemplo muestra cómo eliminar una cuenta de una organización:

```
aws organizations remove-account-from-organization --account-id 333333333333
```

- Para API obtener más información, consulte [RemoveAccountFromOrganization](#) la Referencia de AWS CLI comandos.

## update-organizational-unit

En el siguiente ejemplo de código, se muestra cómo usar `update-organizational-unit`.

### AWS CLI

Para cambiar el nombre de una unidad organizativa

En este ejemplo se muestra cómo cambiar el nombre de una unidad organizativa: en este ejemplo, la unidad organizativa pasa a llamarse «AccountingOU»:

```
aws organizations update-organizational-unit --organizational-unit-id ou-examplerootid111-exampleoid111 --name AccountingOU
```

El resultado muestra el nuevo nombre:

```
{
  "OrganizationalUnit": {
    "Id": "ou-examplerootid111-exampleoid111"
    "Name": "AccountingOU",
    "Arn": "arn:aws:organizations::111111111111:ou/o-exampleorgid/ou-examplerootid111-exampleoid111"
  }
}
```

- Para API obtener más información, consulte [UpdateOrganizationalUnit](#) la Referencia de AWS CLI comandos.

## update-policy

En el siguiente ejemplo de código, se muestra cómo usar `update-policy`.

### AWS CLI

Ejemplo 1: cambiar el nombre de una política

En el siguiente `update-policy` ejemplo, se cambia el nombre de una política y se le da una nueva descripción.

```
aws organizations update-policy \
  --policy-id p-examplepolicyid111 \
  --name Renamed-Policy \
  --description "This description replaces the original."
```

El resultado muestra el nombre y la descripción nuevos.

```
{
  "Policy": {
    "Content": "{\n  \"Version\": \"2012-10-17\",\n  \"Statement\": {\n\n    \"Effect\": \"Allow\",\n    \"Action\": \"ec2:*\",\n    \"Resource\": \"*\"\n  }\n}\n",
    "PolicySummary": {
      "Id": "p-examplepolicyid111",
      "AwsManaged": false,
      "Arn": "arn:aws:organizations::111111111111:policy/o-exampleorgid/service_control_policy/p-examplepolicyid111",
      "Description": "This description replaces the original.",
      "Name": "Renamed-Policy",
      "Type": "SERVICE_CONTROL_POLICY"
    }
  }
}
```

Ejemplo 2: Para reemplazar el contenido de JSON texto de una política

El siguiente ejemplo muestra cómo reemplazar el JSON texto del SCP ejemplo anterior por una nueva cadena de texto de JSON política que permita S3 en lugar de EC2:

```
aws organizations update-policy \
  --policy-id p-examplepolicyid111 \
  --content "{\"Version\":\"2012-10-17\",\"Statement\":{\"Effect\":\"Allow\",
  \"Action\":\"s3:*\",\"Resource\":\"*\"}}"
```

El resultado muestra el nuevo contenido:

```
{
  "Policy": {
    "Content": "{ \"Version\": \"2012-10-17\", \"Statement\": { \"Effect\":
  \"Allow\", \"Action\": \"s3:*\", \"Resource\": \"*\" } }",
    "PolicySummary": {
      "Arn": "arn:aws:organizations::111111111111:policy/o-exampleorgid/
  service_control_policy/p-examplepolicyid111",
      "AwsManaged": false;
      "Description": "This description replaces the original.",
      "Id": "p-examplepolicyid111",
      "Name": "Renamed-Policy",
      "Type": "SERVICE_CONTROL_POLICY"
    }
  }
}
```

- Para API obtener más información, consulte [UpdatePolicy](#) la Referencia de AWS CLI comandos.

## AWS Outposts ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Outposts.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### get-outpost-instance-types

En el siguiente ejemplo de código, se muestra cómo usar `get-outpost-instance-types`.

#### AWS CLI

Para obtener los tipos de instancias de tu Outpost

En el siguiente `get-outpost-instance-types` ejemplo, se obtienen los tipos de instancia del Outpost especificado.

```
aws outposts get-outpost-instance-types \
  --outpost-id op-0ab23c4567EXAMPLE
```

Salida:

```
{
  "InstanceTypes": [
    {
      "InstanceType": "c5d.large"
    },
    {
      "InstanceType": "i3en.24xlarge"
    },
    {
      "InstanceType": "m5d.large"
    },
    {
      "InstanceType": "r5d.large"
    }
  ],
  "OutpostId": "op-0ab23c4567EXAMPLE",
  "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-0ab23c4567EXAMPLE"
}
```

Para obtener más información, consulta [Lanza una instancia en tu Outpost](#) en la Guía del usuario de AWS Outposts.

- Para API obtener más información, consulta la Referencia [GetOutpostInstanceTypes](#) de AWS CLI comandos.

## get-outpost

En el siguiente ejemplo de código, se muestra cómo usar `get-outpost`.

### AWS CLI

Para obtener los detalles de Outpost

En el siguiente `get-outpost` ejemplo, se muestran los detalles del Outpost especificado.

```
aws outposts get-outpost \  
  --outpost-id op-0ab23c4567EXAMPLE
```

Salida:

```
{  
  "Outpost": {  
    "OutpostId": "op-0ab23c4567EXAMPLE",  
    "OwnerId": "123456789012",  
    "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/  
op-0ab23c4567EXAMPLE",  
    "SiteId": "os-0ab12c3456EXAMPLE",  
    "Name": "EXAMPLE",  
    "LifecycleStatus": "ACTIVE",  
    "AvailabilityZone": "us-west-2a",  
    "AvailabilityZoneId": "usw2-az1",  
    "Tags": {}  
  }  
}
```

Para obtener más información, consulte [Trabajar con Outposts](#) en la Guía del usuario de AWS Outposts.

- Para API obtener más información, consulte la Referencia [GetOutpost](#) de AWS CLI comandos.

## list-outposts

En el siguiente ejemplo de código, se muestra cómo usar `list-outposts`.

### AWS CLI

Para enumerar Outposts

En el siguiente `list-outposts` ejemplo, se muestran los Outposts de tu AWS cuenta.

```
aws outposts list-outposts
```

Salida:

```
{
  "Outposts": [
    {
      "OutpostId": "op-0ab23c4567EXAMPLE",
      "OwnerId": "123456789012",
      "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-0ab23c4567EXAMPLE",
      "SiteId": "os-0ab12c3456EXAMPLE",
      "Name": "EXAMPLE",
      "Description": "example",
      "LifecycleStatus": "ACTIVE",
      "AvailabilityZone": "us-west-2a",
      "AvailabilityZoneId": "usw2-az1",
      "Tags": {
        "Name": "EXAMPLE"
      }
    },
    {
      "OutpostId": "op-4fe3dc21baEXAMPLE",
      "OwnerId": "123456789012",
      "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-4fe3dc21baEXAMPLE",
      "SiteId": "os-0ab12c3456EXAMPLE",
      "Name": "EXAMPLE2",
      "LifecycleStatus": "ACTIVE",
      "AvailabilityZone": "us-west-2a",
      "AvailabilityZoneId": "usw2-az1",
      "Tags": {}
    }
  ]
}
```

Para obtener más información, consulte [Trabajar con Outposts](#) en la Guía del usuario de AWS Outposts.

- Para API obtener más información, consulte la Referencia [ListOutposts](#) de AWS CLI comandos.

## list-sites

En el siguiente ejemplo de código, se muestra cómo usar `list-sites`.

### AWS CLI

Para enumerar sitios

En el siguiente `list-sites` ejemplo, se muestran los sitios de Outpost disponibles en tu AWS cuenta.

```
aws outposts list-sites
```

Salida:

```
{
  "Sites": [
    {
      "SiteId": "os-0ab12c3456EXAMPLE",
      "AccountId": "123456789012",
      "Name": "EXAMPLE",
      "Description": "example",
      "Tags": {}
    }
  ]
}
```

Para obtener más información, consulte [Trabajar con Outposts](#) en la Guía del usuario de AWS Outposts.

- Para API obtener más información, consulte la Referencia [ListSites](#) de AWS CLI comandos.

## AWS Payment Cryptography ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Payment Cryptography.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.



Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **create-alias**

En el siguiente ejemplo de código, se muestra cómo usar `create-alias`.

#### AWS CLI

Para crear un alias para una clave

En el siguiente `create-alias` ejemplo, se crea un alias para una clave.

```
aws payment-cryptography create-alias \  
  --alias-name alias/sampleAlias1 \  
  --key-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/  
  kwapwa6qaifllw2h
```

Salida:

```
{  
  "Alias": {  
    "AliasName": "alias/sampleAlias1",  
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/  
    kwapwa6qaifllw2h"  
  }  
}
```

Para obtener más información, consulte [Acerca de los alias](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte la Referencia [CreateAlias](#) de AWS CLI comandos.

### **create-key**

En el siguiente ejemplo de código, se muestra cómo usar `create-key`.

## AWS CLI

Para crear una clave

El siguiente `create-key` ejemplo genera una KEY TDES clave 2 que puede usar para generar y verificar CVV2 los valores CVV/.

```
aws payment-cryptography create-key \  
  --exportable \  
  --key-attributes KeyAlgorithm=TDES_2KEY, KeyUsage=TR31_C0_CARD_VERIFICATION_KEY,KeyClass=SYMMETRIC
```

Salida:

```
{  
  "Key": {  
    "CreateTimestamp": "1686800690",  
    "Enabled": true,  
    "Exportable": true,  
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/  
kwapwa6qaiifllw2h",  
    "KeyAttributes": {  
      "KeyAlgorithm": "TDES_2KEY",  
      "KeyClass": "SYMMETRIC_KEY",  
      "KeyModesOfUse": {  
        "Decrypt": false,  
        "DeriveKey": false,  
        "Encrypt": false,  
        "Generate": true,  
        "NoRestrictions": false,  
        "Sign": false,  
        "Unwrap": false,  
        "Verify": true,  
        "Wrap": false  
      },  
      "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"  
    },  
    "KeyCheckValue": "F2E50F",  
    "KeyCheckValueAlgorithm": "ANSI_X9_24",  
    "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",  
    "KeyState": "CREATE_COMPLETE",  
    "UsageStartTimestamp": "1686800690"  
  }  
}
```

```
}
```

Para obtener más información, consulte [Generación de claves](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [CreateKey](#) la Referencia de AWS CLI comandos.

## delete-alias

En el siguiente ejemplo de código, se muestra cómo usar `delete-alias`.

### AWS CLI

Para eliminar un alias

En el siguiente `delete-alias` ejemplo, se elimina un alias. No afecta a la clave.

```
aws payment-cryptography delete-alias \  
  --alias-name alias/sampleAlias1
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Acerca de los alias](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte la Referencia [DeleteAlias](#) de AWS CLI comandos.

## delete-key

En el siguiente ejemplo de código, se muestra cómo usar `delete-key`.

### AWS CLI

Para eliminar una clave

En el siguiente `delete-key` ejemplo, se programa la eliminación de una clave después de 7 días, que es el período de espera predeterminado.

```
aws payment-cryptography delete-key \  
  --key-identifier arn:aws:payment-cryptography:us-west-2:123456789012:key/  
  kwapwa6qaiFlLw2h
```

**Salida:**

```
{
  "Key": {
    "CreateTimestamp": "1686801198",
    "DeletePendingTimestamp": "1687405998",
    "Enabled": true,
    "Exportable": true,
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
kwapwa6qaifllw2h",
    "KeyAttributes": {
      "KeyAlgorithm": "TDES_2KEY",
      "KeyClass": "SYMMETRIC_KEY",
      "KeyModesOfUse": {
        "Decrypt": false,
        "DeriveKey": false,
        "Encrypt": false,
        "Generate": true,
        "NoRestrictions": false,
        "Sign": false,
        "Unwrap": false,
        "Verify": true,
        "Wrap": false
      },
      "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"
    },
    "KeyCheckValue": "F2E50F",
    "KeyCheckValueAlgorithm": "ANSI_X9_24",
    "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
    "KeyState": "DELETE_PENDING",
    "UsageStartTimestamp": "1686801190"
  }
}
```

Para obtener más información, consulte [Eliminar claves](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [DeleteKey](#) la Referencia de AWS CLI comandos.

**export-key**

En el siguiente ejemplo de código, se muestra cómo usar `export-key`.

## AWS CLI

Para exportar una clave

En el siguiente `export-key` ejemplo, se exporta una clave.

```
aws payment-cryptography export-key \
  --export-key-identifier arn:aws:payment-cryptography:us-west-2:123456789012:key/
lco3w6agsk7zgu2l \
  --key-material '{"Tr34KeyBlock": { \
    "CertificateAuthorityPublicKeyIdentifier": "arn:aws:payment-cryptography:us-  
west-2:123456789012:key/ftobshq7pvioc5fx", \
    "ExportToken": "export-token-cu4lg26ofcziixny", \
    "KeyBlockFormat": "X9_TR34_2012", \
    "WrappingKeyCertificate": file://wrapping-key-certificate.pem } }'
```

Contenidos de `wrapping-key-certificate.pem`:

```
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUV2VENDQXFZ0F3SUJBZ01SQ1ZZS8xMXFUK2svVz1RUDJQOE1V
```

Salida:

```
{
  "WrappedKey": {
    "KeyMaterial":
    "308205A106092A864886F70D010702A08205923082058E020101310D300B06096086480165030402013082031F
    "WrappedKeyMaterialFormat": "TR34_KEY_BLOCK"
  }
}
```

Para obtener más información, consulte [Exportación de claves](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [ExportKey](#) la Referencia de AWS CLI comandos.

## get-alias

En el siguiente ejemplo de código, se muestra cómo usar `get-alias`.

## AWS CLI

Para obtener un alias

El siguiente `get-alias` ejemplo devuelve ARN la clave asociada al alias.

```
aws payment-cryptography get-alias \  
  --alias-name alias/sampleAlias1
```

Salida:

```
{  
  "Alias": {  
    "AliasName": "alias/sampleAlias1",  
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/  
kwapwa6qaiif1lw2h"  
  }  
}
```

Para obtener más información, consulte [Acerca de los alias](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte la Referencia [GetAlias](#) de AWS CLI comandos.

## get-key

En el siguiente ejemplo de código, se muestra cómo usar `get-key`.

### AWS CLI

Para obtener los metadatos de una clave

El siguiente `get-key` ejemplo devuelve los metadatos de la clave asociada al alias. Esta operación no devuelve material criptográfico.

```
aws payment-cryptography get-key \  
  --key-identifier alias/sampleAlias1
```

Salida:

```
{  
  "Key": {  
    "CreateTimestamp": "1686800690",  
    "DeletePendingTimestamp": "1687405998",
```

```

    "Enabled": true,
    "Exportable": true,
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
kwapwa6qaifllw2h",
    "KeyAttributes": {
      "KeyAlgorithm": "TDES_2KEY",
      "KeyClass": "SYMMETRIC_KEY",
      "KeyModesOfUse": {
        "Decrypt": false,
        "DeriveKey": false,
        "Encrypt": false,
        "Generate": true,
        "NoRestrictions": false,
        "Sign": false,
        "Unwrap": false,
        "Verify": true,
        "Wrap": false
      },
      "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"
    },
    "KeyCheckValue": "F2E50F",
    "KeyCheckValueAlgorithm": "ANSI_X9_24",
    "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
    "KeyState": "DELETE_PENDING",
    "UsageStartTimestamp": "1686801190"
  }
}

```

Para obtener más información, consulte [Obtener claves](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [GetKey](#) la Referencia de AWS CLI comandos.

## get-parameters-for-export

En el siguiente ejemplo de código, se muestra cómo usar `get-parameters-for-export`.

### AWS CLI

Para inicializar el proceso de exportación

En el `get-parameters-for-export` ejemplo siguiente se genera un par de claves, se firma la clave y, a continuación, se devuelven el certificado y la raíz del certificado.

```
aws payment-cryptography get-parameters-for-export \
  --signing-key-algorithm RSA_2048 \
  --key-material-type TR34_KEY_BLOCK
```

Salida:

```
{
  "ExportToken": "export-token-ep5cwyzone7oya53",
  "ParametersValidUntilTimestamp": "1687415640",
  "SigningKeyAlgorithm": "RSA_2048",
  "SigningKeyCertificate":

  "MIICiTCCAfICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMakGA1UEBhMCMC
  VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
  b24xFDASBgNVBAStC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVxHmAd
  BgkqhkiG9w0BCQEWEG5vb25lQGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
  MTIwNDI0MjA0NTIxWjCBiDELMakGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYD
  VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAStC01BTSBDb25z
  b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVxHmAdBgkqhkiG9w0BCQEWEG5vb25lQGft
  YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
  21uUSfwfEvySwTC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
  rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
  Ibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
  nUHVvXyUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
  FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStB
  NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=",
  "SigningKeyCertificateChain":
  "MIICiTCCAfICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMakGA1UEBhMCMC
  VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
  b24xFDASBgNVBAStC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVxHmAd
  BgkqhkiG9w0BCQEWEG5vb25lQGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
  MTIwNDI0MjA0NTIxWjCBiDELMakGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYD
  VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAStC01BTSBDb25z
  b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWVxHmAdBgkqhkiG9w0BCQEWEG5vb25lQGft
  YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
  21uUSfwfEvySwTC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
  rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
  Ibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
  nUHVvXyUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
  FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStB
  NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE="
}
```



Para obtener más información, consulte [Exportación de claves](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [GetParametersForExport](#) la Referencia de AWS CLI comandos.

## get-parameters-for-import

En el siguiente ejemplo de código, se muestra cómo usar `get-parameters-for-import`.

### AWS CLI

Para inicializar el proceso de importación

En el `get-parameters-for-import` ejemplo siguiente se genera un par de claves, se firma la clave y, a continuación, se devuelven el certificado y la raíz del certificado.

```
aws payment-cryptography get-parameters-for-import \
  --key-material-type TR34_KEY_BLOCK \
  --wrapping-key-algorithm RSA_2048
```

Salida:

```
{
  "ImportToken": "import-token-qgmafpa7nt2kfbb",
  "ParametersValidUntilTimestamp": "1687415640",
  "WrappingKeyAlgorithm": "RSA_2048",
  "WrappingKeyCertificate":
  "MIICiTCCAfICCD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMakGA1UEBhMC
  VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
  b24xFDASBgNVBAwTC0lBTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAd
  BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
  MTIwNDI0MjA0NTIxWjCBiDELMakGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
  VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAwTC0lBTSBDb25z
  b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAdBgkqhkiG9w0BCQEWEG5vb251QGft
  YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
  21uUSfwfEvySWtC2XADZ4nB+BLygVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
  rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
  Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
  nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
  FFBjvSfpJI1LJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjStB
  NYiytVbZPQUQ5Yaxu2jXnimv3rrszlaEXAMPLE=",
  "WrappingKeyCertificateChain":
```

```

"NIICiTCCAfICCQD6m7oRw0uX0jANBqkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAdDgYDVQQHEwDTZWf0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBA5TC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWxhZAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAdDgYD
VQQHEwDTZWf0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC01BTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTc2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcVQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUHVvXyUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJ10ZxBHjJnyp3780D8uTs7fLvJx79LjStb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE="
}

```

Para obtener más información, consulte [Importación de claves](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [GetParametersForImport](#) la Referencia de AWS CLI comandos.

## get-public-key-certificate

En el siguiente ejemplo de código, se muestra cómo usar `get-public-key-certificate`.

### AWS CLI

Para devolver la clave pública

El siguiente `get-public-key-certificate` ejemplo devuelve la parte de clave pública de un par de claves.

```

aws payment-cryptography get-public-key-certificate \
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaiFlLw2h

```

Salida:

```

{
  "KeyCertificate":
    "MIICiTCCAfICCQD6m7oRw0uX0jANBqkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC

```

```

VVMxCzAJBgNVBAGTAldBMRAwDgYDVQHQEwdTZWF0dGx1MQ8wDQYDVQKKEwZBbWF6
b24xFDASBgNVBAStC01BTSBDb25zb2x1MRIwEAYDVQKQDEwLUZXN0Q21sYWxHZAAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAKGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQHQEwdTZWF0dGx1MQ8wDQYDVQKKEwZBbWF6b24xFDASBgNVBAStC01BTSBDb25z
b2x1MRIwEAYDVQKQDEwLUZXN0Q21sYWxHZAAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZncvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVvxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFbjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=",
"KeyCertificateChain":
"NIICiTCCAfICCD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAKGA1UEBhMCM
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQHQEwdTZWF0dGx1MQ8wDQYDVQKKEwZBbWF6
b24xFDASBgNVBAStC01BTSBDb25zb2x1MRIwEAYDVQKQDEwLUZXN0Q21sYWxHZAAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAKGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQHQEwdTZWF0dGx1MQ8wDQYDVQKKEwZBbWF6b24xFDASBgNVBAStC01BTSBDb25z
b2x1MRIwEAYDVQKQDEwLUZXN0Q21sYWxHZAAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZncvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVvxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFbjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE="
}

```

Para obtener más información, consulte [Obtener la clave pública o el certificado asociado a un par de claves](#) en la Guía del usuario de criptografía de AWS pagos.

- Para API obtener más información, consulte la Referencia de [GetPublicKeyCertificate](#) comandos AWS CLI .

## import-key

En el siguiente ejemplo de código, se muestra cómo usar import-key.

### AWS CLI

Para importar una clave TR-34

En el siguiente `import-key` ejemplo, se importa una clave TR-34.

```
aws payment-cryptography import-key \
  --key-material='{ "Tr34KeyBlock": {" \
    CertificateAuthorityPublicKeyIdentifier": "arn:aws:payment-
cryptology:us-west-2:123456789012:key/rmm5wn2q564nijnm", \
    "ImportToken": "import-token-5ott6ho5nts7bbc", \
    "KeyBlockFormat": "X9_TR34_2012", \
    "SigningKeyCertificate": file://signing-key-certificate.pem, \
    "WrappedKeyBlock": file://wrapped-key-block.pem } }'
```

Contenidos de `signing-key-certificate.pem`:

```
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUV2RENDQXFTZ0F3SUJBZ01RYWVCK25IbE1WZU1PR1ZiNjU1Q2Jz
```

Contenidos de `wrapped-key-block.pem`:

```
3082059806092A864886F70D010702A082058930820585020101310D300B06096086480165030402013082031606
```

Salida:

```
{
  "Key": {
    "CreateTimestamp": "2023-06-09T16:56:27.621000-07:00",
    "Enabled": true,
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
bzmvgyx dg3sktwxd",
    "KeyAttributes": {
      "KeyAlgorithm": "TDES_2KEY",
      "KeyClass": "SYMMETRIC_KEY",
      "KeyModesOfUse": {
        "Decrypt": false,
        "DeriveKey": false,
        "Encrypt": false,
        "Generate": true,
        "NoRestrictions": false,
        "Sign": false,
        "Unwrap": false,
        "Verify": true,
        "Wrap": false
      },
    },
    "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"
```

```
    },
    "KeyCheckValue": "D9B20E",
    "KeyCheckValueAlgorithm": "ANSI_X9_24",
    "KeyOrigin": "EXTERNAL",
    "KeyState": "CREATE_COMPLETE",
    "UsageStartTimestamp": "2023-06-09T16:56:27.621000-07:00"
  }
}
```

Para obtener más información, consulte [Importación de claves](#) en la Guía del usuario de criptografía AWS de pagos.

- Para API obtener más información, consulte [ImportKey](#) la Referencia de AWS CLI comandos.

## list-aliases

En el siguiente ejemplo de código, se muestra cómo usar `list-aliases`.

### AWS CLI

Para obtener una lista de alias

En el siguiente `list-aliases` ejemplo, se muestran todos los alias de su cuenta en esta región.

```
aws payment-cryptography list-aliases
```

Salida:

```
{
  "Aliases": [
    {
      "AliasName": "alias/sampleAlias1",
      "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/kwapwa6qaifllw2h"
    },
    {
      "AliasName": "alias/sampleAlias2",
      "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/kwapwa6qaifllw2h"
    }
  ]
}
```

Para obtener más información, consulte [Acerca de los alias](#) en la Guía del usuario de criptografía AWS de pagos.

- Para API obtener más información, consulte la Referencia [ListAliases](#) de AWS CLI comandos.

## list-keys

En el siguiente ejemplo de código, se muestra cómo usar `list-keys`.

### AWS CLI

Para obtener una lista de claves

En el siguiente `list-keys` ejemplo, se muestran todas las claves de su cuenta en esta región.

```
aws payment-cryptography list-keys
```

Salida:

```
{
  "Keys": [
    {
      "CreateTimestamp": "1666506840",
      "Enabled": false,
      "Exportable": true,
      "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaiifllw2h",
      "KeyAttributes": {
        "KeyAlgorithm": "TDES_3KEY",
        "KeyClass": "SYMMETRIC_KEY",
        "KeyModesOfUse": {
          "Decrypt": true,
          "DeriveKey": false,
          "Encrypt": true,
          "Generate": false,
          "NoRestrictions": false,
          "Sign": false,
          "Unwrap": true,
          "Verify": false,
          "Wrap": true
        },
        "KeyUsage": "TR31_P1_PIN_GENERATION_KEY"
      },
    },
  ],
}
```

```

    "KeyCheckValue": "369D",
    "KeyCheckValueAlgorithm": "ANSI_X9_24",
    "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
    "KeyState": "CREATE_COMPLETE",
    "UsageStopTimestamp": "1666938840"
  }
]
}

```

Para obtener más información, consulte [Listar claves](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [ListKeys](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para obtener la lista de etiquetas de una clave

En el siguiente `list-tags-for-resource` ejemplo, se obtienen las etiquetas de una clave.

```

aws payment-cryptography list-tags-for-resource \
  --resource-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaiFlLw2h

```

Salida:

```

{
  "Tags": [
    {
      "Key": "BIN",
      "Value": "20151120"
    },
    {
      "Key": "Project",
      "Value": "Production"
    }
  ]
}

```

Para obtener más información, consulte [Administrar etiquetas clave con API operaciones](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## restore-key

En el siguiente ejemplo de código, se muestra cómo usar `restore-key`.

### AWS CLI

Para restaurar una clave cuya eliminación está programada

El siguiente `restore-key` ejemplo cancela la eliminación de una clave.

```
aws payment-cryptography restore-key \  
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/  
  kwapwa6qaifllw2h
```

Salida:

```
{  
  "Key": {  
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/  
kwapwa6qaifllw2h",  
    "KeyAttributes": {  
      "KeyUsage": "TR31_V2_VISA_PIN_VERIFICATION_KEY",  
      "KeyClass": "SYMMETRIC_KEY",  
      "KeyAlgorithm": "TDES_3KEY",  
      "KeyModesOfUse": {  
        "Encrypt": false,  
        "Decrypt": false,  
        "Wrap": false,  
        "Unwrap": false,  
        "Generate": true,  
        "Sign": false,  
        "Verify": true,  
        "DeriveKey": false,  
        "NoRestrictions": false  
      }  
    }  
  },  
  "KeyCheckValue": "",  
}
```



```

    "KeyCheckValueAlgorithm": "ANSI_X9_24",
    "Enabled": false,
    "Exportable": true,
    "KeyState": "CREATE_COMPLETE",
    "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
    "CreateTimestamp": "1686800690",
    "UsageStopTimestamp": "1687405998"
  }
}

```

Para obtener más información, consulte [Eliminar claves](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [RestoreKey](#) la Referencia de AWS CLI comandos.

## start-key-usage

En el siguiente ejemplo de código, se muestra cómo usar `start-key-usage`.

### AWS CLI

Para activar una clave

El siguiente `start-key-usage` ejemplo permite utilizar una clave.

```

aws payment-cryptography start-key-usage \
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaiFl1w2h

```

Salida:

```

{
  "Key": {
    "CreateTimestamp": "1686800690",
    "Enabled": true,
    "Exportable": true,
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/alsuwfxug3pgy6xh",
    "KeyAttributes": {
      "KeyAlgorithm": "TDES_3KEY",
      "KeyClass": "SYMMETRIC_KEY",
      "KeyModesOfUse": {
        "Decrypt": true,

```

```

        "DeriveKey": false,
        "Encrypt": true,
        "Generate": false,
        "NoRestrictions": false,
        "Sign": false,
        "Unwrap": true,
        "Verify": false,
        "Wrap": true
    },
    "KeyUsage": "TR31_P1_PIN_GENERATION_KEY"
},
"KeyCheckValue": "369D",
"KeyCheckValueAlgorithm": "ANSI_X9_24",
"KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
"KeyState": "CREATE_COMPLETE",
"UsageStartTimestamp": "1686800690"
}
}

```

Para obtener más información, consulte [Habilitar y deshabilitar claves](#) en la Guía del usuario de criptografía de AWS pagos.

- Para API obtener más información, consulte la Referencia [StartKeyUsage](#) de AWS CLI comandos.

## stop-key-usage

En el siguiente ejemplo de código, se muestra cómo usar stop-key-usage.

### AWS CLI

Para deshabilitar una clave

En el siguiente stop-key-usage ejemplo, se desactiva una clave.

```

aws payment-cryptography stop-key-usage \
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaif1lw2h

```

Salida:

```
{
```

```
"Key": {
  "CreateTimestamp": "1686800690",
  "Enabled": true,
  "Exportable": true,
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
alsuwfxug3pgy6xh",
  "KeyAttributes": {
    "KeyAlgorithm": "TDES_3KEY",
    "KeyClass": "SYMMETRIC_KEY",
    "KeyModesOfUse": {
      "Decrypt": true,
      "DeriveKey": false,
      "Encrypt": true,
      "Generate": false,
      "NoRestrictions": false,
      "Sign": false,
      "Unwrap": true,
      "Verify": false,
      "Wrap": true
    },
    "KeyUsage": "TR31_P1_PIN_GENERATION_KEY"
  },
  "KeyCheckValue": "369D",
  "KeyCheckValueAlgorithm": "ANSI_X9_24",
  "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
  "KeyState": "CREATE_COMPLETE",
  "UsageStartTimestamp": "1686800690"
}
}
```

Para obtener más información, consulte [Habilitar y deshabilitar las claves](#) en la Guía del usuario de criptografía AWS de pagos.

- Para API obtener más información, consulte la Referencia [StopKeyUsage](#) de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar una clave

En el siguiente `tag-resource` ejemplo, se etiqueta una clave.

```
aws payment-cryptography tag-resource \  
  --resource-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/  
kwapwa6qaifllw2h \  
  --tags Key=sampleTag,Value=sampleValue
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de etiquetas clave](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## **untag-resource**

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar una etiqueta de una clave

En el siguiente `untag-resource` ejemplo, se quita una etiqueta de una clave.

```
aws payment-cryptography untag-resource \  
  --resource-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/  
kwapwa6qaifllw2h \  
  --tag-keys sampleTag
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración de etiquetas clave](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## **update-alias**

En el siguiente ejemplo de código, se muestra cómo usar `update-alias`.

## AWS CLI

Para actualizar un alias

En el siguiente `update-alias` ejemplo, se asocia el alias a una clave diferente.

```
aws payment-cryptography update-alias \  
  --alias-name alias/sampleAlias1 \  
  --key-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/  
tqv5yij6wtxx64pi
```

Salida:

```
{  
  "Alias": {  
    "AliasName": "alias/sampleAlias1",  
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/  
tqv5yij6wtxx64pi "  
  }  
}
```

Para obtener más información, consulte [Acerca de los alias](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte la Referencia [UpdateAlias](#) de AWS CLI comandos.

## AWS Payment Cryptography Ejemplos de planos de datos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el AWS Command Line Interface uso de AWS Payment Cryptography Data Plane.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **decrypt-data**

En el siguiente ejemplo de código, se muestra cómo usar `decrypt-data`.

#### AWS CLI

Para descifrar el texto cifrado

En el siguiente `decrypt-data` ejemplo, se descifran los datos de texto cifrado mediante una clave simétrica. Para esta operación, la clave debe estar establecida en `KeyModesOfUse` establecida en `Decrypt KeyUsage TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY`

```
aws payment-cryptography-data decrypt-data \  
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/  
  kwapwa6qaifllw2h \  
  --cipher-text 33612AB9D6929C3A828EB6030082B2BD \  
  --decryption-attributes 'Symmetric={Mode=CBC}'
```

Salida:

```
{  
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/  
  kwapwa6qaifllw2h",  
  "KeyCheckValue": "71D7AE",  
  "PlainText": "31323334313233343132333431323334"  
}
```

Para obtener más información, consulte [Descifrar datos](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte la Referencia [DecryptData](#) de AWS CLI comandos.

### **encrypt-data**

En el siguiente ejemplo de código, se muestra cómo usar `encrypt-data`.

## AWS CLI

Para cifrar datos

En el siguiente `encrypt-data` ejemplo, se cifran los datos de texto sin formato mediante una clave simétrica. Para esta operación, la clave debe estar establecida en `Encrypt` y `KeyModesOfUse` `KeyUsage` establecida en `TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY`

```
aws payment-cryptography-data encrypt-data \  
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/  
  kwapwa6qaiFlw2h \  
  --plain-text 31323334313233343132333431323334 \  
  --encryption-attributes 'Symmetric={Mode=CBC}'
```

Salida:

```
{  
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/  
  kwapwa6qaiFlw2h",  
  "KeyCheckValue": "71D7AE",  
  "CipherText": "33612AB9D6929C3A828EB6030082B2BD"  
}
```

Para obtener más información, consulte [Cifrar datos](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte la Referencia [EncryptData](#) de AWS CLI comandos.

## generate-card-validation-data

En el siguiente ejemplo de código, se muestra cómo usar `generate-card-validation-data`.

## AWS CLI

Para generar un CVV

El siguiente `generate-card-validation-data` ejemplo genera un CVV/CVV2.

```
aws payment-cryptography-data generate-card-validation-data \  
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/  
  kwapwa6qaiFlw2h \  
  --primary-account-number=171234567890123 \  
  \
```

```
--generation-attributes CardVerificationValue2={CardExpiryDate=0123}
```

Salida:

```
{
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
  kwapwa6qaifl1w2h",
  "KeyCheckValue": "CADD1",
  "ValidationData": "801"
}
```

Para obtener más información, consulte [Generar datos de tarjetas](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [GenerateCardValidationData](#) la Referencia de AWS CLI comandos.

## generate-mac

En el siguiente ejemplo de código, se muestra cómo usar generate-mac.

### AWS CLI

Para generar un MAC

El siguiente generate-card-validation-data ejemplo genera un código de autenticación de mensajes basado en hash (HMAC) para la autenticación de los datos de la tarjeta mediante el algoritmo HMAC\_SHA256 y una clave de HMAC cifrado. La clave debe estar KeyUsage configurada en TR31\_M7\_HMAC\_KEY y KeyModesOfUse en Generate

```
aws payment-cryptography-data generate-mac \
  --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
  kwapwa6qaifl1w2h \
  --message-
  data "3b313038383439303031303733393431353d32343038323236303030373030303f33" \
  --generation-attributes Algorithm=HMAC_SHA256
```

Salida:

```
{
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
  kwapwa6qaifl1w2h,
```



```

    "KeyCheckValue": "2976E7",
    "Mac": "ED87F26E961C6D0DDB78DA5038AA2BDDEA0DCE03E5B5E96BDDD494F4A7AA470C"
  }

```

Para obtener más información, consulte [Generar MAC](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [GenerateMac](#) la Referencia de AWS CLI comandos.

## generate-pin-data

En el siguiente ejemplo de código, se muestra cómo usar generate-pin-data.

### AWS CLI

Para generar un PIN

En el siguiente generate-card-validation-data ejemplo, se genera un nuevo aleatorio PIN mediante el PIN esquema Visa.

```

aws payment-cryptography-data generate-pin-data \
  --generation-key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/37y2tsl45p5zjbh2 \
  --encryption-key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/ivi5ksfsuplneuyt \
  --primary-account-number 171234567890123 \
  --pin-block-format ISO_FORMAT_0 \
  --generation-attributes VisaPin={PinVerificationKeyIndex=1}

```

Salida:

```

{
  "GenerationKeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/37y2tsl45p5zjbh2",
  "GenerationKeyCheckValue": "7F2363",
  "EncryptionKeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/ivi5ksfsuplneuyt",
  "EncryptionKeyCheckValue": "7CC9E2",
  "EncryptedPinBlock": "AC17DC148BDA645E",
  "PinData": {
    "VerificationValue": "5507"
  }
}

```

```
}
}
```

Para obtener más información, consulte [Generar PIN datos](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [GeneratePinData](#) la Referencia de AWS CLI comandos.

## re-encrypt-data

En el siguiente ejemplo de código, se muestra cómo usar `re-encrypt-data`.

### AWS CLI

Para volver a cifrar los datos con una clave diferente

En el siguiente `re-encrypt-data` ejemplo, se descifra el texto cifrado con una clave AES simétrica y se vuelve a cifrar con una clave única derivada por transacción (`DUKPT`).

```
aws payment-cryptography-data re-encrypt-data \
  --incoming-key-identifier arn:aws:payment-cryptography:us-
west-2:111122223333:key/hyvv7ymboitd4vfy \
  --outgoing-key-identifier arn:aws:payment-cryptography:us-
west-2:111122223333:key/jl6ythkcvzesbxen \
  --cipher-
text 4D2B0BDBA192D5AEFEAA5B3EC28E4A65383C313FFA25140101560F75FE1B99F27192A90980AB9334
\
  --incoming-encryption-
attributes "Dukpt={Mode=ECB,KeySerialNumber=012345678911111}" \
  --outgoing-encryption-attributes '{"Symmetric": {"Mode": "ECB"}}'
```

Salida:

```
{
  "CipherText":
  "F94959DA30EEFF0C035483C6067667CF6796E3C1AD28C2B61F9CFEB772A8DD41C0D6822931E0D3B1",
  "KeyArn": "arn:aws:payment-cryptography:us-west-2:111122223333:key/
jl6ythkcvzesbxen",
  "KeyCheckValue": "2E8CD9"
}
```

Para obtener más información, consulte [Cifrar y descifrar datos](#) en la Guía del usuario de criptografía de pagos.AWS

- Para API obtener más información, consulte la Referencia de [ReEncryptData](#)comandos AWS CLI .

## translate-pin-data

En el siguiente ejemplo de código, se muestra cómo usar translate-pin-data.

### AWS CLI

Para traducir PIN datos

El siguiente translate-pin-data ejemplo traduce un PEK TDES cifrado PIN de un PIN bloque ISO 0 a un PIN bloque de AES ISO 4 mediante el DUKPT algoritmo.

```
aws payment-cryptography-data translate-pin-data \
  --encrypted-pin-block "AC17DC148BDA645E" \
  --incoming-translation-
attributes=IsoFormat0='{PrimaryAccountNumber=171234567890123}' \
  --incoming-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/ivi5ksfsuplneuyt \
  --outgoing-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/4pmyquwjs3yj4vwe \
  --outgoing-translation-attributes
IsoFormat4="{PrimaryAccountNumber=171234567890123}" \
  --outgoing-dukpt-attributes KeySerialNumber="FFFF9876543210E00008"
```

Salida:

```
{
  "PinBlock": "1F4209C670E49F83E75CC72E81B787D9",
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt
  "KeyCheckValue": "7CC9E2"
}
```

Para obtener más información, consulte [Traducir PIN datos](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [TranslatePinData](#) la Referencia de AWS CLI comandos.

## verify-auth-request-ciphertext

En el siguiente ejemplo de código, se muestra cómo usar `verify-auth-request-ciphertext`.

### AWS CLI

Para verificar una solicitud de autenticación

El siguiente `verify-auth-request-ciphertext` ejemplo verifica un ciphertext de solicitud de autorización (). ARQC

```
aws payment-cryptography-data verify-auth-request-ciphertext \
  --auth-request-ciphertext F6E1BD1E6037FB3E \
  --auth-response-attributes '{"ArpcMethod1": {"AuthResponseCode": "1111"}}' \
  --key-identifier arn:aws:payment-cryptography:us-west-2:111122223333:key/
pboipdfzd4mdklya \
  --major-key-derivation-mode "EMV_OPTION_A" \
  --session-key-derivation-attributes '{"EmvCommon":
```

```
"ApplicationTransactionCounter": "1234", "PanSequenceNumber":
"01", "PrimaryAccountNumber": "471234567890123"}' \
  --transaction-data "123456789ABCDEF"
```

Salida:

```
{
  "AuthResponseValue": "D899B8C6FBF971AA",
  "KeyArn": "arn:aws:payment-cryptography:us-west-2:111122223333:key/
pboipdfzd4mdklya",
  "KeyCheckValue": "985792"
}
```

Para obtener más información, consulte [Verificar el ciphertext de la solicitud de autenticación \(ARQC\) en la Guía del usuario de criptografía](#) de AWS pagos.

- Para API obtener más información, consulte la Referencia de [VerifyAuthRequestCiphertext](#) comandos AWS CLI .

## verify-card-validation-data

En el siguiente ejemplo de código, se muestra cómo usar `verify-card-validation-data`.

### AWS CLI

Para validar un CVV

El siguiente `verify-card-validation-data` ejemplo valida un CVV/CVV2 para aPAN.

```
aws payment-cryptography-data verify-card-validation-data \
  --key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi \
  --primary-account-number=171234567890123 \
  --verification-attributes CardVerificationValue2={CardExpiryDate=0123} \
  --validation-data 801
```

Salida:

```
{
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi",
  "KeyCheckValue": "CADD1"
}
```

Para obtener más información, consulte [Verificar los datos de la tarjeta](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [VerifyCardValidationData](#) la Referencia de AWS CLI comandos.

## verify-mac

En el siguiente ejemplo de código, se muestra cómo usar `verify-mac`.

### AWS CLI

Para verificar un MAC

El siguiente `verify-mac` ejemplo verifica un código de autenticación de mensajes basado en hash (HMAC) para la autenticación de los datos de la tarjeta mediante el algoritmo HMAC \_ SHA256 y una clave de HMAC cifrado.

```
aws payment-cryptography-data verify-mac \
  --key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
qnobl5lghrzunce6 \
  --message-
data "3b343038383439303031303733393431353d32343038323236303030373030303f33" \
  --verification-attributes='Algorithm=HMAC_SHA256' \
  --mac ED87F26E961C6D0DDB78DA5038AA2BDDEA0DCE03E5B5E96BDDD494F4A7AA470C
```

Salida:

```
{
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
qnobl5lghrzunce6,
  "KeyCheckValue": "2976E7",
}
```

Para obtener más información, consulte [Verificar MAC](#) en la Guía del usuario de criptografía AWS de pagos.

- Para API obtener más información, consulte [VerifyMac](#) la Referencia de AWS CLI comandos.

## verify-pin-data

En el siguiente ejemplo de código, se muestra cómo usar `verify-pin-data`.

### AWS CLI

Para verificar un PIN

El siguiente `verify-pin-data` ejemplo valida a PIN para aPAN.

```
aws payment-cryptography-data verify-pin-data \
  --verification-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/37y2tsl45p5zjbh2 \
  --encryption-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/ivi5ksfsuplneuyt \
  --primary-account-number 171234567890123 \
  --pin-block-format ISO_FORMAT_0 \
  --verification-attributes
VisaPin="{PinVerificationKeyIndex=1,VerificationValue=5507}" \
  --encrypted-pin-block AC17DC148BDA645E
```

**Salida:**

```
{
  "VerificationKeyArn": "arn:aws:payment-cryptography:us-
east-2:111122223333:key/37y2tsl45p5zjbh2",
  "VerificationKeyCheckValue": "7F2363",
  "EncryptionKeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt",
  "EncryptionKeyCheckValue": "7CC9E2",
}
```

Para obtener más información, consulte [Verificar PIN datos](#) en la Guía del usuario AWS de criptografía de pagos.

- Para API obtener más información, consulte [VerifyPinData](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon Pinpoint que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Amazon Pinpoint.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

#### **create-app**

En el siguiente ejemplo de código, se muestra cómo usar `create-app`.

### AWS CLI

#### Ejemplo 1: Creación de una aplicación

En el siguiente ejemplo de `create-app` se crea una nueva aplicación (proyecto).

```
aws pinpoint create-app \  
  --create-application-request Name=ExampleCorp
```

Salida:

```
{  
  "ApplicationResponse": {  
    "Arn": "arn:aws:mobiletargeting:us-  
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",  
    "Id": "810c7aab86d42fb2b56c8c966example",  
    "Name": "ExampleCorp",  
    "tags": {}  
  }  
}
```

Ejemplo 2: Creación de una aplicación etiquetada

En el siguiente ejemplo de `create-app` se crea una nueva aplicación (proyecto) y asocia una etiqueta (clave y valor) a la aplicación.

```
aws pinpoint create-app \  
  --create-application-request Name=ExampleCorp,tags={"Stack"="Test"}
```

Salida:

```
{  
  "ApplicationResponse": {  
    "Arn": "arn:aws:mobiletargeting:us-  
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",  
    "Id": "810c7aab86d42fb2b56c8c966example",  
    "Name": "ExampleCorp",  
    "tags": {  
      "Stack": "Test"  
    }  
  }  
}
```

- Para API obtener más información, consulte [CreateApp](#) la Referencia de AWS CLI comandos.



## create-sms-template

En el siguiente ejemplo de código, se muestra cómo usar `create-sms-template`.

### AWS CLI

Crea una plantilla de mensajes para los mensajes que se envían a través del SMS canal

En el siguiente `create-sms-template` ejemplo, se crea una plantilla de SMS mensaje.

```
aws pinpoint create-sms-template \  
  --template-name TestTemplate \  
  --sms-template-request file://myfile.json \  
  --region us-east-1
```

Contenidos de `myfile.json`:

```
{  
  "Body": "hello\n how are you?\n food is good",  
  "TemplateDescription": "Test SMS Template"  
}
```

Salida:

```
{  
  "CreateTemplateMessageBody": {  
    "Arn": "arn:aws:mobiletargeting:us-east-1:AIDACKCEVSQ6C2EXAMPLE:templates/  
TestTemplate/SMS",  
    "Message": "Created",  
    "RequestID": "8c36b17f-a0b0-400f-ac21-29e9b62a975d"  
  }  
}
```

Para obtener más información, consulte las [plantillas de mensajes de Amazon Pinpoint](#) en la Guía del usuario de Amazon Pinpoint.

- Para API obtener más información, consulte la Referencia [CreateSmsTemplate](#) de AWS CLI comandos.

## delete-app

En el siguiente ejemplo de código, se muestra cómo usar `delete-app`.

## AWS CLI

### Eliminación de una aplicación

En el siguiente ejemplo `delete-app` se elimina una aplicación (proyecto).

```
aws pinpoint delete-app \  
  --application-id 810c7aab86d42fb2b56c8c966example
```

Salida:

```
{  
  "ApplicationResponse": {  
    "Arn": "arn:aws:mobiletargeting:us-  
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",  
    "Id": "810c7aab86d42fb2b56c8c966example",  
    "Name": "ExampleCorp",  
    "tags": {}  
  }  
}
```

- Para API obtener más información, consulte [DeleteApp](#) la Referencia de AWS CLI comandos.

### get-apns-channel

En el siguiente ejemplo de código, se muestra cómo usar `get-apns-channel`.

## AWS CLI

Para recuperar información sobre el estado y la configuración del APNs canal de una aplicación

El siguiente `get-apns-channel` ejemplo recupera información sobre el estado y la configuración del APNs canal de una aplicación.

```
aws pinpoint get-apns-channel \  
  --application-id 9ab1068eb0a6461c86cce7f27ce0efd7 \  
  --region us-east-1
```

Salida:

```
{  
  "APNSChannelResponse": {
```

```

    "ApplicationId": "9ab1068eb0a6461c86cce7f27ce0efd7",
    "CreationDate": "2019-05-09T21:54:45.082Z",
    "DefaultAuthenticationMethod": "CERTIFICATE",
    "Enabled": true,
    "HasCredential": true,
    "HasTokenKey": false,
    "Id": "apns",
    "IsArchived": false,
    "LastModifiedDate": "2019-05-09T22:04:01.067Z",
    "Platform": "APNS",
    "Version": 2
  }
}

```

- Para API obtener más información, consulte [GetApnsChannel](#) en la Referencia de AWS CLI comandos.

## get-app

En el siguiente ejemplo de código, se muestra cómo usar `get-app`.

### AWS CLI

Para recuperar información sobre una aplicación (proyecto)

El siguiente `get-app` ejemplo recupera información sobre una aplicación (proyecto).

```

aws pinpoint get-app \
  --application-id 810c7aab86d42fb2b56c8c966example \
  --region us-east-1

```

Salida:

```

{
  "ApplicationResponse": {
    "Arn": "arn:aws:mobiletargeting:us-east-1:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",
    "Id": "810c7aab86d42fb2b56c8c966example",
    "Name": "ExampleCorp",
    "tags": {
      "Year": "2019",
      "Stack": "Production"
    }
  }
}

```

```

    }
  }
}

```

- Para API obtener más información, consulte [GetAppla Referencia de AWS CLI comandos](#).

## get-apps

En el siguiente ejemplo de código, se muestra cómo usar `get-apps`.

### AWS CLI

Para recuperar información sobre todas sus aplicaciones

En el siguiente `get-apps` ejemplo, se recupera información sobre todas las aplicaciones (proyectos).

```
aws pinpoint get-apps
```

Salida:

```

{
  "ApplicationsResponse": {
    "Item": [
      {
        "Arn": "arn:aws:mobiletargeting:us-west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",
        "Id": "810c7aab86d42fb2b56c8c966example",
        "Name": "ExampleCorp",
        "tags": {
          "Year": "2019",
          "Stack": "Production"
        }
      },
      {
        "Arn": "arn:aws:mobiletargeting:us-west-2:AIDACKCEVSQ6C2EXAMPLE:apps/42d8c7eb0990a57ba1d5476a3example",
        "Id": "42d8c7eb0990a57ba1d5476a3example",
        "Name": "AnyCompany",
        "tags": {}
      }
    ]
  }
}

```

```

        "Arn": "arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/80f5c382b638ffe5ad12376bbexample",
        "Id": "80f5c382b638ffe5ad12376bbexample",
        "Name": "ExampleCorp_Test",
        "tags": {
            "Year": "2019",
            "Stack": "Test"
        }
    ],
    "NextToken":
    "eyJJDcmVhdGlvbkRhdGUiOiIyMDE5LTA3LTE2VDE0jM4OjUzLjkwM1oiLCJBY2NvdW50SWQiOiI1MTIzOTcxODM4NmZlZDQ1IiwiaWF0IjoiMjAxOS0xMi0yNSAwOjowOjAwIn0="
}

```

La presencia del valor de `NextToken` respuesta indica que hay más resultados disponibles. Vuelva a ejecutar el comando e introduzca ese valor como parámetro de `NextToken` entrada.

- Para API obtener más información, consulte [GetApps](#) la Referencia de AWS CLI comandos.

## get-campaign

En el siguiente ejemplo de código, se muestra cómo usar `get-campaign`.

### AWS CLI

Para recuperar información sobre el estado, la configuración y otros ajustes de una campaña

En el siguiente `get-campaign` ejemplo, se recupera información sobre el estado, la configuración y otros ajustes de una campaña.

```

aws pinpoint get-campaign \
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
  --campaign-id a1e63c6cc0eb43ed826ffcc3cc90b30d \
  --region us-east-1

```

Salida:

```

{
  "CampaignResponse": {
    "AdditionalTreatments": [],
    "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",

```

```

    "Arn": "arn:aws:mobiletargeting:us-
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/6e0b7591a90841d2b5d93fa11143e5a7/campaigns/
a1e63c6cc0eb43ed826ffcc3cc90b30d",
    "CreationDate": "2019-10-08T18:40:16.581Z",
    "Description": " ",
    "HoldoutPercent": 0,
    "Id": "a1e63c6cc0eb43ed826ffcc3cc90b30d",
    "IsPaused": false,
    "LastModifiedDate": "2019-10-08T18:40:16.581Z",
    "Limits": {
      "Daily": 0,
      "MaximumDuration": 60,
      "MessagesPerSecond": 50,
      "Total": 0
    },
    "MessageConfiguration": {
      "EmailMessage": {
        "FromAddress": "sender@example.com",
        "HtmlBody": "<!DOCTYPE html>\n <html lang=\"en\">\n <head>\n
<meta http-equiv=\"Content-Type\" content=\"text/html; charset=utf-8\" />\n</head>
\n<body>Hello</body>\n</html>",
        "Title": "PinpointDemo"
      }
    },
    "Name": "MyCampaign",
    "Schedule": {
      "IsLocalTime": false,
      "StartTime": "IMMEDIATE",
      "Timezone": "utc"
    },
    "SegmentId": "b66c9e42f71444b2aa2e0ffc1df28f60",
    "SegmentVersion": 1,
    "State": {
      "CampaignStatus": "COMPLETED"
    },
    "tags": {},
    "TemplateConfiguration": {},
    "Version": 1
  }
}

```

- Para API obtener más información, consulte [GetCampaign](#) la Referencia de AWS CLI comandos.

## get-campaigns

En el siguiente ejemplo de código, se muestra cómo usar `get-campaigns`.

### AWS CLI

Para recuperar información sobre el estado, la configuración y otros ajustes de todas las campañas asociadas a una aplicación

En el siguiente `get-campaigns` ejemplo, se recupera información sobre el estado, la configuración y otros ajustes de todas las campañas asociadas a una aplicación.

```
aws pinpoint get-campaigns \  
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \  
  --region us-east-1
```

Salida:

```
{  
  "CampaignsResponse": {  
    "Item": [  
      {  
        "AdditionalTreatments": [],  
        "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",  
        "Arn": "arn:aws:mobiletargeting:us-  
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/6e0b7591a90841d2b5d93fa11143e5a7/  
campaigns/7e1280344c8f4a9aa40a00b006fe44f1",  
        "CreationDate": "2019-10-08T18:40:22.905Z",  
        "Description": " ",  
        "HoldoutPercent": 0,  
        "Id": "7e1280344c8f4a9aa40a00b006fe44f1",  
        "IsPaused": false,  
        "LastModifiedDate": "2019-10-08T18:40:22.905Z",  
        "Limits": {},  
        "MessageConfiguration": {  
          "EmailMessage": {  
            "FromAddress": "sender@example.com",  
            "HtmlBody": "<!DOCTYPE html>\n    <html lang=\"en  
\n    <head>\n    <meta http-equiv=\"Content-Type\" content=\"text/html;  
charset=utf-8\" />\n</head>\n<body>Hello</body>\n</html>",  
            "Title": "PinpointDemo Test"  
          }  
        }  
      }  
    ],  
  }  
}
```

```

    "Name": "MyCampaign1",
    "Schedule": {
      "IsLocalTime": false,
      "QuietTime": {},
      "StartTime": "IMMEDIATE",
      "Timezone": "UTC"
    },
    "SegmentId": "b66c9e42f71444b2aa2e0ffc1df28f60",
    "SegmentVersion": 1,
    "State": {
      "CampaignStatus": "COMPLETED"
    },
    "tags": {},
    "TemplateConfiguration": {},
    "Version": 1
  },
  {
    "AdditionalTreatments": [],
    "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
    "Arn": "arn:aws:mobiletargeting:us-east-1:AIDACKCEVSQ6C2EXAMPLE:apps/6e0b7591a90841d2b5d93fa11143e5a7/campaigns/a1e63c6cc0eb43ed826ffcc3cc90b30d",
    "CreationDate": "2019-10-08T18:40:16.581Z",
    "Description": " ",
    "HoldoutPercent": 0,
    "Id": "a1e63c6cc0eb43ed826ffcc3cc90b30d",
    "IsPaused": false,
    "LastModifiedDate": "2019-10-08T18:40:16.581Z",
    "Limits": {
      "Daily": 0,
      "MaximumDuration": 60,
      "MessagesPerSecond": 50,
      "Total": 0
    },
    "MessageConfiguration": {
      "EmailMessage": {
        "FromAddress": "sender@example.com",
        "HtmlBody": "<!DOCTYPE html>\n  <html lang=\"en\n\n\n  <head>\n    <meta http-equiv=\"Content-Type\" content=\"text/html; charset=utf-8\" />\n  </head>\n  <body>Demo</body>\n</html>",
        "Title": "PinpointDemo"
      }
    },
    "Name": "MyCampaign2",

```



```

        "Schedule": {
            "IsLocalTime": false,
            "StartTime": "IMMEDIATE",
            "Timezone": "utc"
        },
        "SegmentId": "b66c9e42f71444b2aa2e0ffc1df28f60",
        "SegmentVersion": 1,
        "State": {
            "CampaignStatus": "COMPLETED"
        },
        "tags": {},
        "TemplateConfiguration": {},
        "Version": 1
    }
]
}

```

- Para API obtener más información, consulte [GetCampaigns](#) la Referencia de AWS CLI comandos.

## get-channels

En el siguiente ejemplo de código, se muestra cómo usar `get-channels`.

### AWS CLI

Para recuperar información sobre el historial y el estado de cada canal de una aplicación

El siguiente `get-channels` ejemplo recupera información sobre el historial y el estado de cada canal de una aplicación.

```

aws pinpoint get-channels \
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
  --region us-east-1

```

Salida:

```

{
  "ChannelsResponse": {
    "Channels": {

```

```
"GCM": {
  "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
  "CreationDate": "2019-10-08T18:28:23.182Z",
  "Enabled": true,
  "HasCredential": true,
  "Id": "gcm",
  "IsArchived": false,
  "LastModifiedDate": "2019-10-08T18:28:23.182Z",
  "Version": 1
},
"SMS": {
  "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
  "CreationDate": "2019-10-08T18:39:18.511Z",
  "Enabled": true,
  "Id": "sms",
  "IsArchived": false,
  "LastModifiedDate": "2019-10-08T18:39:18.511Z",
  "Version": 1
},
"EMAIL": {
  "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
  "CreationDate": "2019-10-08T18:27:23.990Z",
  "Enabled": true,
  "Id": "email",
  "IsArchived": false,
  "LastModifiedDate": "2019-10-08T18:27:23.990Z",
  "Version": 1
},
"IN_APP": {
  "Enabled": true,
  "IsArchived": false,
  "Version": 0
}
}
}
```

- Para API obtener más información, consulte [GetChannels](#) la Referencia de AWS CLI comandos.

## get-email-channel

En el siguiente ejemplo de código, se muestra cómo usar `get-email-channel`.

## AWS CLI

Para recuperar información sobre el estado y la configuración del canal de correo electrónico de una aplicación

El siguiente `get-email-channel` ejemplo recupera el estado y la configuración del canal de correo electrónico de una aplicación.

```
aws pinpoint get-email-channel \  
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \  
  --region us-east-1
```

Salida:

```
{  
  "EmailChannelResponse": {  
    "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",  
    "CreationDate": "2019-10-08T18:27:23.990Z",  
    "Enabled": true,  
    "FromAddress": "sender@example.com",  
    "Id": "email",  
    "Identity": "arn:aws:ses:us-east-1:AIDACKCEVSQ6C2EXAMPLE:identity/  
sender@example.com",  
    "IsArchived": false,  
    "LastModifiedDate": "2019-10-08T18:27:23.990Z",  
    "MessagesPerSecond": 1,  
    "Platform": "EMAIL",  
    "RoleArn": "arn:aws:iam::AIDACKCEVSQ6C2EXAMPLE:role/pinpoint-events",  
    "Version": 1  
  }  
}
```

- Para API obtener más información, consulte [GetEmailChannel](#) la Referencia de AWS CLI comandos.

## get-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `get-endpoint`.

## AWS CLI

Recuperación de información sobre la configuración y los atributos de un punto de conexión específico de una aplicación

En el siguiente ejemplo `get-endpoint` se recupera información sobre la configuración y los atributos de un punto de conexión específico de una aplicación.

```
aws pinpoint get-endpoint \  
  --application-id 611e3e3cdd47474c9c1399a505665b91 \  
  --endpoint-id testendpoint \  
  --region us-east-1
```

Salida:

```
{  
  "EndpointResponse": {  
    "Address": "+11234567890",  
    "ApplicationId": "611e3e3cdd47474c9c1399a505665b91",  
    "Attributes": {},  
    "ChannelType": "SMS",  
    "CohortId": "63",  
    "CreationDate": "2019-01-28T23:55:11.534Z",  
    "EffectiveDate": "2021-08-06T00:04:51.763Z",  
    "EndpointStatus": "ACTIVE",  
    "Id": "testendpoint",  
    "Location": {  
      "Country": "USA"  
    },  
    "Metrics": {  
      "SmsDelivered": 1.0  
    },  
    "OptOut": "ALL",  
    "RequestId": "a204b1f2-7e26-48a7-9c80-b49a2143489d",  
    "User": {  
      "UserAttributes": {  
        "Age": [  
          "24"  
        ]  
      },  
      "UserId": "testuser"  
    }  
  }  
}
```

```
}
```

- Para API obtener más información, consulte [GetEndpoint](#) la Referencia de AWS CLI comandos.

## get-gcm-channel

En el siguiente ejemplo de código, se muestra cómo usar `get-gcm-channel`.

### AWS CLI

Para recuperar información sobre el estado y la configuración del GCM canal de una aplicación

El siguiente `get-gcm-channel` ejemplo recupera información sobre el estado y la configuración del GCM canal de una aplicación.

```
aws pinpoint get-gcm-channel \  
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \  
  --region us-east-1
```

Salida:

```
{  
  "GCMChannelResponse": {  
    "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",  
    "CreationDate": "2019-10-08T18:28:23.182Z",  
    "Enabled": true,  
    "HasCredential": true,  
    "Id": "gcm",  
    "IsArchived": false,  
    "LastModifiedDate": "2019-10-08T18:28:23.182Z",  
    "Platform": "GCM",  
    "Version": 1  
  }  
}
```

- Para API obtener más información, consulte [GetGcmChannel](#) la Referencia de AWS CLI comandos.

## get-sms-channel

En el siguiente ejemplo de código, se muestra cómo usar `get-sms-channel`.

## AWS CLI

Para recuperar información sobre el estado y la configuración del SMS canal de una aplicación

En el siguiente ejemplo de `get-sms-channel` se recupera el estado y la configuración del canal de SMS de una aplicación.

```
aws pinpoint get-sms-channel \  
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \  
  --region us-east-1
```

Salida:

```
{  
  "SMSChannelResponse": {  
    "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",  
    "CreationDate": "2019-10-08T18:39:18.511Z",  
    "Enabled": true,  
    "Id": "sms",  
    "IsArchived": false,  
    "LastModifiedDate": "2019-10-08T18:39:18.511Z",  
    "Platform": "SMS",  
    "PromotionalMessagesPerSecond": 20,  
    "TransactionalMessagesPerSecond": 20,  
    "Version": 1  
  }  
}
```

- Para API obtener más información, consulte [GetSmsChannel](#) la Referencia de AWS CLI comandos.

## get-sms-template

En el siguiente ejemplo de código, se muestra cómo usar `get-sms-template`.

## AWS CLI

Recupera el contenido y la configuración de una plantilla de mensajes para los mensajes que se envían a través del canal SMS

En el siguiente `get-sms-template` ejemplo, se recupera el contenido y la configuración de una plantilla de SMS mensaje.

```
aws pinpoint get-sms-template \  
  --template-name TestTemplate \  
  --region us-east-1
```

Salida:

```
{  
  "SMSTemplateResponse": {  
    "Arn": "arn:aws:mobiletargeting:us-east-1:AIDACKCEVSQ6C2EXAMPLE:templates/  
TestTemplate/SMS",  
    "Body": "hello\n how are you?\n food is good",  
    "CreationDate": "2023-06-20T21:37:30.124Z",  
    "LastModifiedDate": "2023-06-20T21:37:30.124Z",  
    "tags": {},  
    "TemplateDescription": "Test SMS Template",  
    "TemplateName": "TestTemplate",  
    "TemplateType": "SMS",  
    "Version": "1"  
  }  
}
```

Para obtener más información, consulte las [plantillas de mensajes de Amazon Pinpoint](#) en la Guía del usuario de Amazon Pinpoint.

- Para API obtener más información, consulte la Referencia [GetSmsTemplate](#) de AWS CLI comandos.

## get-voice-channel

En el siguiente ejemplo de código, se muestra cómo usar `get-voice-channel`.

AWS CLI

Para recuperar información sobre el estado y la configuración del canal de voz de una aplicación

El siguiente `get-voice-channel` ejemplo recupera el estado y la configuración del canal de voz de una aplicación.

```
aws pinpoint get-voice-channel \  
  --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \  
  --region us-east-1
```

Salida:

```
{
  "VoiceChannelResponse": {
    "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
    "CreationDate": "2022-04-28T00:17:03.836Z",
    "Enabled": true,
    "Id": "voice",
    "IsArchived": false,
    "LastModifiedDate": "2022-04-28T00:17:03.836Z",
    "Platform": "VOICE",
    "Version": 1
  }
}
```

- Para API obtener más información, consulte [GetVoiceChannel](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

AWS CLI

Para recuperar una lista de etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se recuperan todas las etiquetas (nombres y valores clave) que están asociadas al recurso especificado.

```
aws pinpoint list-tags-for-resource \
  --resource-arn arn:aws:mobiletargeting:us-west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example
```

Salida:

```
{
  "TagsModel": {
    "tags": {
      "Year": "2019",
      "Stack": "Production"
    }
  }
}
```



```
}  
}
```

Para obtener más información, consulte «Etiquetado de los recursos de Amazon Pinpoint <https://docs.aws.amazon.com/pinpoint/latest/developerguide/tagging-resources.html>» en la Guía para desarrolladores de Amazon Pinpoint.

- Para AWS CLI obtener API más información, [ListTagsForResource](#) consulte la Referencia de comandos.

## phone-number-validate

En el siguiente ejemplo de código, se muestra cómo usar `phone-number-validate`.

### AWS CLI

Recupera información sobre un número de teléfono

A continuación, se `phone-number-validate` recupera información sobre un número de teléfono.

```
aws pinpoint phone-number-validate \  
  --number-validate-request PhoneNumber="+12065550142" \  
  --region us-east-1
```

Salida:

```
{  
  "NumberValidateResponse": {  
    "Carrier": "ExampleCorp Mobile",  
    "City": "Seattle",  
    "CleansedPhoneNumberE164": "+12065550142",  
    "CleansedPhoneNumberNational": "2065550142",  
    "Country": "United States",  
    "CountryCodeIso2": "US",  
    "CountryCodeNumeric": "1",  
    "OriginalPhoneNumber": "+12065550142",  
    "PhoneType": "MOBILE",  
    "PhoneTypeCode": 0,  
    "Timezone": "America/Los_Angeles",  
    "ZipCode": "98101"  
  }  
}
```

```
}
```

Para obtener más información, consulte el [SMS canal Amazon Pinpoint](#) en la Guía del usuario de Amazon Pinpoint.

- Para API obtener más información, consulte la Referencia [PhoneNumberValidate](#) de AWS CLI comandos.

## send-messages

En el siguiente ejemplo de código, se muestra cómo usar send-messages.

### AWS CLI

Para enviar un SMS mensaje mediante el punto final de una aplicación

En el siguiente ejemplo de send-messages se envía un mensaje directo a una aplicación con un punto de conexión.

```
aws pinpoint send-messages \  
  --application-id 611e3e3cdd47474c9c1399a505665b91 \  
  --message-request file://myfile.json \  
  --region us-west-2
```

Contenidos de myfile.json:

```
{  
  "MessageConfiguration": {  
    "SMSMessage": {  
      "Body": "hello, how are you?"  
    }  
  },  
  "Endpoints": {  
    "testendpoint": {}  
  }  
}
```

Salida:

```
{  
  "MessageResponse": {
```

```

    "ApplicationId": "611e3e3cdd47474c9c1399a505665b91",
    "EndpointResult": {
      "testendpoint": {
        "Address": "+12345678900",
        "DeliveryStatus": "SUCCESSFUL",
        "MessageId": "itnuqhai5alf1n6ahv3udc05n7hhddr6gb3lq6g0",
        "StatusCode": 200,
        "StatusMessage": "MessageId:
itnuqhai5alf1n6ahv3udc05n7hhddr6gb3lq6g0"
      }
    },
    "RequestId": "c7e23264-04b2-4a46-b800-d24923f74753"
  }
}

```

Para obtener más información, consulte el [SMS canal Amazon Pinpoint](#) en la Guía del usuario de Amazon Pinpoint.

- Para API obtener más información, consulte la Referencia [SendMessages](#) de AWS CLI comandos.

## send-users-messages

En el siguiente ejemplo de código, se muestra cómo usar send-users-messages.

### AWS CLI

Para enviar un SMS mensaje a un usuario de una aplicación

El siguiente send-users-messages ejemplo envía un mensaje directo a un usuario de una aplicación.

```

aws pinpoint send-users-messages \
  --application-id 611e3e3cdd47474c9c1399a505665b91 \
  --send-users-message-request file://myfile.json \
  --region us-west-2

```

Contenidos de myfile.json:

```

{
  "MessageConfiguration": {
    "SMSMessage": {

```

```

        "Body": "hello, how are you?"
      }
    },
    "Users": {
      "testuser": {}
    }
  }
}

```

Salida:

```

{
  "SendUsersMessageResponse": {
    "ApplicationId": "611e3e3cdd47474c9c1399a505665b91",
    "RequestId": "e0b12cf5-2359-11e9-bb0b-d5fb91876b25",
    "Result": {
      "testuser": {
        "testuserendpoint": {
          "DeliveryStatus": "SUCCESSFUL",
          "MessageId": "7qu4hk5bqhda3i7i2n4pjf98qcu8b7p45ifsmo0",
          "StatusCode": 200,
          "StatusMessage": "MessageId:
7qu4hk5bqhda3i7i2n4pjf98qcu8b7p45ifsmo0",
          "Address": "+12345678900"
        }
      }
    }
  }
}

```

Para obtener más información, consulte el [SMS canal Amazon Pinpoint](#) en la Guía del usuario de Amazon Pinpoint.

- Para API obtener más información, consulte la Referencia [SendUsersMessages](#) de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir etiquetas a un recurso

En el siguiente ejemplo, se agregan dos etiquetas (nombres y valores clave) a un recurso.

```
aws pinpoint list-tags-for-resource \  
  --resource-arn arn:aws:mobiletargeting:us-  
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example \  
  --tags-model tags={Stack=Production,Year=2019}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte «Etiquetado de los recursos de Amazon Pinpoint <https://docs.aws.amazon.com/pinpoint/latest/developerguide/tagging-resources.html>» en la Guía para desarrolladores de Amazon Pinpoint.

- Para AWS CLI obtener API más información, [TagResource](#) consulte la Referencia de comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar untag-resource.

### AWS CLI

Ejemplo 1: Para eliminar una etiqueta de un recurso

En el siguiente untag-resource ejemplo, se elimina la etiqueta especificada (nombre y valor de la clave) de un recurso.

```
aws pinpoint untag-resource \  
  --resource-arn arn:aws:mobiletargeting:us-  
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example \  
  --tag-keys Year
```

Este comando no genera ninguna salida.

Ejemplo 2: Para eliminar varias etiquetas de un recurso

En el siguiente untag-resource ejemplo, se eliminan las etiquetas especificadas (nombres y valores clave) de un recurso.

```
aws pinpoint untag-resource \  
  --resource-arn arn:aws:mobiletargeting:us-  
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example \  
  --tag-keys Year
```

```
--tag-keys Year Stack
```

Este comando no genera ninguna salida.

Para obtener más información, consulte «Etiquetado de los recursos de Amazon Pinpoint <https://docs.aws.amazon.com/pinpoint/latest/developerguide/tagging-resources.html>» en la Guía para desarrolladores de Amazon Pinpoint.

- Para AWS CLI obtener API más información, [UntagResource](#) consulte la Referencia de comandos.

## update-sms-channel

En el siguiente ejemplo de código, se muestra cómo usar `update-sms-channel`.

### AWS CLI

Para habilitar el SMS canal o actualizar el estado y la configuración del SMS canal para una aplicación.

El siguiente `update-sms-channel` ejemplo habilita el SMS canal para un SMS canal de una aplicación.

```
aws pinpoint update-sms-channel \  
  --application-id 611e3e3cdd47474c9c1399a505665b91 \  
  --sms-channel-request Enabled=true \  
  --region us-west-2
```

Salida:

```
{  
  "SMSChannelResponse": {  
    "ApplicationId": "611e3e3cdd47474c9c1399a505665b91",  
    "CreationDate": "2019-01-28T23:25:25.224Z",  
    "Enabled": true,  
    "Id": "sms",  
    "IsArchived": false,  
    "LastModifiedDate": "2023-05-18T23:22:50.977Z",  
    "Platform": "SMS",  
    "PromotionalMessagesPerSecond": 20,  
    "TransactionalMessagesPerSecond": 20,  
    "Version": 3
```

```
}  
}
```

Para obtener más información, consulte el [SMS canal Amazon Pinpoint](#) en la Guía del usuario de Amazon Pinpoint.

- Para API obtener más información, consulte la Referencia [UpdateSmsChannel](#) de AWS CLI comandos.

## Ejemplos de Amazon Polly que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Amazon Polly.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **delete-lexicon**

En el siguiente ejemplo de código, se muestra cómo usar `delete-lexicon`.

AWS CLI

Eliminación de un lexicón

En el siguiente ejemplo de `delete-lexicon` se elimina el lexicón especificado.

```
aws polly delete-lexicon \  
  --name w3c
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de la DeleteLexicon operación](#) en la Guía para desarrolladores de Amazon Polly.

- Para API obtener más información, consulte [DeleteLexicon](#) la Referencia de AWS CLI comandos.

## get-lexicon

En el siguiente ejemplo de código, se muestra cómo usar `get-lexicon`.

### AWS CLI

Recuperación del contenido de un léxico

En el siguiente ejemplo de `get-lexicon` se recupera el contenido del lexicón de pronunciación especificado.

```
aws polly get-lexicon \  
  --name w3c
```

Salida:

```
{  
  "Lexicon": {  
    "Content": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<lexicon version=  
\"1.0\" \n      xmlns=      \"http://www.w3.org/2005/01/pronunciation-lexicon  
\" \n      xmlns:xsi= \"http://www.w3.org/2001/XMLSchema-instance\" \n      xsi:schemaLocation= \"http://www.w3.org/2005/01/pronunciation-lexicon \n      http://www.w3.org/TR/2007/CR-pronunciation-lexicon-20071212/pls.xsd\" \n      alphabet= \"ipa\" \n      xml:lang= \"en-US\">\n  <lexeme>\n    <grapheme>W3C</  
grapheme>\n    <alias>World Wide Web Consortium</alias>\n  </lexeme>\n</lexicon>  
 \n",  
    "Name": "w3c"  
  },  
  "LexiconAttributes": {  
    "Alphabet": "ipa",  
    "LanguageCode": "en-US",  
    "LastModified": 1603908910.99,  
    "LexiconArn": "arn:aws:polly:us-west-2:880185128111:lexicon/w3c",  
    "LexemesCount": 1,  
    "Size": 492  
  }  
}
```



```
}
```

Para obtener más información, consulte [Uso de la GetLexicon operación](#) en la Guía para desarrolladores de Amazon Polly.

- Para API obtener más información, consulte [GetLexicon](#) la Referencia de AWS CLI comandos.

## get-speech-synthesis-task

En el siguiente ejemplo de código, se muestra cómo usar `get-speech-synthesis-task`.

### AWS CLI

Obtención de información sobre una tarea de síntesis de voz

En el siguiente ejemplo de `get-speech-synthesis-task` se recupera información sobre la tarea de síntesis de voz especificada.

```
aws polly get-speech-synthesis-task \  
  --task-id 70b61c0f-57ce-4715-a247-cae8729dcce9
```

Salida:

```
{  
  "SynthesisTask": {  
    "TaskId": "70b61c0f-57ce-4715-a247-cae8729dcce9",  
    "TaskStatus": "completed",  
    "OutputUri": "https://s3.us-west-2.amazonaws.com/my-s3-  
bucket/70b61c0f-57ce-4715-a247-cae8729dcce9.mp3",  
    "CreationTime": 1603911042.689,  
    "RequestCharacters": 1311,  
    "OutputFormat": "mp3",  
    "TextType": "text",  
    "VoiceId": "Joanna"  
  }  
}
```

Para obtener más información, consulte [Creación de archivos de audio largos](#) en la Guía para desarrolladores de Amazon Polly.

- Para API obtener más información, consulte [GetSpeechSynthesisTask](#) la Referencia de AWS CLI comandos.

## list-lexicons

En el siguiente ejemplo de código, se muestra cómo usar `list-lexicons`.

### AWS CLI

Creación de una lista de léxicos

En el siguiente ejemplo de `list-lexicons` se enumeran sus léxicos de pronunciación.

```
aws polly list-lexicons
```

Salida:

```
{
  "Lexicons": [
    {
      "Name": "w3c",
      "Attributes": {
        "Alphabet": "ipa",
        "LanguageCode": "en-US",
        "LastModified": 1603908910.99,
        "LexiconArn": "arn:aws:polly:us-east-2:123456789012:lexicon/w3c",
        "LexemesCount": 1,
        "Size": 492
      }
    }
  ]
}
```

Para obtener más información, consulte [Uso de la ListLexicons operación](#) en la Guía para desarrolladores de Amazon Polly.

- Para API obtener más información, consulte [ListLexicons](#) la Referencia de AWS CLI comandos.

## list-speech-synthesis-tasks

En el siguiente ejemplo de código, se muestra cómo usar `list-speech-synthesis-tasks`.

### AWS CLI

Para enumerar sus tareas de síntesis de voz

En el `list-speech-synthesis-tasks` ejemplo siguiente se enumeran las tareas de síntesis de voz.

```
aws polly list-speech-synthesis-tasks
```

Salida:

```
{
  "SynthesisTasks": [
    {
      "TaskId": "70b61c0f-57ce-4715-a247-cae8729dcce9",
      "TaskStatus": "completed",
      "OutputUri": "https://s3.us-west-2.amazonaws.com/my-s3-
bucket/70b61c0f-57ce-4715-a247-cae8729dcce9.mp3",
      "CreationTime": 1603911042.689,
      "RequestCharacters": 1311,
      "OutputFormat": "mp3",
      "TextType": "text",
      "VoiceId": "Joanna"
    }
  ]
}
```

Para obtener más información, consulte [Creación de archivos de audio largos](#) en la Guía para desarrolladores de Amazon Polly.

- Para API obtener más información, consulte [ListSpeechSynthesisTasks](#) la Referencia de AWS CLI comandos.

## put-lexicon

En el siguiente ejemplo de código, se muestra cómo usar `put-lexicon`.

AWS CLI

Almacenamiento de un léxico

En el siguiente ejemplo de `put-lexicon` se almacena el léxico de pronunciación especificado. El `example.pls` archivo especifica un léxico compatible con el W3CPLS.

```
aws polly put-lexicon \
```

```
--name w3c \  
--content file://example.pls
```

## Contenido de `example.pls`

```
{  
  <?xml version="1.0" encoding="UTF-8"?>  
  <lexicon version="1.0"  
    xmlns="http://www.w3.org/2005/01/pronunciation-lexicon"  
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
    xsi:schemaLocation="http://www.w3.org/2005/01/pronunciation-lexicon  
      http://www.w3.org/TR/2007/CR-pronunciation-lexicon-20071212/pls.xsd"  
    alphabet="ipa"  
    xml:lang="en-US">  
    <lexeme>  
      <grapheme>W3C</grapheme>  
      <alias>World Wide Web Consortium</alias>  
    </lexeme>  
  </lexicon>  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de la PutLexicon operación](#) en la Guía para desarrolladores de Amazon Polly.

- Para API obtener más información, consulte [PutLexicon](#) la Referencia de AWS CLI comandos.

## start-speech-synthesis-task

En el siguiente ejemplo de código, se muestra cómo usar `start-speech-synthesis-task`.

### AWS CLI

#### Sintetización de texto

El siguiente `start-speech-synthesis-task` ejemplo sintetiza el texto `text_file.txt` y almacena el MP3 archivo resultante en el depósito especificado.

```
aws polly start-speech-synthesis-task \  
--output-format mp3 \  

```

```
--output-s3-bucket-name my-s3-bucket \  
--text file://text_file.txt \  
--voice-id Joanna
```

Salida:

```
{  
  "SynthesisTask": {  
    "TaskId": "70b61c0f-57ce-4715-a247-cae8729dcce9",  
    "TaskStatus": "scheduled",  
    "OutputUri": "https://s3.us-east-2.amazonaws.com/my-s3-  
bucket/70b61c0f-57ce-4715-a247-cae8729dcce9.mp3",  
    "CreationTime": 1603911042.689,  
    "RequestCharacters": 1311,  
    "OutputFormat": "mp3",  
    "TextType": "text",  
    "VoiceId": "Joanna"  
  }  
}
```

Para obtener más información, consulte [Creación de archivos de audio largos](#) en la Guía para desarrolladores de Amazon Polly.

- Para API obtener más información, consulte [StartSpeechSynthesisTask](#) la Referencia de AWS CLI comandos.

## Lista de precios de AWS ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with Lista de precios de AWS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **describe-services**

En el siguiente ejemplo de código, se muestra cómo usar `describe-services`.

#### AWS CLI

Para recuperar los metadatos del servicio

En este ejemplo, se recuperan los metadatos del código de EC2 servicio de Amazon.

Comando:

```
aws pricing describe-services --service-code AmazonEC2 --format-version aws_v1 --  
max-items 1
```

Salida:

```
{  
  "Services": [  
    {  
      "ServiceCode": "AmazonEC2",  
      "AttributeNames": [  
        "volumeType",  
        "maxIopsvolume",  
        "instance",  
        "instanceCapacity10xlarge",  
        "locationType",  
        "instanceFamily",  
        "operatingSystem",  
        "clockSpeed",  
        "LeaseContractLength",  
        "ecu",  
        "networkPerformance",  
        "instanceCapacity8xlarge",  
        "group",  
        "maxThroughputvolume",  
        "gpuMemory",  
        "ebsOptimized",  
        "elasticGpuType",  
        "maxVolumeSize",  
        "gpu",  
        "processorFeatures",
```

```
"intelAvxAvailable",
"instanceCapacity4xlarge",
"servicecode",
"groupDescription",
"processorArchitecture",
"physicalCores",
"productFamily",
"enhancedNetworkingSupported",
"intelTurboAvailable",
"memory",
"dedicatedEbsThroughput",
"vcpu",
"OfferingClass",
"instanceCapacityLarge",
"capacitystatus",
"termType",
"storage",
"intelAvx2Available",
"storageMedia",
"physicalProcessor",
"provisioned",
"servicename",
"PurchaseOption",
"instanceCapacity18xlarge",
"instanceType",
"tenancy",
"usagetype",
"normalizationSizeFactor",
"instanceCapacity2xlarge",
"instanceCapacity16xlarge",
"maxIopsBurstPerformance",
"instanceCapacity12xlarge",
"instanceCapacity32xlarge",
"instanceCapacityXlarge",
"licenseModel",
"currentGeneration",
"preInstalledSw",
"location",
"instanceCapacity24xlarge",
"instanceCapacity9xlarge",
"instanceCapacityMedium",
"operation"
]
}
```

```
  ],  
  "FormatVersion": "aws_v1"  
}
```

- Para API obtener más información, consulte [DescribeServices](#) la Referencia de AWS CLI comandos.

## get-attribute-values

En el siguiente ejemplo de código, se muestra cómo usar `get-attribute-values`.

### AWS CLI

Para recuperar una lista de valores de atributos

El siguiente `get-attribute-values` ejemplo recupera una lista de valores disponibles para el atributo dado.

```
aws pricing get-attribute-values \  
  --service-code AmazonEC2 \  
  --attribute-name volumeType \  
  --max-items 2
```

Salida:

```
{  
  "NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ==",  
  "AttributeValues": [  
    {  
      "Value": "Cold HDD"  
    },  
    {  
      "Value": "General Purpose"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [GetAttributeValues](#) la Referencia de AWS CLI comandos.



## get-products

En el siguiente ejemplo de código, se muestra cómo usar `get-products`.

### AWS CLI

Para recuperar una lista de productos

En este ejemplo, se recupera una lista de productos que cumplen los criterios indicados.

Comando:

```
aws pricing get-products --filters file://filters.json --format-version aws_v1 --
max-results 1 --service-code AmazonEC2
```

filters.json:

```
[
  {
    "Type": "TERM_MATCH",
    "Field": "ServiceCode",
    "Value": "AmazonEC2"
  },
  {
    "Type": "TERM_MATCH",
    "Field": "volumeType",
    "Value": "Provisioned IOPS"
  }
]
```

Salida:

```
{
  "FormatVersion": "aws_v1",
  "NextToken": "WGDY7ko8fQXdlaUZVdasFQ==:RVSagyIFn770XQ0zdUIc09BY6ucBG9itXAZGZF/
zioUz0sUKh6PCcPwa0yPZRiMePb986TeoKYB9155fw/
CyoMq5ymnGmT1Vj39T1jbbAlhcqnVfTmPIilx8Uy5bdDaBYy/e/20fw9Edzsykbs8LTBUbNbiDQ
+BBds5yeI9AQkUepuKk3aEahFPxJ55kx/zk",
  "PriceList": [
    {"product":{"productFamily":"Storage","attributes":{"storageMedia":"
SSD-backed","maxThroughputVolume":"320 MB/sec","volumeType":"Provisioned
IOPS","maxIopsVolume":"20000","serviceCode":"AmazonEC2","usageType
":"APS1-EBS:VolumeUsage.piops","locationType":"AWS Region","location":
```

```

  \ "Asia Pacific (Singapore)\", \ "servicename\": \ "Amazon Elastic Compute Cloud\",
  \ "maxVolumeSize\": \ "16 TiB\", \ "operation\": \ "\", \ "sku\": \ "3MKHN58N7RDDVGKJ\",
  \ "serviceCode\": \ "AmazonEC2\", \ "terms\": { \ "OnDemand\": { \ "3MKHN58N7RDDVGKJ.JRTCKXETXF
  \": { \ "priceDimensions\": { \ "3MKHN58N7RDDVGKJ.JRTCKXETXF.6YS6EN2CT7\": { \ "unit\": \ "GB-
  Mo\", \ "endRange\": \ "Inf\", \ "description\": \ "$0.138 per GB-month of Provisioned
  IOPS SSD (io1) provisioned storage - Asia Pacific (Singapore)\", \ "appliesTo
  \": [], \ "rateCode\": \ "3MKHN58N7RDDVGKJ.JRTCKXETXF.6YS6EN2CT7\", \ "beginRange\":
  \ "0\", \ "pricePerUnit\": { \ "USD\": \ "0.1380000000\"}}, \ "sku\": \ "3MKHN58N7RDDVGKJ
  \", \ "effectiveDate\": \ "2018-08-01T00:00:00Z\", \ "offerTermCode\": \ "JRTCKXETXF
  \", \ "termAttributes\": {}}}}, \ "version\": \ "20180808005701\", \ "publicationDate\":
  \ "2018-08-08T00:57:01Z\"}
  ]
}

```

- Para API obtener más información, consulte la Referencia de [GetProducts](#) comandos AWS CLI .

## AWS Private CA ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Private CA.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **create-certificate-authority-audit-report**

En el siguiente ejemplo de código, se muestra cómo usar `create-certificate-authority-audit-report`.

### AWS CLI

Para crear un informe de auditoría de una entidad emisora de certificados

El siguiente `create-certificate-authority-audit-report` comando crea un informe de auditoría para la CA privada identificada por ARN.

```
aws acm-pca create-certificate-authority-audit-report --certificate-  
authority-arn arn:aws:acm-pca:us-east-1:accountid:certificate-  
authority/12345678-1234-1234-1234-123456789012 --s3-bucket-name your-bucket-name --  
audit-report-response-format JSON
```

- Para API obtener más información, consulte [CreateCertificateAuthorityAuditReport](#) la Referencia de AWS CLI comandos.

## `create-certificate-authority`

En el siguiente ejemplo de código, se muestra cómo usar `create-certificate-authority`.

### AWS CLI

Para crear una entidad emisora de certificados privada

El siguiente `create-certificate-authority` comando crea una autoridad de certificación privada en su AWS cuenta.

```
aws acm-pca create-certificate-authority --certificate-authority-configuration  
file://C:\ca_config.txt --revocation-configuration file://C:\revoke_config.txt --  
certificate-authority-type "SUBORDINATE" --idempotency-token 98256344
```

- Para API obtener más información, consulte [CreateCertificateAuthority](#) la Referencia de AWS CLI comandos.

## `delete-certificate-authority`

En el siguiente ejemplo de código, se muestra cómo usar `delete-certificate-authority`.

### AWS CLI

Para eliminar una entidad emisora de certificados privada

El siguiente `delete-certificate-authority` comando elimina la entidad emisora de certificados identificada por ARN

```
aws acm-pca delete-certificate-authority --certificate-  
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-  
authority/12345678-1234-1234-1234-123456789012
```

- Para API obtener más información, consulte [DeleteCertificateAuthority](#) la Referencia de AWS CLI comandos.

## **describe-certificate-authority-audit-report**

En el siguiente ejemplo de código, se muestra cómo usar `describe-certificate-authority-audit-report`.

### AWS CLI

Para describir un informe de auditoría para una autoridad de certificación

El siguiente `describe-certificate-authority-audit-report` comando muestra información sobre el informe de auditoría especificado para la CA identificada por elARN.

```
aws acm-pca describe-certificate-authority-audit-report --certificate-  
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-  
authority/99999999-8888-7777-6666-555555555555 --audit-report-  
id 11111111-2222-3333-4444-555555555555
```

- Para API obtener más información, consulte [DescribeCertificateAuthorityAuditReport](#) la Referencia de AWS CLI comandos.

## **describe-certificate-authority**

En el siguiente ejemplo de código, se muestra cómo usar `describe-certificate-authority`.

### AWS CLI

Para describir una entidad emisora de certificados privada

El siguiente `describe-certificate-authority` comando muestra información sobre la CA privada identificada porARN.

```
aws acm-pca describe-certificate-authority --certificate-  
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-  
authority/12345678-1234-1234-1234-123456789012
```

- Para API obtener más información, consulte [DescribeCertificateAuthority](#) la Referencia de AWS CLI comandos.

## get-certificate-authority-certificate

En el siguiente ejemplo de código, se muestra cómo usar `get-certificate-authority-certificate`.

### AWS CLI

Para recuperar un certificado de una entidad emisora de certificados (CA)

El siguiente `get-certificate-authority-certificate` comando recupera el certificado y la cadena de certificados de la CA privada especificada por. ARN

```
aws acm-pca get-certificate-authority-certificate --certificate-  
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-  
authority/12345678-1234-1234-1234-123456789012 --output text
```

- Para API obtener más información, consulte [GetCertificateAuthorityCertificate](#) la Referencia de AWS CLI comandos.

## get-certificate-authority-csr

En el siguiente ejemplo de código, se muestra cómo usar `get-certificate-authority-csr`.

### AWS CLI

Para recuperar la solicitud de firma del certificado de una entidad emisora de certificados

El siguiente `get-certificate-authority-csr` comando recupera el CSR de la CA privada especificada por. ARN

```
aws acm-pca get-certificate-authority-csr --certificate-  
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-  
authority/12345678-1234-1234-1234-123456789012 --output text
```

- Para API obtener más información, consulte [GetCertificateAuthorityCsr](#) la Referencia de AWS CLI comandos.

## get-certificate

En el siguiente ejemplo de código, se muestra cómo usar `get-certificate`.

### AWS CLI

Para recuperar un certificado emitido

El siguiente `get-certificate` ejemplo recupera un certificado de la CA privada especificada.

```
aws acm-pca get-certificate \
  --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
  authority/12345678-1234-1234-1234-123456789012 \
  --certificate-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
  authority/12345678-1234-1234-1234-123456789012/
  certificate/6707447683a9b7f4055627ffd55cebcc \
  --output text
```

Salida:

```
-----BEGIN CERTIFICATE-----
MIIEDzCCAvegAwIBAgIRAJuJ8f6ZVYL7gG/rS3qvrZMwDQYJKoZIhvcNAQELBQAw
cTElMAkGA1UEBhMCVVMxEzARBgNVBAGMC1dhc2hpbmd0b24xEDA0BgNVBACMB1Nl
...certificate body truncated for brevity...
tKCSglgZZrd4FdLw1EkGm+UVXnodwMtJEQyy3oTfZjURPIyyaqsKtu/KSS7YDjK0
KQny73D6LtmD0EbaYq10XiDxqY41lvKHJ1eZrPaBmYNABxU=
-----END CERTIFICATE----- -----BEGIN CERTIFICATE-----
MIIDrzCCApegAwIBAgIRA0skdzLvcj1eShkoyEE693AwDQYJKoZIhvcNAQELBQAw
cTElMAkGA1UEBhMCVVMxEzARBgNVBAGMC1dhc2hpbmd0b24xEDA0BgNVBACMB1Nl
...certificate body truncated for brevity...
kdRGB6P2hpxstDOUIwAoCbhoaWwfA4ybJznf+j0QhAziN1RdKQRR8nODWpKt7H9w
dJ5nxsTk/fniJz86Ddtp6n8s82wYdkN3cVffeK72A9aTCOU=
-----END CERTIFICATE-----
```

La primera parte del resultado es el propio certificado. La segunda parte es la cadena de certificados que se enlaza con el certificado raíz de la CA. Tenga en cuenta que al utilizar la `--output text` opción, se inserta un TAB carácter entre las dos partes del certificado (esa es la

causa del texto sangreado). Si pretende utilizar este resultado y analizar los certificados con otras herramientas, puede que tenga que eliminar el TAB carácter para que se procese correctamente.

- Para API obtener más información, consulte [GetCertificate](#) la Referencia de AWS CLI comandos.

## **import-certificate-authority-certificate**

En el siguiente ejemplo de código, se muestra cómo usar `import-certificate-authority-certificate`.

### AWS CLI

Para importar el certificado de la entidad emisora de certificados a ACM PCA

El siguiente `import-certificate-authority-certificate` comando importa el certificado de CA privado firmado para la CA especificada ARN en into ACMPCA.

```
aws acm-pca import-certificate-authority-certificate --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-authority/12345678-1234-1234-1234-123456789012 --certificate file://C:\ca_cert.pem --certificate-chain file://C:\ca_cert_chain.pem
```

- Para API obtener más información, consulte [ImportCertificateAuthorityCertificate](#) la Referencia de AWS CLI comandos.

## **issue-certificate**

En el siguiente ejemplo de código, se muestra cómo usar `issue-certificate`.

### AWS CLI

Para emitir un certificado privado

El siguiente `issue-certificate` comando utiliza la CA privada especificada por el ARN para emitir un certificado privado.

```
aws acm-pca issue-certificate --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-authority/12345678-1234-1234-1234-123456789012 --csr file://C:\cert_1.csr --signing-algorithm "SHA256WITHRSA" --validity Value=365,Type="DAYS" --idempotency-token 1234
```

- Para API obtener más información, consulte [IssueCertificate](#) la Referencia de AWS CLI comandos.

## list-certificate-authorities

En el siguiente ejemplo de código, se muestra cómo usar `list-certificate-authorities`.

AWS CLI

Para enumerar sus autoridades de certificación privadas

El siguiente `list-certificate-authorities` comando muestra información sobre todos los datos privados CAs de su cuenta.

```
aws acm-pca list-certificate-authorities --max-results 10
```

- Para API obtener más información, consulte [ListCertificateAuthorities](#) la Referencia de AWS CLI comandos.

## list-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-tags`.

AWS CLI

Para ver una lista de las etiquetas de su autoridad de certificación

El siguiente `list-tags` comando muestra las etiquetas asociadas a la CA privada especificada porARN.

```
aws acm-pca list-tags --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-authority/123455678-1234-1234-1234-123456789012 --max-results 10
```

- Para API obtener más información, consulte [ListTags](#) la Referencia de AWS CLI comandos.

## revoke-certificate

En el siguiente ejemplo de código, se muestra cómo usar `revoke-certificate`.



## AWS CLI

Para revocar un certificado privado

El siguiente `revoke-certificate` comando revoca un certificado privado de la CA identificada por el ARN

```
aws acm-pca revoke-certificate --certificate-authority-arn arn:aws:acm-pca:us-west-2:1234567890:certificate-authority/12345678-1234-1234-1234-123456789012 --certificate-serial 67:07:44:76:83:a9:b7:f4:05:56:27:ff:d5:5c:eb:cc --revocation-reason "KEY_COMPROMISE"
```

- Para API obtener más información, consulte [RevokeCertificate](#) la Referencia de AWS CLI comandos.

## `tag-certificate-authority`

En el siguiente ejemplo de código, se muestra cómo usar `tag-certificate-authority`.

## AWS CLI

Para adjuntar etiquetas a una entidad emisora de certificados privada

El siguiente `tag-certificate-authority` comando adjunta una o más etiquetas a su CA privada.

```
aws acm-pca tag-certificate-authority --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-authority/12345678-1234-1234-1234-123456789012 --tags Key=Admin,Value=Alice
```

- Para API obtener más información, consulte [TagCertificateAuthority](#) la Referencia de AWS CLI comandos.

## `untag-certificate-authority`

En el siguiente ejemplo de código, se muestra cómo usar `untag-certificate-authority`.

## AWS CLI

Para eliminar una o más etiquetas de su entidad emisora de certificados privada

El siguiente `untag-certificate-authority` comando elimina las etiquetas de la CA privada identificada por ARN.

```
aws acm-pca untag-certificate-authority --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-authority/12345678-1234-1234-1234-123456789012 --tags Key=Purpose,Value=Website
```

- Para API obtener más información, consulte [UntagCertificateAuthority](#) la Referencia de AWS CLI comandos.

## update-certificate-authority

En el siguiente ejemplo de código, se muestra cómo usar `update-certificate-authority`.

### AWS CLI

Para actualizar la configuración de su entidad emisora de certificados privada

El siguiente `update-certificate-authority` comando actualiza el estado y la configuración de la CA privada identificada por ARN.

```
aws acm-pca update-certificate-authority --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-authority/12345678-1234-1234-1234-1232456789012 --revocation-configuration file://C:\revoke_config.txt --status "DISABLED"
```

- Para API obtener más información, consulte [UpdateCertificateAuthority](#) la Referencia de AWS CLI comandos.

## AWS Proton ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Proton.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### cancel-service-instance-deployment

En el siguiente ejemplo de código, se muestra cómo usar `cancel-service-instance-deployment`.

#### AWS CLI

Para cancelar el despliegue de una instancia de servicio

En el siguiente `cancel-service-instance-deployment` ejemplo, se cancela la implementación de una instancia de servicio.

```
aws proton cancel-service-instance-deployment \  
  --service-instance-name "instance-one" \  
  --service-name "simple-svc"
```

Salida:

```
{  
  "serviceInstance": {  
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/service-  
instance/instance-one",  
    "createdAt": "2021-04-02T21:29:59.962000+00:00",  
    "deploymentStatus": "CANCELLING",  
    "environmentName": "simple-env",  
    "lastDeploymentAttemptedAt": "2021-04-02T21:45:15.406000+00:00",  
    "lastDeploymentSucceededAt": "2021-04-02T21:38:00.823000+00:00",  
    "name": "instance-one",  
    "serviceName": "simple-svc",  
    "spec": "proton: ServiceSpec\npipeline:\n  
my_sample_pipeline_optional_input: abc\n my_sample_pipeline_required_input:  
'123'\ninstances:\n- name: my-instance\n environment: MySimpleEnv  
\n spec:\n  my_sample_service_instance_optional_input: def\n my_sample_service_instance_required_input: '456'\n- name: my-other-instance\n environment: MySimpleEnv\n spec:\n  my_sample_service_instance_required_input:  
'789'\n",  
  }
```

```
    "templateMajorVersion": "1",
    "templateMinorVersion": "1",
    "templateName": "svc-simple"
  }
}
```

Para obtener más información, consulte [Actualizar una instancia de servicio](#) en la Guía del administrador de AWS Proton o [Actualizar una instancia de servicio](#) en la Guía del usuario de AWS Proton.

- Para API obtener más información, consulte la Referencia [CancelServiceInstanceDeployment](#) de AWS CLI comandos.

## cancel-service-pipeline-deployment

En el siguiente ejemplo de código, se muestra cómo usar `cancel-service-pipeline-deployment`.

### AWS CLI

Para cancelar el despliegue de una canalización de servicios

El siguiente `cancel-service-pipeline-deployment` ejemplo cancela el despliegue de una canalización de servicios.

```
aws proton cancel-service-pipeline-deployment \
  --service-name simple-svc
```

Salida:

```
{
  "pipeline": {
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/pipeline",
    "createdAt": "2021-04-02T21:29:59.962000+00:00",
    "deploymentStatus": "CANCELLING",
    "lastDeploymentAttemptedAt": "2021-04-02T22:02:45.095000+00:00",
    "lastDeploymentSucceededAt": "2021-04-02T21:39:28.991000+00:00",
    "templateMajorVersion": "1",
    "templateMinorVersion": "1",
    "templateName": "svc-simple"
  }
}
```

```
}
```

Para obtener más información, consulte [Actualizar una canalización de servicios](#) en la Guía del administrador de AWS Proton o [Actualizar una canalización de servicios](#) en la Guía del usuario de AWS Proton.

- Para API obtener más información, consulte la Referencia [CancelServicePipelineDeployment](#) de AWS CLI comandos.

## create-service

En el siguiente ejemplo de código, se muestra cómo usar `create-service`.

### AWS CLI

Para crear un servicio

En el siguiente `create-service` ejemplo, se crea un servicio con una canalización de servicios.

```
aws proton create-service \  
  --name "MySimpleService" \  
  --template-name "fargate-service" \  
  --template-major-version "1" \  
  --branch-name "mainline" \  
  --repository-connection-arn "arn:aws:codestar-connections:region-id:account-  
id:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" \  
  --repository-id "myorg/myapp" \  
  --spec file://spec.yaml
```

Contenidos de `spec.yaml`:

```
proton: ServiceSpec  
  
pipeline:  
  my_sample_pipeline_required_input: "hello"  
  my_sample_pipeline_optional_input: "bye"  
  
instances:  
  - name: "acme-network-dev"  
    environment: "ENV_NAME"  
    spec:  
      my_sample_service_instance_required_input: "hi"
```

```
my_sample_service_instance_optional_input: "ho"
```

Salida:

```
{
  "service": {
    "arn": "arn:aws:proton:region-id:123456789012:service/MySimpleService",
    "createdAt": "2020-11-18T19:50:27.460000+00:00",
    "lastModifiedAt": "2020-11-18T19:50:27.460000+00:00",
    "name": "MySimpleService",
    "repositoryConnectionArn": "arn:aws:codestar-connections:region-
id:123456789012connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "repositoryId": "myorg/myapp",
    "status": "CREATE_IN_PROGRESS",
    "templateName": "fargate-service"
  }
}
```

Para obtener más información, consulte [Crear un servicio](#) en la Guía del administrador de AWS Proton y [Crear un servicio](#) en la Guía del usuario de AWS Proton.

- Para API obtener más información, consulte la Referencia [CreateService](#) de AWS CLI comandos.

## delete-service

En el siguiente ejemplo de código, se muestra cómo usar delete-service.

AWS CLI

Eliminación de un servicio

En el siguiente delete-service ejemplo, se elimina un servicio.

```
aws proton delete-service \
  --name "simple-svc"
```

Salida:

```
{
  "service": {
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc",
```

```

    "branchName": "mainline",
    "createdAt": "2020-11-28T22:40:50.512000+00:00",
    "description": "Edit by updating description",
    "lastModifiedAt": "2020-11-29T00:30:39.248000+00:00",
    "name": "simple-svc",
    "repositoryConnectionArn": "arn:aws:codestar-connections:region-
id:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "repositoryId": "myorg/myapp",
    "status": "DELETE_IN_PROGRESS",
    "templateName": "fargate-service"
  }
}

```

Para obtener más información, consulte [Eliminar un servicio](#) en la Guía del administrador de AWS Proton.

- Para API obtener más información, consulte [DeleteService](#) la Referencia de AWS CLI comandos.

## get-service-instance

En el siguiente ejemplo de código, se muestra cómo usar `get-service-instance`.

### AWS CLI

Para obtener los detalles de la instancia de servicio

En el siguiente `get-service-instance` ejemplo, se obtienen datos detallados de una instancia de servicio.

```

aws proton get-service-instance \
  --name "instance-one" \
  --service-name "simple-svc"

```

Salida:

```

{
  "serviceInstance": {
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/service-
instance/instance-one",
    "createdAt": "2020-11-28T22:40:50.512000+00:00",
    "deploymentStatus": "SUCCEEDED",

```

```

    "environmentName": "simple-env",
    "lastDeploymentAttemptedAt": "2020-11-28T22:40:50.512000+00:00",
    "lastDeploymentSucceededAt": "2020-11-28T22:40:50.512000+00:00",
    "name": "instance-one",
    "serviceName": "simple-svc",
    "spec": "proton: ServiceSpec\npipeline:\n
my_sample_pipeline_optional_input: hello world\n
my_sample_pipeline_required_input: pipeline up\ninstances:\n- name: instance-one\n
environment: my-simple-env\n spec:\n   my_sample_service_instance_optional_input:
Ola\n   my_sample_service_instance_required_input: Ciao\n",
    "templateMajorVersion": "1",
    "templateMinorVersion": "0",
    "templateName": "svc-simple"
  }
}

```

Para obtener más información, consulte [Ver los datos del servicio](#) en la Guía del administrador de AWS Proton o [Ver los datos del servicio](#) en la Guía del usuario de AWS Proton.

- Para API obtener más información, consulte la Referencia [GetServiceInstance](#) de AWS CLI comandos.

## get-service

En el siguiente ejemplo de código, se muestra cómo usar `get-service`.

### AWS CLI

Para obtener detalles del servicio

En el siguiente `get-service` ejemplo, se obtienen datos detallados de un servicio.

```

aws proton get-service \
  --name "simple-svc"

```

Salida:

```

{
  "service": {
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc",
    "branchName": "mainline",
    "createdAt": "2020-11-28T22:40:50.512000+00:00",

```



```

    "lastModifiedAt": "2020-11-28T22:44:51.207000+00:00",
    "name": "simple-svc",
    "pipeline": {
      "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/
pipeline/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "createdAt": "2020-11-28T22:40:50.512000+00:00",
      "deploymentStatus": "SUCCEEDED",
      "lastDeploymentAttemptedAt": "2020-11-28T22:40:50.512000+00:00",
      "lastDeploymentSucceededAt": "2020-11-28T22:40:50.512000+00:00",
      "spec": "proton: ServiceSpec\npipeline:\n
my_sample_pipeline_required_input: hello\n my_sample_pipeline_optional_input:
bye\ninstances:\n- name: instance-svc-simple\n environment: my-simple-
env\n spec:\n  my_sample_service_instance_required_input: hi\n
my_sample_service_instance_optional_input: ho\n",
      "templateMajorVersion": "1",
      "templateMinorVersion": "1",
      "templateName": "svc-simple"
    },
    "repositoryConnectionArn": "arn:aws:codestar-connections:region-
id:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "repositoryId": "myorg/myapp",
    "spec": "proton: ServiceSpec\npipeline:\n
my_sample_pipeline_required_input: hello\n my_sample_pipeline_optional_input:
bye\ninstances:\n- name: instance-svc-simple\n environment: my-simple-
env\n spec:\n  my_sample_service_instance_required_input: hi\n
my_sample_service_instance_optional_input: ho\n",
    "status": "ACTIVE",
    "templateName": "svc-simple"
  }
}

```

Para obtener más información, consulte [Ver los datos del servicio](#) en la Guía del administrador de AWS Proton o [Ver los datos del servicio](#) en la Guía del usuario de AWS Proton.

- Para API obtener más información, consulte la Referencia [GetService](#) de AWS CLI comandos.

## list-service-instances

En el siguiente ejemplo de código, se muestra cómo usar `list-service-instances`.

### AWS CLI

Ejemplo 1: Para enumerar todas las instancias de servicio

En el siguiente `list-service-instances` ejemplo, se enumeran las instancias de servicio.

```
aws proton list-service-instances
```

Salida:

```
{
  "serviceInstances": [
    {
      "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/
service-instance/instance-one",
      "createdAt": "2020-11-28T22:40:50.512000+00:00",
      "deploymentStatus": "SUCCEEDED",
      "environmentArn": "arn:aws:proton:region-id:123456789012:environment/
simple-env",
      "lastDeploymentAttemptedAt": "2020-11-28T22:40:50.512000+00:00",
      "lastDeploymentSucceededAt": "2020-11-28T22:40:50.512000+00:00",
      "name": "instance-one",
      "serviceName": "simple-svc",
      "templateMajorVersion": "1",
      "templateMinorVersion": "0",
      "templateName": "fargate-service"
    }
  ]
}
```

Para obtener más información, consulte [Ver los datos de la instancia de servicio](#) en la Guía del administrador de AWS Proton o [Ver los datos de la instancia de servicio](#) en la Guía del usuario de AWS Proton.

Ejemplo 2: Para enumerar la instancia de servicio especificada

En el siguiente `get-service-instance` ejemplo, se obtiene una instancia de servicio.

```
aws proton get-service-instance \
  --name "instance-one" \
  --service-name "simple-svc"
```

Salida:

```
{
```

```

    "serviceInstance": {
      "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/service-
instance/instance-one",
      "createdAt": "2020-11-28T22:40:50.512000+00:00",
      "deploymentStatus": "SUCCEEDED",
      "environmentName": "simple-env",
      "lastDeploymentAttemptedAt": "2020-11-28T22:40:50.512000+00:00",
      "lastDeploymentSucceededAt": "2020-11-28T22:40:50.512000+00:00",
      "name": "instance-one",
      "serviceName": "simple-svc",
      "spec": "proton: ServiceSpec\npipeline:\n
my_sample_pipeline_optional_input: hello world\n
my_sample_pipeline_required_input: pipeline up\ninstances:\n- name: instance-one\n
environment: my-simple-env\n spec:\n   my_sample_service_instance_optional_input:
Ola\n   my_sample_service_instance_required_input: Ciao\n",
      "templateMajorVersion": "1",
      "templateMinorVersion": "0",
      "templateName": "svc-simple"
    }
  }
}

```

Para obtener más información, consulte [Ver los datos de la instancia de servicio](#) en la Guía del administrador de AWS Proton o [Ver los datos de la instancia de servicio](#) en la Guía del usuario de AWS Proton.

- Para API obtener más información, consulte la Referencia [ListServiceInstances](#) de AWS CLI comandos.

## update-service-instance

En el siguiente ejemplo de código, se muestra cómo usar `update-service-instance`.

### AWS CLI

Para actualizar una instancia de servicio a una nueva versión secundaria

En el siguiente `update-service-instance` ejemplo, se actualiza una instancia de servicio a una nueva versión secundaria de su plantilla de servicio que añade una nueva instancia `my-other-instance` denominada «» con una nueva entrada obligatoria.

```

aws proton update-service-instance \
  --service-name "simple-svc" \

```

```
--spec "file://service-spec.yaml" \
--template-major-version "1" \
--template-minor-version "1" \
--deployment-type "MINOR_VERSION" \
--name "instance-one"
```

### Contenidos de service-spec.yaml:

```
proton: ServiceSpec
pipeline:
  my_sample_pipeline_optional_input: "abc"
  my_sample_pipeline_required_input: "123"
instances:
  - name: "instance-one"
    environment: "simple-env"
    spec:
      my_sample_service_instance_optional_input: "def"
      my_sample_service_instance_required_input: "456"
  - name: "my-other-instance"
    environment: "simple-env"
    spec:
      my_sample_service_instance_required_input: "789"
```

### Salida:

```
{
  "serviceInstance": {
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/service-
instance/instance-one",
    "createdAt": "2021-04-02T21:29:59.962000+00:00",
    "deploymentStatus": "IN_PROGRESS",
    "environmentName": "arn:aws:proton:region-id:123456789012:environment/
simple-env",
    "lastDeploymentAttemptedAt": "2021-04-02T21:38:00.823000+00:00",
    "lastDeploymentSucceededAt": "2021-04-02T21:29:59.962000+00:00",
    "name": "instance-one",
    "serviceName": "simple-svc",
    "templateMajorVersion": "1",
    "templateMinorVersion": "0",
    "templateName": "svc-simple"
  }
}
```

Para obtener más información, consulte [Actualizar una instancia de servicio](#) en la Guía del administrador de AWS Proton o [Actualizar una instancia de servicio](#) en la Guía del usuario de AWS Proton.

- Para API obtener más información, consulte la Referencia [UpdateServiceInstance](#) de AWS CLI comandos.

## update-service-pipeline

En el siguiente ejemplo de código, se muestra cómo usar `update-service-pipeline`.

### AWS CLI

Para actualizar una canalización de servicios

En el siguiente `update-service-pipeline` ejemplo, se actualiza una canalización de servicios a una nueva versión secundaria de su plantilla de servicio.

```
aws proton update-service-pipeline \
  --service-name "simple-svc" \
  --spec "file://service-spec.yaml" \
  --template-major-version "1" \
  --template-minor-version "1" \
  --deployment-type "MINOR_VERSION"
```

Salida:

```
{
  "pipeline": {
    "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/pipeline/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "createdAt": "2021-04-02T21:29:59.962000+00:00",
    "deploymentStatus": "IN_PROGRESS",
    "lastDeploymentAttemptedAt": "2021-04-02T21:39:28.991000+00:00",
    "lastDeploymentSucceededAt": "2021-04-02T21:29:59.962000+00:00",
    "spec": "proton: ServiceSpec\n\npipeline:\n
my_sample_pipeline_optional_input: \"abc\"\n my_sample_pipeline_required_input:
\"123\"\n\ninstances:\n - name: \"my-instance\"\n   environment: \"MySimpleEnv
\"\n   spec:\n     my_sample_service_instance_optional_input: \"def
\"\n     my_sample_service_instance_required_input: \"456\"\n - name:
\"my-other-instance\"\n   environment: \"MySimpleEnv\"\n   spec:\n
my_sample_service_instance_required_input: \"789\"\n\n",
```

```
    "templateMajorVersion": "1",
    "templateMinorVersion": "0",
    "templateName": "svc-simple"
  }
}
```

Para obtener más información, consulte [Actualizar una canalización de servicios](#) en la Guía del administrador de AWS Proton o [Actualizar una canalización de servicios](#) en la Guía del usuario de AWS Proton.

- Para API obtener más información, consulte la Referencia [UpdateServicePipeline](#) de AWS CLI comandos.

## update-service

En el siguiente ejemplo de código, se muestra cómo usar `update-service`.

### AWS CLI

Para actualizar un servicio

En el siguiente `update-service` ejemplo, se edita la descripción de un servicio.

```
aws proton update-service \
  --name "MySimpleService" \
  --description "Edit by updating description"
```

Salida:

```
{
  "service": {
    "arn": "arn:aws:proton:region-id:123456789012:service/MySimpleService",
    "branchName": "mainline",
    "createdAt": "2021-03-12T22:39:42.318000+00:00",
    "description": "Edit by updating description",
    "lastModifiedAt": "2021-03-12T22:44:21.975000+00:00",
    "name": "MySimpleService",
    "repositoryConnectionArn": "arn:aws:codestar-connections:region-
id:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "repositoryId": "myorg/myapp",
    "status": "ACTIVE",
    "templateName": "fargate-service"
  }
}
```

```
}  
}
```

Para obtener más información, consulte [Editar un servicio](#) en la Guía del administrador de AWS Proton o [Editar un servicio](#) en la Guía del usuario de AWS Proton.

- Para API obtener más información, consulte la Referencia [UpdateService](#) de AWS CLI comandos.

## QLDBejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface withQLDB.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **cancel-journal-kinesis-stream**

En el siguiente ejemplo de código, se muestra cómo usar `cancel-journal-kinesis-stream`.

AWS CLI

Para cancelar la transmisión de un diario

En el siguiente `cancel-journal-kinesis-stream` ejemplo, se cancela el flujo de diario especificado de un libro mayor.

```
aws qldb cancel-journal-kinesis-stream \  
  --ledger-name myExampleLedger \  
  --stream-name myExampleStream \  
  --stream-id myExampleStreamId \  
  --force
```

```
--stream-id 7ISCKqwe4y25YyHLzYUFAf
```

Salida:

```
{  
  "StreamId": "7ISCKqwe4y25YyHLzYUFAf"  
}
```

Para obtener más información, consulte [Transmitir datos de revistas de Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [CancelJournalKinesisStream](#) la Referencia de AWS CLI comandos.

## create-ledger

En el siguiente ejemplo de código, se muestra cómo usar `create-ledger`.

### AWS CLI

Ejemplo 1: Creación de un libro mayor con las propiedades predeterminadas

En el siguiente ejemplo de `create-ledger` se crea un libro mayor con el nombre `myExampleLedger` y el modo de permisos `STANDARD`. Los parámetros opcionales para la protección contra la eliminación y la AWS KMS clave no están especificados, por lo que los valores predeterminados son una clave `true` propia y AWS una KMS clave propia, respectivamente.

```
aws qlldb create-ledger \  
  --name myExampleLedger \  
  --permissions-mode STANDARD
```

Salida:

```
{  
  "State": "CREATING",  
  "Arn": "arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger",  
  "DeletionProtection": true,  
  "CreationDateTime": 1568839243.951,  
  "Name": "myExampleLedger",
```



```
"PermissionsMode": "STANDARD"
}
```

Ejemplo 2: Para crear un libro mayor con la protección contra la eliminación desactivada, una KMS clave gestionada por el cliente y etiquetas especificadas

En el siguiente ejemplo de `create-ledger` se crea un libro mayor con el nombre `myExampleLedger2` y el modo de permisos `STANDARD`. La función de protección contra la eliminación está desactivada, la KMS clave gestionada por el cliente especificada se utiliza para el cifrado en reposo y las etiquetas especificadas se adjuntan al recurso.

```
aws qlldb create-ledger \
  --name myExampleLedger2 \
  --permissions-mode STANDARD \
  --no-deletion-protection \
  --kms-key arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --tags IsTest=true,Domain=Test
```

Salida:

```
{
  "Arn": "arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger2",
  "DeletionProtection": false,
  "CreationDateTime": 1568839543.557,
  "State": "CREATING",
  "Name": "myExampleLedger2",
  "PermissionsMode": "STANDARD",
  "KmsKeyArn": "arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Para obtener más información, consulte [Operaciones básicas de Amazon QLDB Ledgers en la Guía para QLDB](#) desarrolladores de Amazon.

- Para API obtener más información, consulte [CreateLedger](#) la Referencia de AWS CLI comandos.

## delete-ledger

En el siguiente ejemplo de código, se muestra cómo usar `delete-ledger`.

## AWS CLI

Para eliminar un libro mayor

En el siguiente delete-ledger ejemplo, se elimina el libro mayor especificado.

```
aws qldb delete-ledger \  
  --name myExampleLedger
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Operaciones básicas de Amazon QLDB Ledgers en la Guía para QLDB](#) desarrolladores de Amazon.

- Para API obtener más información, consulte [DeleteLedger](#) la Referencia de AWS CLI comandos.

## describe-journal-kinesis-stream

En el siguiente ejemplo de código, se muestra cómo usar describe-journal-kinesis-stream.

### AWS CLI

Para describir un flujo de diario

En el siguiente describe-journal-kinesis-stream ejemplo, se muestran los detalles del flujo de diario especificado desde un libro mayor.

```
aws qldb describe-journal-kinesis-stream \  
  --ledger-name myExampleLedger \  
  --stream-id 7ISCKqwe4y25YyHLzYUFaf
```

Salida:

```
{  
  "Stream": {  
    "LedgerName": "myExampleLedger",  
    "CreationTime": 1591221984.677,  
    "InclusiveStartTime": 1590710400.0,  
    "ExclusiveEndTime": 1590796799.0,  
    "RoleArn": "arn:aws:iam::123456789012:role/my-kinesis-stream-role",  
    "StreamId": "7ISCKqwe4y25YyHLzYUFaf",
```

```

    "Arn": "arn:aws:qldb:us-east-1:123456789012:stream/
myExampleLedger/7ISCKqwe4y25YyHLzYUFAf",
    "Status": "ACTIVE",
    "KinesisConfiguration": {
        "StreamArn": "arn:aws:kinesis:us-east-1:123456789012:stream/stream-for-
qldb",
        "AggregationEnabled": true
    },
    "StreamName": "myExampleLedger-stream"
}
}

```

Para obtener más información, consulte [Transmitir datos de revistas de Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeJournalKinesisStream](#) la Referencia de AWS CLI comandos.

## describe-journal-s3-export

En el siguiente ejemplo de código, se muestra cómo usar `describe-journal-s3-export`.

### AWS CLI

Para describir un trabajo de exportación de revistas

En el siguiente `describe-journal-s3-export` ejemplo, se muestran los detalles del trabajo de exportación especificado desde un libro mayor.

```

aws qldb describe-journal-s3-export \
  --name myExampleLedger \
  --export-id ADR2ONPKN5LINYGb4dp7yZ

```

Salida:

```

{
  "ExportDescription": {
    "S3ExportConfiguration": {
      "Bucket": "awsExampleBucket",
      "Prefix": "ledgerexport1/",
      "EncryptionConfiguration": {
        "ObjectEncryptionType": "SSE_S3"
      }
    }
  }
}

```

```

    }
  },
  "RoleArn": "arn:aws:iam::123456789012:role/my-s3-export-role",
  "Status": "COMPLETED",
  "ExportCreationTime": 1568847801.418,
  "InclusiveStartTime": 1568764800.0,
  "ExclusiveEndTime": 1568847599.0,
  "LedgerName": "myExampleLedger",
  "ExportId": "ADR20NPKN5LINYGb4dp7yZ"
}
}

```

Para obtener más información, consulte [Exportación de su diario en Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeJournalS3Export](#) en la referencia de AWS CLI comandos.

## describe-ledger

En el siguiente ejemplo de código, se muestra cómo usar `describe-ledger`.

### AWS CLI

Para describir un libro mayor

En el siguiente `describe-ledger` ejemplo, se muestran los detalles del libro mayor especificado.

```
aws qlldb describe-ledger \
  --name myExampleLedger
```

Salida:

```

{
  "CreationDateTime": 1568839243.951,
  "Arn": "arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger",
  "State": "ACTIVE",
  "Name": "myExampleLedger",
  "DeletionProtection": true,
  "PermissionsMode": "STANDARD",
  "EncryptionDescription": {

```

```

    "KmsKeyArn": "arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
    "EncryptionStatus": "ENABLED"
  }
}

```

Para obtener más información, consulte [Operaciones básicas de Amazon QLDB Ledgers en la Guía para QLDB](#) desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeLedger](#) la Referencia de AWS CLI comandos.

## export-journal-to-s3

En el siguiente ejemplo de código, se muestra cómo usar `export-journal-to-s3`.

### AWS CLI

Para exportar bloques de diario a S3

En el siguiente `export-journal-to-s3` ejemplo, se crea un trabajo de exportación de bloques de diario dentro de un intervalo de fechas y horas especificado a partir de un libro mayor con ese nombre `myExampleLedger`. El trabajo de exportación escribe los bloques en un bucket de Amazon S3 específico.

```

aws qldb export-journal-to-s3 \
  --name myExampleLedger \
  --inclusive-start-time 2019-09-18T00:00:00Z \
  --exclusive-end-time 2019-09-18T22:59:59Z \
  --role-arn arn:aws:iam::123456789012:role/my-s3-export-role \
  --s3-export-configuration file://my-s3-export-config.json

```

Contenidos de `my-s3-export-config.json`:

```

{
  "Bucket": "awsExampleBucket",
  "Prefix": "ledgerexport1/",
  "EncryptionConfiguration": {
    "ObjectEncryptionType": "SSE_S3"
  }
}

```

**Salida:**

```
{
  "ExportId": "ADR20NPKN5LINYGb4dp7yZ"
}
```

Para obtener más información, consulte [Exportación de su diario en Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [ExportJournalToS3](#) en la Referencia de AWS CLI comandos.

**get-block**

En el siguiente ejemplo de código, se muestra cómo usar `get-block`.

## AWS CLI

Ejemplo 1: Para obtener un bloque de diario y una prueba de verificación mediante archivos de entrada

En el siguiente `get-block` ejemplo, se solicita un objeto de datos en bloque y una prueba del libro mayor especificado. La solicitud es para una dirección de resumen y una dirección de bloque especificadas.

```
aws qlldb get-block \
  --name vehicle-registration \
  --block-address file://myblockaddress.json \
  --digest-tip-address file://mydigesttipaddress.json
```

Contenidos de `myblockaddress.json`:

```
{
  "IonText": "{strandId:\"KmA3ZZca7vAIiJAK9S5Iw1\",sequenceNo:100}"
}
```

Contenidos de `mydigesttipaddress.json`:

```
{
  "IonText": "{strandId:\"KmA3ZZca7vAIiJAK9S5Iw1\",sequenceNo:123}"
}
```

}

**Salida:**

```
{
  "Block": {
    "IonText": "{blockAddress:{strandId:\\"KmA3ZZca7vAIiJAK9S5Iwl
\\",sequenceNo:100},transactionId:\\"FnQeJBAicTX0Ah32ZnVtSX
\\",blockTimestamp:2019-09-16T19:37:05.360Z,blockHash:
{{NoChM92yKRuJAb/jeLd1VnYn4DHiWIf071ACfic9uHc=}},entriesHash:
{{105L0siKV14SDbuaYnH7uwXzUvqzIwUiRLXGbTyj/nY=}},previousBlockHash:
{{7kewBXhpdBc1cZKxhVmpoMHPUGOJtWQD0iY2LPfZkYA=}},entriesHashList:
[{{eRSwnmAM7WWANWDD5iG0yK+T4tDXyzUq6HZ/0fgLHos=}},{{mHVex/
yJHAWjFPpwhBuH2GKXmKjK2FBa9faquUVNtg=}}],
{{y5cCB7p0AIUfsVQ1j0TqtE97b4b4oo1R0vnYyE5wWM=}},{{TvTXygML1bMe6NvEZtGkX
+KR+W/EJ14qD1mmV77KZQg=}}}],transactionInfo:{statements:[{statement:
\\"FROM VehicleRegistration AS r \\nWHERE r.VIN = '1N4AL11D75C109151'\\
\\nINSERT INTO r.Owners.SecondaryOwners\\n  VALUE { 'PersonId' :
'CMVdR77XP8zAg1mmFDGTvt' }\\",startTime:2019-09-16T19:37:05.302Z,statementDigest:
{{jcgPX2vs0J0waum4qmDYtn1pCAT9xKNIzA+2k4R+mxA=}}}],documents:
{JUJgkIcNbhS2goq8RqLuZ4:{tableName:\\"VehicleRegistration\\",tableId:
\\"BFJKdXgzT9oF4wjMbuXy4G\\",statements:[0]}}],revisions:[{blockAddress:
{strandId:\\"KmA3ZZca7vAIiJAK9S5Iwl\\",sequenceNo:100},hash:
{{mHVex/yJHAWjFPpwhBuH2GKXmKjK2FBa9faquUVNtg=}},data:{VIN:
\\"1N4AL11D75C109151\\",LicensePlateNumber:\\"LEWISR261LL\\",State:\\"WA
\\",PendingPenaltyTicketAmount:90.25,ValidFromDate:2017-08-21,ValidToDate:2020-05-11,Owners:
{PrimaryOwner:{PersonId:\\"BFJKdXhnLRT27sXBnojNGW\\"},SecondaryOwners:
[{{PersonId:\\"CMVdR77XP8zAg1mmFDGTvt\\"}]}],City:\\"Everett\\"},metadata:{id:
\\"JUJgkIcNbhS2goq8RqLuZ4\\",version:3,txTime:2019-09-16T19:37:05.344Z,txId:
\\"FnQeJBAicTX0Ah32ZnVtSX\\"}}]}}",
  },
  "Proof": {
    "IonText": "[{{13+EXs69K1+rehlqyWLkt+oHDlw4Zi9pCLW/t/mgTPM=}},
{{48CXG3ehPqsxCYd34EEa8Fso00RpWwA08010RJKf3Do=}},{{9UnwnKSQT0i3ge1JMVa
+tMIqCEDaOPTkwxmyHSn8UPQ=}},{{3nW6Vryghk+7pd6wFCtLufgPM6qXHyTNeCb1sCwcDaI=}},
{{Irb5fNhBrNEQ1VPhz1nGT/ZQPadSmgfdtMYcwkN0xoI=}},{{+3CwpYG/ytf/
vq9GidpzSx6JJiLXt1hMQWnNq0y3jfY=}},{{NPx6cRhwsiy5m9UEWS5JTJrZoUd02jB0AA0myZAT
+qE=}}]"
  }
}
```

Para obtener más información, consulta la [verificación de datos en Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

Ejemplo 2: Para obtener un bloque de diario y una prueba de verificación mediante una sintaxis abreviada

En el siguiente `get-block` ejemplo, se solicita un objeto de datos en bloque y una prueba del libro mayor especificado mediante una sintaxis abreviada. La solicitud es para una dirección de sugerencia resumida y una dirección de bloque especificadas.

```
aws qlldb get-block \
  --name vehicle-registration \
  --block-address 'IonText="{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:100}"'
  \
  --digest-tip-address 'IonText="{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:123}"'
```

Salida:

```
{
  "Block": {
    "IonText": "{blockAddress:{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:100},transactionId:\\"FnQeJBAicTX0Ah32ZnVtSX\\",blockTimestamp:2019-09-16T19:37:05.360Z,blockHash:{{NoChM92yKRuJAb/jeLd1VnYn4DHiWIf071ACfic9uHc=}},entriesHash:{{105L0siKV14SDbuaYnH7uwXzUvqzIwUiRLXGbTyj/nY=}},previousBlockHash:{{7kewBXhpdBc1cZKxhVmpoMhpUG0JtWQD0iY2LPfZkYA=}},entriesHashList:{{eRSwnmAM7WWANWDd5iG0yK+T4tDXyzUq6HZ/0fgLHos=}},{{mHVex/yjHAWjFPpwhBuH2GKXmKJjK2FBa9faquUVNtg=}},{{y5cCB7p0AIUfsVQ1j0TqtE97b4b4oo1R0vnYyE5wWM=}},{{TvTXygML1bMe6NvEZtGkX+KR+W/EJl4qD1mmV77KZQg=}}},transactionInfo:{statements:[{statement:\\"FROM VehicleRegistration AS r \\nWHERE r.VIN = '1N4AL11D75C109151'\\nINSERT INTO r.Owners.SecondaryOwners\\n  VALUE { 'PersonId' : 'CMVdR77XP8zAg1mmFDGTvt' }\\",startTime:2019-09-16T19:37:05.302Z,statementDigest:{{jcgPX2vs0J0waum4qmDYtn1pCAT9xKNIzA+2k4R+mxA=}}}],documents:[JUJgkIcNbhS2goq8RqLuZ4:{tableName:\\"VehicleRegistration\\",tableId:\\"BFJKdXgzt9oF4wjMbuXy4G\\",statements:[0]}],revisions:[{blockAddress:{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:100},hash:{{mHVex/yjHAWjFPpwhBuH2GKXmKJjK2FBa9faquUVNtg=}},data:{VIN:\\"1N4AL11D75C109151\\",LicensePlateNumber:\\"LEWISR261LL\\",State:\\"WA\\",PendingPenaltyTicketAmount:90.25,ValidFromDate:2017-08-21,ValidToDate:2020-05-11,Owners:{PrimaryOwner:{PersonId:\\"BFJKdXhnLRT27sXBnojNGW\\"},SecondaryOwners:[{PersonId:\\"CMVdR77XP8zAg1mmFDGTvt\\"}]}],City:\\"Everett\\"},metadata:{id:\\"JUJgkIcNbhS2goq8RqLuZ4\\",version:3,txTime:2019-09-16T19:37:05.344Z,txId:\\"FnQeJBAicTX0Ah32ZnVtSX\\"}}]}]"},
  },
```



```

    "Proof": {
      "IonText": "[{{13+EXs69K1+rehlqyWLkt+oHDlw4Zi9pCLW/t/mgTPM=}},
{{48CXG3ehPqsxCYd34EEa8Fso00RpWwA08010RJKf3Do=}}, {{9UnwnKSQT0i3ge1JMVa
+tMIqCEDa0PTkwxmyHSn8UPQ=}}, {{3nW6Vryghk+7pd6wFCtLufgPM6qXHyTNeCb1sCwcDaI=}},
{{Irb5fNhBrNEQ1VPhzlnGT/ZQPadSmgfdtMYcwkN0xoI=}}, {{+3CWpYG/ytf/
vq9GidpzSx6JJiLXt1hMQWnq0y3jfY=}}, {{NPx6cRhwsiy5m9UEWS5JTJrZoUd02jB0AA0myZAT
+qE=}}]"
    }
  }
}

```

Para obtener más información, consulta la [verificación de datos en Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [GetBlock](#) la Referencia de AWS CLI comandos.

## get-digest

En el siguiente ejemplo de código, se muestra cómo usar `get-digest`.

### AWS CLI

Para obtener un resumen de un libro mayor

En el siguiente `get-digest` ejemplo, se solicita un resumen del libro mayor especificado en el último bloque comprometido del diario.

```

aws qlldb get-digest \
  --name vehicle-registration

```

Salida:

```

{
  "Digest": "6m6BMXobbJKpMhahwVthAEsN6awgnHK62Qq5McGP1Gk=",
  "DigestTipAddress": {
    "IonText": "{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\", sequenceNo:123}"
  }
}

```

Para obtener más información, consulta la [verificación de datos en Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [GetDigest](#) la Referencia de AWS CLI comandos.

## get-revision

En el siguiente ejemplo de código, se muestra cómo usar `get-revision`.

### AWS CLI

Ejemplo 1: Para obtener una revisión de un documento y una prueba de verificación mediante archivos de entrada

En el siguiente `get-revision` ejemplo, se solicita un objeto de datos de revisión y una prueba del libro mayor especificado. La solicitud se refiere a una dirección de sugerencia resumida, un identificador de documento y una dirección de bloque específicos de la revisión.

```
aws qlldb get-revision \
  --name vehicle-registration \
  --block-address file://myblockaddress.json \
  --document-id JUJgkIcNbhS2goq8RqLuZ4 \
  --digest-tip-address file://mydigesttipaddress.json
```

Contenidos de `myblockaddress.json`:

```
{
  "IonText": "{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:100}"
}
```

Contenidos de `mydigesttipaddress.json`:

```
{
  "IonText": "{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:123}"
}
```

Salida:

```
{
  "Revision": {
    "IonText": "{blockAddress:{strandId:\\"KmA3ZZca7vAIiJAK9S5Iw1\\",sequenceNo:100},hash:{{mHVex/yjHAWjFPpwhBuH2GKXmKJjK2FBa9faquUVNtg=}},data:
    {VIN:\\"1N4AL11D75C109151\\",LicensePlateNumber:\\"LEWISR261LL\\",State:\\"WA\\",PendingPenaltyTicketAmount:90.25,ValidFromDate:2017-08-21,ValidToDate:2020-05-11,Owners:
    {PrimaryOwner:{PersonId:\\"BFJKdXhnLRT27sXBnojNGW\\"},SecondaryOwners:
```

```
[{"PersonId":"CMVdR77XP8zAg1mmFDGTvt\"}],City\":\"Everett\"},metadata:{id:
\"JUJgkIcNbhS2goq8RqLuZ4\",version:3,txTime:2019-09-16T19:37:05.344Z,txId:
\"FnQeJBAicTX0Ah32ZnVtSX\"}]
},
  \"Proof\": {
    \"IonText\": \"[{{eRSwnmAM7WWANWd5iG0yK+T4tDXyzUq6HZ/0fgLHos=}},{{VV1rdaNuf
+yJZVG1msM6gr2T52QvB08Lg+KgpjcnWAU=}},
{{7kewBXhpdBc1cZKxhVmpoMHPUG0JtWQD0iY2LPfZkYA=}},{{13+EXs69K1+rehlqyWLkt
+oHD1w4Zi9pCLW/t/mgTPM=}},{{48CXG3ehPqsxCYd34EEa8Fso00RpWWA08010RJKf3Do=}},
{{9UnwnKSQT0i3ge1JMVa+tMIqCEDaOPTkwxmyHSn8UPQ=}},{{3nW6Vryghk
+7pd6wFCtLufgPM6qXHyTNECb1sCwcDaI=}},{{Irb5fNhBrNEQ1VPhz1nGT/
ZQPadSmgfdtMYcwkN0xoI=}},{{+3CWpYG/ytf/vq9GidpzSx6JJiLXt1hMQWNnq0y3jfY=}},
{{NPx6cRhwsiy5m9UEWS5JTJrZoUd02jB0AA0myZAT+qE=}}]\"
  }
}
```

Para obtener más información, consulta la [verificación de datos en Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

Ejemplo 2: Para obtener una revisión de un documento y una prueba de verificación mediante una sintaxis abreviada

En el siguiente `get-revision` ejemplo, se solicita un objeto de datos de revisión y una prueba del libro mayor especificado mediante una sintaxis abreviada. La solicitud se refiere a una dirección de sugerencia resumida, un identificador de documento y una dirección de bloque específicos de la revisión.

```
aws qlldb get-revision \
  --name vehicle-registration \
  --block-address 'IonText="{strandId:\"KmA3ZZca7vAIiJAK9S5Iw1\",sequenceNo:100}"'
  \
  --document-id JUJgkIcNbhS2goq8RqLuZ4 \
  --digest-tip-address 'IonText="{strandId:\"KmA3ZZca7vAIiJAK9S5Iw1\",sequenceNo:123}"'
```

Salida:

```
{
  \"Revision\": {
    \"IonText\": \"{blockAddress:{strandId:\"KmA3ZZca7vAIiJAK9S5Iw1
\",sequenceNo:100},hash:{{mHVex/yjHAWjFPpwhBuH2GKXmKjK2FBa9faquUVNtg=}},data:
{VIN:\"1N4AL11D75C109151\",LicensePlateNumber:\"LEWISR261LL\",State:\"WA
```

```

\",PendingPenaltyTicketAmount:90.25,ValidFromDate:2017-08-21,ValidToDate:2020-05-11,Owners:
{PrimaryOwner:{PersonId:\"BFJKdXhnLRT27sXBnojNGW\"},SecondaryOwners:
[PersonId:\"CMVdR77XP8zAg1mmFDGTvt\"}],City:\"Everett\"},metadata:{id:
\"JUJgkIcNbhS2goq8RqLuZ4\",version:3,txTime:2019-09-16T19:37:05.344Z,txId:
\"FnQeJBAicTX0Ah32ZnVtSX\"}}
  },
  \"Proof\": {
    \"IonText\": \"[{{eRSwnmAM7WWANWDD5iG0yK+T4tDXyzUq6HZ/0fgLHos=}},{{VV1rdaNuf
+yJZVG1msM6gr2T52QvB08Lg+KgpjcnWAU=}},
{{7kewBXhpdBc1cZKxhVmpoMHPUGOJtWQD0iY2LPfZkYA=}},{{13+EXs69K1+rehlqyWLkt
+oHD1w4Zi9pCLW/t/mgTPM=}},{{48CXG3ehPqsxCYd34EEa8Fso00RpWwA08010RJKf3Do=}},
{{9UnwnKSQT0i3ge1JMVa+tMIqCEDaOPTkWxmyHSn8UPQ=}},{{3nW6Vryghk
+7pd6wFCtLufgPM6qXHyTNECb1sCwcDaI=}},{{Irb5fNhBrNEQ1VPhz1nGT/
ZQPadSmgfdtMYcwkN0xoI=}},{{+3CwPYG/ytf/vq9GidpzSx6JJiLXt1hMQWNnq0y3jfY=}},
{{NPx6cRhwsiy5m9UEWS5JTJrZoUd02jB0AA0myZAT+qE=}}]\"
  }
}

```

Para obtener más información, consulta la [verificación de datos en Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [GetRevision](#) la Referencia de AWS CLI comandos.

## list-journal-kinesis-streams-for-ledger

En el siguiente ejemplo de código, se muestra cómo usar `list-journal-kinesis-streams-for-ledger`.

### AWS CLI

Para enumerar los flujos de diario de un libro mayor

En el siguiente `list-journal-kinesis-streams-for-ledger` ejemplo, se muestran los flujos de diario del libro mayor especificado.

```

aws qlldb list-journal-kinesis-streams-for-ledger \
  --ledger-name myExampleLedger

```

Salida:

```

{
  \"Streams\": [

```

```

    {
      "LedgerName": "myExampleLedger",
      "CreationTime": 1591221984.677,
      "InclusiveStartTime": 1590710400.0,
      "ExclusiveEndTime": 1590796799.0,
      "RoleArn": "arn:aws:iam::123456789012:role/my-kinesis-stream-role",
      "StreamId": "7ISCKqwe4y25YyHLzYUFAf",
      "Arn": "arn:aws:qldb:us-east-1:123456789012:stream/
myExampleLedger/7ISCKqwe4y25YyHLzYUFAf",
      "Status": "ACTIVE",
      "KinesisConfiguration": {
        "StreamArn": "arn:aws:kinesis:us-east-1:123456789012:stream/stream-
for-qldb",
        "AggregationEnabled": true
      },
      "StreamName": "myExampleLedger-stream"
    }
  ]
}

```

Para obtener más información, consulte [Transmitir datos de revistas de Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [ListJournalKinesisStreamsForLedger](#) la Referencia de AWS CLI comandos.

## list-journal-s3-exports-for-ledger

En el siguiente ejemplo de código, se muestra cómo usar `list-journal-s3-exports-for-ledger`.

### AWS CLI

Para enumerar los trabajos de exportación de diarios para un libro mayor

En el siguiente `list-journal-s3-exports-for-ledger` ejemplo, se enumeran los trabajos de exportación de asientos para el libro mayor especificado.

```

aws qldb list-journal-s3-exports-for-ledger \
  --name myExampleLedger

```

Salida:

```
{
  "JournalS3Exports": [
    {
      "LedgerName": "myExampleLedger",
      "ExclusiveEndTime": 1568847599.0,
      "ExportCreationTime": 1568847801.418,
      "S3ExportConfiguration": {
        "Bucket": "awsExampleBucket",
        "Prefix": "ledgerexport1/",
        "EncryptionConfiguration": {
          "ObjectEncryptionType": "SSE_S3"
        }
      },
      "ExportId": "ADR20NPKN5LINYGb4dp7yZ",
      "RoleArn": "arn:aws:iam::123456789012:role/qlldb-s3-export",
      "InclusiveStartTime": 1568764800.0,
      "Status": "IN_PROGRESS"
    }
  ]
}
```

Para obtener más información, consulte [Exportación de su diario en Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [ListJournalS3 ExportsForLedger](#) en la Referencia de AWS CLI comandos.

## **list-journal-s3-exports**

En el siguiente ejemplo de código, se muestra cómo usar `list-journal-s3-exports`.

### AWS CLI

Para enumerar los trabajos de exportación de revistas

En el siguiente `list-journal-s3-exports` ejemplo, se enumeran los trabajos de exportación de diarios de todos los libros de contabilidad asociados a la AWS cuenta corriente y a la región.

```
aws qlldb list-journal-s3-exports
```

Salida:

```

{
  "JournalS3Exports": [
    {
      "Status": "IN_PROGRESS",
      "LedgerName": "myExampleLedger",
      "S3ExportConfiguration": {
        "EncryptionConfiguration": {
          "ObjectEncryptionType": "SSE_S3"
        },
        "Bucket": "awsExampleBucket",
        "Prefix": "ledgerexport1/"
      },
      "RoleArn": "arn:aws:iam::123456789012:role/my-s3-export-role",
      "ExportCreationTime": 1568847801.418,
      "ExportId": "ADR20NPKN5LINYGb4dp7yZ",
      "InclusiveStartTime": 1568764800.0,
      "ExclusiveEndTime": 1568847599.0
    },
    {
      "Status": "COMPLETED",
      "LedgerName": "myExampleLedger2",
      "S3ExportConfiguration": {
        "EncryptionConfiguration": {
          "ObjectEncryptionType": "SSE_S3"
        },
        "Bucket": "awsExampleBucket",
        "Prefix": "ledgerexport1/"
      },
      "RoleArn": "arn:aws:iam::123456789012:role/my-s3-export-role",
      "ExportCreationTime": 1568846847.638,
      "ExportId": "2pdvW8UQrjBAiYTMehEJDI",
      "InclusiveStartTime": 1568592000.0,
      "ExclusiveEndTime": 1568764800.0
    }
  ]
}

```

Para obtener más información, consulte [Exportación de su diario en Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [ListJournalS3Exports](#) en la referencia de AWS CLI comandos.

## list-ledgers

En el siguiente ejemplo de código, se muestra cómo usar `list-ledgers`.

### AWS CLI

Creación de una lista de los libros mayores disponibles

En el siguiente `list-ledgers` ejemplo, se enumeran todos los libros de contabilidad asociados a la AWS cuenta corriente y a la región.

```
aws qlldb list-ledgers
```

Salida:

```
{
  "Ledgers": [
    {
      "State": "ACTIVE",
      "CreationDateTime": 1568839243.951,
      "Name": "myExampleLedger"
    },
    {
      "State": "ACTIVE",
      "CreationDateTime": 1568839543.557,
      "Name": "myExampleLedger2"
    }
  ]
}
```

Para obtener más información, consulte [Operaciones básicas de Amazon QLDB Ledgers en la Guía para QLDB](#) desarrolladores de Amazon.

- Para API obtener más información, consulte [ListLedgers](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas adjuntas a un libro mayor



En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas adjuntas al libro mayor especificado.

```
aws qlldb list-tags-for-resource \  
  --resource-arn arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger
```

Salida:

```
{  
  "Tags": {  
    "IsTest": "true",  
    "Domain": "Test"  
  }  
}
```

Para obtener más información, consulte [Etiquetado de QLDB los recursos de Amazon](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## stream-journal-to-kinesis

En el siguiente ejemplo de código, se muestra cómo usar `stream-journal-to-kinesis`.

AWS CLI

Ejemplo 1: Para transmitir datos del diario a Kinesis Data Streams mediante archivos de entrada

El siguiente `stream-journal-to-kinesis` ejemplo crea un flujo de datos del diario dentro de un intervalo de fechas y horas especificado a partir de un libro mayor con ese nombre. `myExampleLedger` La transmisión envía los datos a una transmisión de datos de Amazon Kinesis específica.

```
aws qlldb stream-journal-to-kinesis \  
  --ledger-name myExampleLedger \  
  --inclusive-start-time 2020-05-29T00:00:00Z \  
  --exclusive-end-time 2020-05-29T23:59:59Z \  
  --role-arn arn:aws:iam::123456789012:role/my-kinesis-stream-role \  
  --kinesis-configuration file://my-kinesis-config.json \  
  --kinesis-stream-name my-kinesis-stream
```

```
--stream-name myExampleLedger-stream
```

Contenidos de `my-kinesis-config.json`:

```
{
  "StreamArn": "arn:aws:kinesis:us-east-1:123456789012:stream/stream-for-qldb",
  "AggregationEnabled": true
}
```

Salida:

```
{
  "StreamId": "7ISCKqwe4y25YyHLzYUFAf"
}
```

Para obtener más información, consulte [Transmitir datos de revistas de Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

Ejemplo 2: Para transmitir los datos del diario a Kinesis Data Streams mediante una sintaxis abreviada

En el siguiente `stream-journal-to-kinesis` ejemplo, se crea un flujo de datos del diario dentro de un intervalo de fechas y horas especificado a partir de un libro mayor con ese nombre. `myExampleLedger` La transmisión envía los datos a una transmisión de datos de Amazon Kinesis específica.

```
aws qldb stream-journal-to-kinesis \
  --ledger-name myExampleLedger \
  --inclusive-start-time 2020-05-29T00:00:00Z \
  --exclusive-end-time 2020-05-29T23:59:59Z \
  --role-arn arn:aws:iam::123456789012:role/my-kinesis-stream-role \
  --stream-name myExampleLedger-stream \
  --kinesis-configuration StreamArn=arn:aws:kinesis:us-east-1:123456789012:stream/stream-for-qldb,AggregationEnabled=true
```

Salida:

```
{
  "StreamId": "7ISCKqwe4y25YyHLzYUFAf"
}
```

Para obtener más información, consulte [Transmitir datos de revistas de Amazon QLDB](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [StreamJournalToKinesis](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para etiquetar un libro mayor

En el siguiente `tag-resource` ejemplo, se agrega un conjunto de etiquetas a un libro mayor especificado.

```
aws qldb tag-resource \  
  --resource-arn arn:aws:qldb:us-west-2:123456789012:ledger/myExampleLedger \  
  --tags IsTest=true,Domain=Test
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de QLDB los recursos de Amazon](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Cómo eliminar etiquetas de un recurso

En el siguiente `untag-resource` ejemplo, se eliminan las etiquetas con las claves de etiquetas especificadas de un libro mayor especificado.

```
aws qldb untag-resource \  
  --resource-arn arn:aws:qldb:us-west-2:123456789012:ledger/myExampleLedger \  
  --tag-keys IsTest,Domain
```

```
--resource-arn arn:aws:qldb:us-west-2:123456789012:ledger/myExampleLedger \  
--tag-keys IsTest Domain
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de QLDB los recursos de Amazon](#) en la Guía para QLDB desarrolladores de Amazon.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-ledger-permissions-mode

En el siguiente ejemplo de código, se muestra cómo usar `update-ledger-permissions-mode`.

AWS CLI

Ejemplo 1: Para actualizar el modo de permisos de un libro mayor a STANDARD

El siguiente `update-ledger-permissions-mode` ejemplo asigna el modo de STANDARD permisos al libro mayor especificado.

```
aws qldb update-ledger-permissions-mode \  
  --name myExampleLedger \  
  --permissions-mode STANDARD
```

Salida:

```
{  
  "Name": "myExampleLedger",  
  "Arn": "arn:aws:qldb:us-west-2:123456789012:ledger/myExampleLedger",  
  "PermissionsMode": "STANDARD"  
}
```

Ejemplo 2: Para actualizar el modo de permisos de un libro mayor a `_ALLOW ALL`

El siguiente `update-ledger-permissions-mode` ejemplo asigna el modo de `ALLOW_ALL` permisos al libro mayor especificado.

```
aws qldb update-ledger-permissions-mode \  
  --name myExampleLedger \  
  --permissions-mode _ALLOW ALL
```

```
--name myExampleLedger \  
--permissions-mode ALLOW_ALL
```

Salida:

```
{  
  "Name": "myExampleLedger",  
  "Arn": "arn:aws:qldb:us-west-2:123456789012:ledger/myExampleLedger",  
  "PermissionsMode": "ALLOW_ALL"  
}
```

Para obtener más información, consulte [Operaciones básicas de Amazon QLDB Ledgers en la Guía para QLDB](#) desarrolladores de Amazon.

- Para API obtener más información, consulte [UpdateLedgerPermissionsModel](#) la Referencia de AWS CLI comandos.

## update-ledger

En el siguiente ejemplo de código, se muestra cómo usar `update-ledger`.

AWS CLI

Ejemplo 1: Para actualizar la propiedad de protección contra la eliminación de un libro mayor

En el siguiente `update-ledger` ejemplo, se actualiza el libro mayor especificado para deshabilitar la función de protección contra la eliminación.

```
aws qldb update-ledger \  
  --name myExampleLedger \  
  --no-deletion-protection
```

Salida:

```
{  
  "CreationDateTime": 1568839243.951,  
  "Arn": "arn:aws:qldb:us-west-2:123456789012:ledger/myExampleLedger",  
  "DeletionProtection": false,  
  "Name": "myExampleLedger",  
  "State": "ACTIVE"
```

```
}
```

Ejemplo 2: Para actualizar la AWS KMS clave de un libro mayor a una clave gestionada por el cliente

En el siguiente `update-ledger` ejemplo, se actualiza el libro mayor especificado para utilizar una KMS clave gestionada por el cliente para el cifrado en reposo.

```
aws qlldb update-ledger \  
  --name myExampleLedger \  
  --kms-key arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "CreationDateTime": 1568839243.951,  
  "Arn": "arn:aws:qlldb:us-west-2:123456789012:ledger/myExampleLedger",  
  "DeletionProtection": false,  
  "Name": "myExampleLedger",  
  "State": "ACTIVE",  
  "EncryptionDescription": {  
    "KmsKeyArn": "arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "EncryptionStatus": "UPDATING"  
  }  
}
```

Ejemplo 3: Para actualizar la AWS KMS clave de un libro mayor a una AWS clave propia

En el siguiente `update-ledger` ejemplo, se actualiza el libro mayor especificado para usar una KMS clave AWS propia para el cifrado en reposo.

```
aws qlldb update-ledger \  
  --name myExampleLedger \  
  --kms-key AWS_OWNED_KMS_KEY
```

Salida:

```
{  
  "CreationDateTime": 1568839243.951,
```

```
"Arn": "arn:aws:qldb:us-west-2:123456789012:ledger/myExampleLedger",
"DeletionProtection": false,
"Name": "myExampleLedger",
"State": "ACTIVE",
"EncryptionDescription": {
  "KmsKeyArn": "AWS_OWNED_KMS_KEY",
  "EncryptionStatus": "UPDATING"
}
}
```

Para obtener más información, consulte [Operaciones básicas de Amazon QLDB Ledgers en la Guía para QLDB](#) desarrolladores de Amazon.

- Para API obtener más información, consulte [UpdateLedger](#) la Referencia de AWS CLI comandos.

## RDSEjemplos de Amazon que utilizan AWS CLI

En los siguientes ejemplos de código, se muestra cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con AmazonRDS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **add-option-to-option-group**

En el siguiente ejemplo de código, se muestra cómo usar add-option-to-option-group.

AWS CLI

Para añadir una opción a un grupo de opciones

En el siguiente `add-option-to-option-group` ejemplo, se agrega una opción al grupo de opciones especificado.

```
aws rds add-option-to-option-group \
  --option-group-name myoptiongroup \
  --options OptionName=OEM,Port=5500,DBSecurityGroupMemberships=default \
  --apply-immediately
```

Salida:

```
{
  "OptionGroup": {
    "OptionGroupName": "myoptiongroup",
    "OptionGroupDescription": "Test Option Group",
    "EngineName": "oracle-ee",
    "MajorEngineVersion": "12.1",
    "Options": [
      {
        "OptionName": "Timezone",
        "OptionDescription": "Change time zone",
        "Persistent": true,
        "Permanent": false,
        "OptionSettings": [
          {
            "Name": "TIME_ZONE",
            "Value": "Australia/Sydney",
            "DefaultValue": "UTC",
            "Description": "Specifies the timezone the user wants to
change the system time to",
            "ApplyType": "DYNAMIC",
            "DataType": "STRING",
            "AllowedValues": "Africa/Cairo,Africa/Casablanca,Africa/
Harare,Africa/Lagos,Africa/Luanda,Africa/Monrovia,Africa/Nairobi,Africa/
Tripoli,Africa/Windhoek,America/Araguaina,America/Argentina/Buenos_Aires,America/
Asuncion,America/Bogota,America/Caracas,America/Chicago,America/Chihuahua,America/
Cuiaba,America/Denver,America/Detroit,America/Fortaleza,America/Godthab,America/
Guatemala,America/Halifax,America/Lima,America/Los_Angeles,America/Manaus,America/
Matamoros,America/Mexico_City,America/Monterrey,America/Montevideo,America/
New_York,America/Phoenix,America/Santiago,America/Sao_Paulo,America/Tijuana,America/
Toronto,Asia/Amman,Asia/Ashgabat,Asia/Baghdad,Asia/Baku,Asia/Bangkok,Asia/
Beirut,Asia/Calcutta,Asia/Damascus,Asia/Dhaka,Asia/Hong_Kong,Asia/Irkutsk,Asia/
Jakarta,Asia/Jerusalem,Asia/Kabul,Asia/Karachi,Asia/Kathmandu,Asia/Kolkata,Asia/
Krasnoyarsk,Asia/Magadan,Asia/Manila,Asia/Muscat,Asia/Novosibirsk,Asia/Rangoon,Asia/
```



```

Riyadh,Asia/Seoul,Asia/Shanghai,Asia/Singapore,Asia/Taipei,Asia/Tehran,Asia/
Tokyo,Asia/Ulaanbaatar,Asia/Vladivostok,Asia/Yakutsk,Asia/Yerevan,Atlantic/
Azores,Atlantic/Cape_Verde,Australia/Adelaide,Australia/Brisbane,Australia/
Darwin,Australia/Eucla,Australia/Hobart,Australia/Lord_Howe,Australia/
Perth,Australia/Sydney,Brazil/DeNoronha,Brazil/East,Canada/Newfoundland,Canada/
Saskatchewan,Etc/GMT-3,Europe/Amsterdam,Europe/Athens,Europe/Berlin,Europe/
Dublin,Europe/Helsinki,Europe/Kaliningrad,Europe/London,Europe/Madrid,Europe/
Moscow,Europe/Paris,Europe/Prague,Europe/Rome,Europe/Sarajevo,Pacific/Apia,Pacific/
Auckland,Pacific/Chatham,Pacific/Fiji,Pacific/Guam,Pacific/Honolulu,Pacific/
Kiritimati,Pacific/Marquesas,Pacific/Samoa,Pacific/Tongatapu,Pacific/Wake,US/
Alaska,US/Central,US/East-Indiana,US/Eastern,US/Pacific,UTC",
        "IsModifiable": true,
        "IsCollection": false
    }
],
"DBSecurityGroupMemberships": [],
"VpcSecurityGroupMemberships": []
},
{
    "OptionName": "OEM",
    "OptionDescription": "Oracle 12c EM Express",
    "Persistent": false,
    "Permanent": false,
    "Port": 5500,
    "OptionSettings": [],
    "DBSecurityGroupMemberships": [
        {
            "DBSecurityGroupName": "default",
            "Status": "authorized"
        }
    ],
    "VpcSecurityGroupMemberships": []
}
],
"AllowsVpcAndNonVpcInstanceMemberships": false,
"OptionGroupArn": "arn:aws:rds:us-east-1:123456789012:og:myoptiongroup"
}
}

```

Para obtener más información, consulta [Cómo añadir una opción a un grupo de opciones](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [AddOptionToOptionGroup](#) la Referencia de AWS CLI comandos.

## add-role-to-db-cluster

En el siguiente ejemplo de código, se muestra cómo usar `add-role-to-db-cluster`.

### AWS CLI

Para asociar una función de AWS Identity and Access Management (IAM) a un clúster de base de datos

El siguiente `add-role-to-db-cluster` ejemplo asocia un rol a un clúster de base de datos.

```
aws rds add-role-to-db-cluster \  
  --db-cluster-identifier mydbcluster \  
  --role-arn arn:aws:iam::123456789012:role/RDSLoadFromS3
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Asociación de un IAM rol a un clúster de Amazon Aurora My SQL DB](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [AddRoleToDbCluster](#) la Referencia de AWS CLI comandos.

## add-role-to-db-instance

En el siguiente ejemplo de código, se muestra cómo usar `add-role-to-db-instance`.

### AWS CLI

Para asociar una función de AWS Identity and Access Management (IAM) a una instancia de base de datos

En el siguiente `add-role-to-db-instance` ejemplo, se agrega el rol a una instancia de base de datos de Oracle denominada `test-instance`.

```
aws rds add-role-to-db-instance \  
  --db-instance-identifier test-instance \  
  --feature-name S3_INTEGRATION \  
  --role-arn arn:aws:iam::111122223333:role/rds-s3-integration-role
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Requisitos previos para la integración de Amazon RDS Oracle con Amazon S3](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [AddRoleToDbInstance](#) la Referencia de AWS CLI comandos.

## add-source-identifier-to-subscription

En el siguiente ejemplo de código, se muestra cómo usar `add-source-identifier-to-subscription`.

### AWS CLI

Para añadir un identificador de origen a una suscripción

En el siguiente `add-source-identifier` ejemplo, se agrega otro identificador de origen a una suscripción existente.

```
aws rds add-source-identifier-to-subscription \  
  --subscription-name my-instance-events \  
  --source-identifier test-instance-repl
```

Salida:

```
{  
  "EventSubscription": {  
    "SubscriptionCreationTime": "Tue Jul 31 23:22:01 UTC 2018",  
    "CustSubscriptionId": "my-instance-events",  
    "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-  
events",  
    "Enabled": false,  
    "Status": "modifying",  
    "EventCategoriesList": [  
      "backup",  
      "recovery"  
    ],  
    "CustomerAwsId": "123456789012",  
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",  
    "SourceType": "db-instance",  
    "SourceIdsList": [  
      "test-instance",  
      "test-instance-repl"  
    ]  
  }  
}
```

```

    ]
  }
}

```

- Para API obtener más información, consulte [AddSourceIdentifierToSubscription](#) la Referencia de AWS CLI comandos.

## add-tags-to-resource

En el siguiente ejemplo de código, se muestra cómo usar `add-tags-to-resource`.

### AWS CLI

Para añadir etiquetas a un recurso

En el siguiente `add-tags-to-resource` ejemplo, se agregan etiquetas a una RDS base de datos.

```

aws rds add-tags-to-resource \
  --resource-name arn:aws:rds:us-east-1:123456789012:db:database-mysql \
  --tags "[{\"Key\": \"Name\", \"Value\": \"MyDatabase\"}, {\"Key\": \"Environment\", \"Value\": \"test\"}]"

```

Este comando no genera ninguna salida.

Para obtener más información, consulta Cómo [etiquetar RDS los recursos de Amazon](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [AddTagsToResource](#) la Referencia de AWS CLI comandos.

## apply-pending-maintenance-action

En el siguiente ejemplo de código, se muestra cómo usar `apply-pending-maintenance-action`.

### AWS CLI

Para aplicar acciones de mantenimiento pendientes

El siguiente `apply-pending-maintenance-action` ejemplo aplica las acciones de mantenimiento pendientes de un clúster de base de datos.

```
aws rds apply-pending-maintenance-action \
  --resource-identifier arn:aws:rds:us-east-1:123456789012:cluster:my-db-cluster \
  --apply-action system-update \
  --opt-in-type immediate
```

Salida:

```
{
  "ResourcePendingMaintenanceActions": {
    "ResourceIdentifier": "arn:aws:rds:us-east-1:123456789012:cluster:my-db-cluster",
    "PendingMaintenanceActionDetails": [
      {
        "Action": "system-update",
        "OptInStatus": "immediate",
        "CurrentApplyDate": "2021-01-23T01:07:36.100Z",
        "Description": "Upgrade to Aurora PostgreSQL 3.3.2"
      }
    ]
  }
}
```

Para obtener más información, consulte [Mantenimiento de una instancia](#) de base de datos en la Guía del RDS usuario de Amazon y [Mantenimiento de un clúster de base de datos de Amazon Aurora](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ApplyPendingMaintenanceAction](#) la Referencia de AWS CLI comandos.

## authorize-db-security-group-ingress

En el siguiente ejemplo de código, se muestra cómo usar `authorize-db-security-group-ingress`.

### AWS CLI

Para asociar una función de AWS Identity and Access Management (IAM) a una instancia de base de datos

El siguiente `authorize-db-security-group-ingress` ejemplo configura el grupo de seguridad predeterminado con una regla de entrada para el rango de CIDR IP 192.0.2.0/24.

```
aws rds authorize-db-security-group-ingress \  
  --db-security-group-name default \  
  --cidrip 192.0.2.0/24
```

Salida:

```
{  
  "DBSecurityGroup": {  
    "OwnerId": "123456789012",  
    "DBSecurityGroupName": "default",  
    "DBSecurityGroupDescription": "default",  
    "EC2SecurityGroups": [],  
    "IPRanges": [  
      {  
        "Status": "authorizing",  
        "CIDRIP": "192.0.2.0/24"  
      }  
    ],  
    "DBSecurityGroupArn": "arn:aws:rds:us-east-1:111122223333:secgrp:default"  
  }  
}
```

Para obtener más información, consulte [Autorizar el acceso a la red a un grupo de seguridad de base de datos desde un rango de IP](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [AuthorizeDbSecurityGroupIngress](#) la Referencia de AWS CLI comandos.

## backtrack-db-cluster

En el siguiente ejemplo de código, se muestra cómo usar `backtrack-db-cluster`.

### AWS CLI

Para dar marcha atrás a un clúster de base de datos Aurora

En el siguiente `backtrack-db-cluster` ejemplo, se retrocede el clúster de base de datos de muestra especificado hasta el 19 de marzo de 2018, a las 10 a. m.

```
aws rds backtrack-db-cluster --db-cluster-identifier sample-cluster --backtrack-  
to 2018-03-19T10:00:00+00:00
```

Este comando genera un JSON bloque que reconoce el cambio en el recurso. RDS

- Para API obtener más información, consulte [BacktrackDbCluster](#) la Referencia de AWS CLI comandos.

## cancel-export-task

En el siguiente ejemplo de código, se muestra cómo usar `cancel-export-task`.

### AWS CLI

Para cancelar la exportación de una instantánea a Amazon S3

El siguiente `cancel-export-task` ejemplo cancela una tarea de exportación en curso que exporta una instantánea a Amazon S3.

```
aws rds cancel-export-task \  
  --export-task-identifier my-s3-export-1
```

Salida:

```
{  
  "ExportTaskIdentifier": "my-s3-export-1",  
  "SourceArn": "arn:aws:rds:us-east-1:123456789012:snapshot:publisher-final-  
snapshot",  
  "SnapshotTime": "2019-03-24T20:01:09.815Z",  
  "S3Bucket": "mybucket",  
  "S3Prefix": "",  
  "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/export-snap-S3-role",  
  "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/abcd0000-7bfd-4594-af38-  
aabbccddeeff",  
  "Status": "CANCELING",  
  "PercentProgress": 0,  
  "TotalExtractedDataInGB": 0  
}
```

Para obtener más información, consulte [Cancelar una tarea de exportación de instantáneas](#) en la Guía del RDS usuario de Amazon o [Cancelar una tarea de exportación de instantáneas](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte la Referencia [CancelExportTask](#) de AWS CLI comandos.

## copy-db-cluster-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `copy-db-cluster-parameter-group`.

### AWS CLI

Para copiar un grupo de parámetros de un clúster de base de datos

El siguiente `copy-db-cluster-parameter-group` ejemplo hace una copia de un grupo de parámetros de un clúster de base de datos.

```
aws rds copy-db-cluster-parameter-group \  
  --source-db-cluster-parameter-group-identifier mydbclusterpg \  
  --target-db-cluster-parameter-group-identifier mydbclusterpgcopy \  
  --target-db-cluster-parameter-group-description "Copy of mydbclusterpg parameter  
group"
```

Salida:

```
{  
  "DBClusterParameterGroup": {  
    "DBClusterParameterGroupName": "mydbclusterpgcopy",  
    "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:123456789012:cluster-  
pg:mydbclusterpgcopy",  
    "DBParameterGroupFamily": "aurora-mysql5.7",  
    "Description": "Copy of mydbclusterpg parameter group"  
  }  
}
```

Para obtener más información, consulte [Copiar un grupo de parámetros de un clúster](#) de base de datos en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [CopyDbClusterParameterGroup](#) la Referencia de AWS CLI comandos.

## copy-db-cluster-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `copy-db-cluster-snapshot`.

### AWS CLI

Para copiar una instantánea de un clúster de base de datos



El siguiente `copy-db-cluster-snapshot` ejemplo crea una copia de una instantánea de un clúster de base de datos, incluidas sus etiquetas.

```
aws rds copy-db-cluster-snapshot \  
  --source-db-cluster-snapshot-identifier arn:aws:rds:us-east-1:123456789012:cluster-snapshot:rds:myaurora-2019-06-04-09-16 \  
  --target-db-cluster-snapshot-identifier myclustersnapshotcopy \  
  --copy-tags
```

Salida:

```
{  
  "DBClusterSnapshot": {  
    "AvailabilityZones": [  
      "us-east-1a",  
      "us-east-1b",  
      "us-east-1e"  
    ],  
    "DBClusterSnapshotIdentifier": "myclustersnapshotcopy",  
    "DBClusterIdentifier": "myaurora",  
    "SnapshotCreateTime": "2019-06-04T09:16:42.649Z",  
    "Engine": "aurora-mysql",  
    "AllocatedStorage": 0,  
    "Status": "available",  
    "Port": 0,  
    "VpcId": "vpc-6594f31c",  
    "ClusterCreateTime": "2019-04-15T14:18:42.785Z",  
    "MasterUsername": "myadmin",  
    "EngineVersion": "5.7.mysql_aurora.2.04.2",  
    "LicenseModel": "aurora-mysql",  
    "SnapshotType": "manual",  
    "PercentProgress": 100,  
    "StorageEncrypted": true,  
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",  
    "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:123456789012:cluster-snapshot:myclustersnapshotcopy",  
    "IAMDatabaseAuthenticationEnabled": false  
  }  
}
```

Para obtener más información, consulte [Copiar una instantánea](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [CopyDbClusterSnapshot](#) la Referencia de AWS CLI comandos.

## copy-db-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `copy-db-parameter-group`.

### AWS CLI

Para copiar un grupo de parámetros de un clúster de base de datos

El siguiente `copy-db-parameter-group` ejemplo hace una copia de un grupo de parámetros de base de datos.

```
aws rds copy-db-parameter-group \  
  --source-db-parameter-group-identifier mydbpg \  
  --target-db-parameter-group-identifier mydbpgcopy \  
  --target-db-parameter-group-description "Copy of mydbpg parameter group"
```

Salida:

```
{  
  "DBParameterGroup": {  
    "DBParameterGroupName": "mydbpgcopy",  
    "DBParameterGroupArn": "arn:aws:rds:us-east-1:814387698303:pg:mydbpgcopy",  
    "DBParameterGroupFamily": "mysql5.7",  
    "Description": "Copy of mydbpg parameter group"  
  }  
}
```

Para obtener más información, consulte [Copiar un grupo de parámetros de base](#) de datos en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [CopyDbParameterGroup](#) la Referencia de AWS CLI comandos.

## copy-db-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `copy-db-snapshot`.

## AWS CLI

Para copiar una instantánea de base de datos

El siguiente copy-db-snapshot ejemplo crea una copia de una instantánea de base de datos.

```
aws rds copy-db-snapshot \  
  --source-db-snapshot-identifier rds:database-mysql-2019-06-06-08-38 \  
  --target-db-snapshot-identifier mydbsnapshotcopy
```

Salida:

```
{  
  "DBSnapshot": {  
    "VpcId": "vpc-6594f31c",  
    "Status": "creating",  
    "Encrypted": true,  
    "SourceDBSnapshotIdentifier": "arn:aws:rds:us-  
east-1:123456789012:snapshot:rds:database-mysql-2019-06-06-08-38",  
    "MasterUsername": "admin",  
    "Iops": 1000,  
    "Port": 3306,  
    "LicenseModel": "general-public-license",  
    "DBSnapshotArn": "arn:aws:rds:us-  
east-1:123456789012:snapshot:mydbsnapshotcopy",  
    "EngineVersion": "5.6.40",  
    "OptionGroupName": "default:mysql-5-6",  
    "ProcessorFeatures": [],  
    "Engine": "mysql",  
    "StorageType": "io1",  
    "DbiResourceId": "db-ZI7UJ5BLKMBYFGX7FDENCKADC4",  
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",  
    "SnapshotType": "manual",  
    "IAMDatabaseAuthenticationEnabled": false,  
    "SourceRegion": "us-east-1",  
    "DBInstanceIdentifier": "database-mysql",  
    "InstanceCreateTime": "2019-04-30T15:45:53.663Z",  
    "AvailabilityZone": "us-east-1f",  
    "PercentProgress": 0,  
    "AllocatedStorage": 100,  
    "DBSnapshotIdentifier": "mydbsnapshotcopy"  
  }  
}
```

Para obtener más información, consulta Cómo [copiar una instantánea](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [CopyDbSnapshot](#) la Referencia de AWS CLI comandos.

## copy-option-group

En el siguiente ejemplo de código, se muestra cómo usar `copy-option-group`.

### AWS CLI

Para copiar un grupo de opciones

En el siguiente `copy-option-group` ejemplo, se hace una copia de un grupo de opciones.

```
aws rds copy-option-group \  
  --source-option-group-identifier myoptiongroup \  
  --target-option-group-identifier new-option-group \  
  --target-option-group-description "My option group copy"
```

Salida:

```
{  
  "OptionGroup": {  
    "Options": [],  
    "OptionGroupName": "new-option-group",  
    "MajorEngineVersion": "11.2",  
    "OptionGroupDescription": "My option group copy",  
    "AllowsVpcAndNonVpcInstanceMemberships": true,  
    "EngineName": "oracle-ee",  
    "OptionGroupArn": "arn:aws:rds:us-east-1:123456789012:og:new-option-group"  
  }  
}
```

Para obtener más información, consulte Cómo [hacer una copia de un grupo de opciones](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [CopyOptionGroup](#) la Referencia de AWS CLI comandos.

## create-blue-green-deployment

En el siguiente ejemplo de código, se muestra cómo usar `create-blue-green-deployment`.

### AWS CLI

Ejemplo 1: Para crear una implementación azul/verde RDS para una instancia de My DB SQL

El siguiente `create-blue-green-deployment` ejemplo crea una implementación azul/verde para una instancia de My DB. SQL

```
aws rds create-blue-green-deployment \  
  --blue-green-deployment-name bgd-cli-test-instance \  
  --source arn:aws:rds:us-east-1:123456789012:db:my-db-instance \  
  --target-engine-version 8.0 \  
  --target-db-parameter-group-name mysql-80-group
```

Salida:

```
{  
  "BlueGreenDeployment": {  
    "BlueGreenDeploymentIdentifier": "bgd-v53303651eexfake",  
    "BlueGreenDeploymentName": "bgd-cli-test-instance",  
    "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",  
    "SwitchoverDetails": [  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-  
instance"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-  
instance-replica-1"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-  
instance-replica-2"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-  
instance-replica-3"  
      }  
    ],  
    "Tasks": [  

```

```

    {
      "Name": "CREATING_READ_REPLICA_OF_SOURCE",
      "Status": "PENDING"
    },
    {
      "Name": "DB_ENGINE_VERSION_UPGRADE",
      "Status": "PENDING"
    },
    {
      "Name": "CONFIGURE_BACKUPS",
      "Status": "PENDING"
    },
    {
      "Name": "CREATING_TOPOLOGY_OF_SOURCE",
      "Status": "PENDING"
    }
  ],
  "Status": "PROVISIONING",
  "CreateTime": "2022-02-25T21:18:51.183000+00:00"
}

```

Para obtener más información, consulta [Cómo crear una implementación azul/verde](#) en la Guía RDS del usuario de Amazon.

Ejemplo 2: Para crear una implementación azul/verde para un clúster Aurora My DB SQL

El siguiente `create-blue-green-deployment` ejemplo crea una implementación azul/verde para un clúster Aurora My SQL DB.

```

aws rds create-blue-green-deployment \
  --blue-green-deployment-name my-blue-green-deployment \
  --source arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster \
  --target-engine-version 8.0 \
  --target-db-cluster-parameter-group-name ams-80-binlog-enabled \
  --target-db-parameter-group-name mysql-80-cluster-group

```

Salida:

```

{
  "BlueGreenDeployment": {
    "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
    "BlueGreenDeploymentName": "my-blue-green-deployment",

```

```
    "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-
cluster",
    "SwitchoverDetails": [
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-
mysql-cluster",
        "Status": "PROVISIONING"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-
cluster-1",
        "Status": "PROVISIONING"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-
cluster-2",
        "Status": "PROVISIONING"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-
cluster-3",
        "Status": "PROVISIONING"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-
excluded-member-endpoint",
        "Status": "PROVISIONING"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-
reader-endpoint",
        "Status": "PROVISIONING"
      }
    ],
    "Tasks": [
      {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "PENDING"
      },
      {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "PENDING"
      }
    ]
  }
```

```

        "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
        "Status": "PENDING"
    },
    {
        "Name": "CREATE_CUSTOM_ENDPOINTS",
        "Status": "PENDING"
    }
],
"Status": "PROVISIONING",
"CreateTime": "2022-02-25T21:12:00.288000+00:00"
}
}

```

Para obtener más información, consulte [Creación de una implementación azul/verde](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte la Referencia [CreateBlueGreenDeployment](#) de AWS CLI comandos.

## create-db-cluster-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `create-db-cluster-endpoint`.

### AWS CLI

Para crear un punto final de clúster de base de datos personalizado

El siguiente `create-db-cluster-endpoint` ejemplo crea un punto final de clúster de base de datos personalizado y lo asocia al clúster de base de datos Aurora especificado.

```

aws rds create-db-cluster-endpoint \
  --db-cluster-endpoint-identifier mycustomendpoint \
  --endpoint-type reader \
  --db-cluster-identifier mydbcluster \
  --static-members dbinstance1 dbinstance2

```

Salida:

```

{
  "DBClusterEndpointIdentifier": "mycustomendpoint",
  "DBClusterIdentifier": "mydbcluster",
  "DBClusterEndpointResourceIdentifier": "cluster-endpoint-ANPAJ4AE5446DAEXAMPLE",

```



```

    "Endpoint": "mycustomendpoint.cluster-custom-cnpxample.us-
east-1.rds.amazonaws.com",
    "Status": "creating",
    "EndpointType": "CUSTOM",
    "CustomEndpointType": "READER",
    "StaticMembers": [
        "dbinstance1",
        "dbinstance2"
    ],
    "ExcludedMembers": [],
    "DBClusterEndpointArn": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:mycustomendpoint"
}

```

Para obtener más información, consulte [Amazon Aurora Connection Management](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [CreateDbClusterEndpoint](#) la Referencia de AWS CLI comandos.

## create-db-cluster-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `create-db-cluster-parameter-group`.

### AWS CLI

Para crear un grupo de parámetros de clúster de base de datos

El siguiente `create-db-cluster-parameter-group` ejemplo crea un grupo de parámetros de clúster de base de datos.

```

aws rds create-db-cluster-parameter-group \
  --db-cluster-parameter-group-name mydbclusterparametergroup \
  --db-parameter-group-family aurora5.6 \
  --description "My new cluster parameter group"

```

Salida:

```

{
  "DBClusterParameterGroup": {
    "DBClusterParameterGroupName": "mydbclusterparametergroup",

```

```

    "DBParameterGroupFamily": "aurora5.6",
    "Description": "My new cluster parameter group",
    "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:123456789012:cluster-
pg:mydbclusterparametergroup"
  }
}

```

Para obtener más información, consulte [Creación de un grupo de parámetros de clúster](#) de base de datos en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [CreateDbClusterParameterGroup](#) la Referencia de AWS CLI comandos.

## create-db-cluster-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `create-db-cluster-snapshot`.

### AWS CLI

Para crear una instantánea de un clúster de base de datos

El siguiente `create-db-cluster-snapshot` ejemplo crea una instantánea de un clúster de base de datos.

```

aws rds create-db-cluster-snapshot \
  --db-cluster-identifier mydbcluster \
  --db-cluster-snapshot-identifier mydbclustersnapshot

```

Salida:

```

{
  "DBClusterSnapshot": {
    "AvailabilityZones": [
      "us-east-1a",
      "us-east-1b",
      "us-east-1e"
    ],
    "DBClusterSnapshotIdentifier": "mydbclustersnapshot",
    "DBClusterIdentifier": "mydbcluster",
    "SnapshotCreateTime": "2019-06-18T21:21:00.469Z",
    "Engine": "aurora-mysql",
    "AllocatedStorage": 1,
  }
}

```

```
"Status": "creating",
"Port": 0,
"VpcId": "vpc-6594f31c",
"ClusterCreateTime": "2019-04-15T14:18:42.785Z",
"MasterUsername": "myadmin",
"EngineVersion": "5.7.mysql_aurora.2.04.2",
"LicenseModel": "aurora-mysql",
"SnapshotType": "manual",
"PercentProgress": 0,
"StorageEncrypted": true,
"KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
"DBClusterSnapshotArn": "arn:aws:rds:us-east-1:123456789012:cluster-
snapshot:mydbclustersnapshot",
"IAMDatabaseAuthenticationEnabled": false
}
}
```

Para obtener más información, consulte [Creación de una instantánea de clúster](#) de base de datos en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [CreateDbClusterSnapshot](#) la Referencia de AWS CLI comandos.

## create-db-cluster

En el siguiente ejemplo de código, se muestra cómo usar `create-db-cluster`.

### AWS CLI

Ejemplo 1: Para crear un clúster de base de datos compatible con My SQL 5.7

En el siguiente `create-db-cluster` ejemplo, se crea un clúster de base de datos SQL compatible con My 5.7 con la versión de motor predeterminada. Sustituya la contraseña de ejemplo por `secret99` una contraseña segura. Cuando utiliza la consola para crear un clúster de base de datos, Amazon crea RDS automáticamente la instancia de base de datos de escritura para su clúster de base de datos. Sin embargo, cuando utilice la AWS CLI para crear un clúster de base de datos, debe crear de forma explícita la instancia de base de datos de escritura para su clúster de base de datos mediante el `create-db-instance` AWS CLI comando.

```
aws rds create-db-cluster \  
  --db-cluster-identifier sample-cluster \  
  --engine aurora-mysql \  
  --master-username myadmin \  
  --master-user-password secret99 \  
  --vpc-subnet-ids vpc-6594f31c \  
  --storage-encrypted
```

```
--engine-version 5.7 \  
--master-username admin \  
--master-user-password secret99 \  
--db-subnet-group-name default \  
--vpc-security-group-ids sg-0b9130572daf3dc16
```

Salida:

```
{  
  "DBCluster": {  
    "DBSubnetGroup": "default",  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-0b9130572daf3dc16",  
        "Status": "active"  
      }  
    ],  
    "AllocatedStorage": 1,  
    "AssociatedRoles": [],  
    "PreferredBackupWindow": "09:12-09:42",  
    "ClusterCreateTime": "2023-02-27T23:21:33.048Z",  
    "DeletionProtection": false,  
    "IAMDatabaseAuthenticationEnabled": false,  
    "ReadReplicaIdentifiers": [],  
    "EngineMode": "provisioned",  
    "Engine": "aurora-mysql",  
    "StorageEncrypted": false,  
    "MultiAZ": false,  
    "PreferredMaintenanceWindow": "mon:04:31-mon:05:01",  
    "HttpEndpointEnabled": false,  
    "BackupRetentionPeriod": 1,  
    "DbClusterResourceId": "cluster-ANPAJ4AE5446DAEXAMPLE",  
    "DBClusterIdentifier": "sample-cluster",  
    "AvailabilityZones": [  
      "us-east-1a",  
      "us-east-1b",  
      "us-east-1e"  
    ],  
    "MasterUsername": "master",  
    "EngineVersion": "5.7.mysql_aurora.2.11.1",  
    "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:sample-cluster",  
    "DBClusterMembers": [],  
    "Port": 3306,  
  }  
}
```

```

    "Status": "creating",
    "Endpoint": "sample-cluster.cluster-cnpexample.us-east-1.rds.amazonaws.com",
    "DBClusterParameterGroup": "default.aurora-mysql5.7",
    "HostedZoneId": "Z2R2ITUGPM61AM",
    "ReaderEndpoint": "sample-cluster.cluster-ro-cnpexample.us-
east-1.rds.amazonaws.com",
    "CopyTagsToSnapshot": false
  }
}

```

Ejemplo 2: Para crear un clúster de base de datos compatible con Postgre SQL --

El siguiente `create-db-cluster` ejemplo crea un clúster de base de datos SQL compatible con Postgre utilizando la versión de motor predeterminada. Sustituya la contraseña de ejemplo por una contraseña segura `secret99`. Cuando utiliza la consola para crear un clúster de base de datos, Amazon crea RDS automáticamente la instancia de base de datos de escritura para su clúster de base de datos. Sin embargo, cuando utilice la AWS CLI para crear un clúster de base de datos, debe crear de forma explícita la instancia de base de datos de escritura para su clúster de base de datos mediante el `create-db-instance` AWS CLI comando.

```

aws rds create-db-cluster \
  --db-cluster-identifier sample-pg-cluster \
  --engine aurora-postgresql \
  --master-username master \
  --master-user-password secret99 \
  --db-subnet-group-name default \
  --vpc-security-group-ids sg-0b9130572daf3dc16

```

Salida:

```

{
  "DBCluster": {
    "Endpoint": "sample-pg-cluster.cluster-cnpexample.us-
east-1.rds.amazonaws.com",
    "HttpEndpointEnabled": false,
    "DBClusterMembers": [],
    "EngineMode": "provisioned",
    "CopyTagsToSnapshot": false,
    "HostedZoneId": "Z2R2ITUGPM61AM",
    "IAMDatabaseAuthenticationEnabled": false,
    "AllocatedStorage": 1,

```

```

    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-0b9130572daf3dc16",
        "Status": "active"
      }
    ],
    "DeletionProtection": false,
    "StorageEncrypted": false,
    "BackupRetentionPeriod": 1,
    "PreferredBackupWindow": "09:56-10:26",
    "ClusterCreateTime": "2023-02-27T23:26:08.371Z",
    "DBClusterParameterGroup": "default.aurora-postgresql13",
    "EngineVersion": "13.7",
    "Engine": "aurora-postgresql",
    "Status": "creating",
    "DBClusterIdentifier": "sample-pg-cluster",
    "MultiAZ": false,
    "Port": 5432,
    "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:sample-pg-
cluster",
    "AssociatedRoles": [],
    "DbClusterResourceId": "cluster-ANPAJ4AE5446DAEXAMPLE",
    "PreferredMaintenanceWindow": "wed:03:33-wed:04:03",
    "ReaderEndpoint": "sample-pg-cluster.cluster-ro-cnpxexample.us-
east-1.rds.amazonaws.com",
    "MasterUsername": "master",
    "AvailabilityZones": [
      "us-east-1a",
      "us-east-1b",
      "us-east-1c"
    ],
    "ReadReplicaIdentifiers": [],
    "DBSubnetGroup": "default"
  }
}

```

Para obtener más información, consulte [Creación de un clúster de base de datos de Amazon Aurora](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [CreateDbCluster](#) la Referencia de AWS CLI comandos.

## create-db-instance-read-replica

En el siguiente ejemplo de código, se muestra cómo usar `create-db-instance-read-replica`.

### AWS CLI

Para crear una instancia de base de datos, lea réplica

En este ejemplo, se crea una réplica de lectura de una instancia de base de datos existente denominada `test-instance`. La réplica de lectura recibe un nombre `test-instance-repl`.

```
aws rds create-db-instance-read-replica \  
  --db-instance-identifier test-instance-repl \  
  --source-db-instance-identifier test-instance
```

Salida:

```
{  
  "DBInstance": {  
    "IAMDatabaseAuthenticationEnabled": false,  
    "MonitoringInterval": 0,  
    "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance-repl",  
    "ReadReplicaSourceDBInstanceIdentifier": "test-instance",  
    "DBInstanceIdentifier": "test-instance-repl",  
    ...some output truncated...  
  }  
}
```

- Para API obtener más información, consulte [CreateDbInstanceReadReplica](#) la Referencia de AWS CLI comandos.

## create-db-instance

En el siguiente ejemplo de código, se muestra cómo usar `create-db-instance`.

### AWS CLI

Creación de una instancia de base de datos

En el siguiente ejemplo `create-db-instance`, se utilizan las opciones necesarias para lanzar una nueva instancia de base de datos.

```
aws rds create-db-instance \  
  --db-instance-identifier test-mysql-instance \  
  --db-instance-class db.t3.micro \  
  --engine mysql \  
  --master-username admin \  
  --master-user-password secret99 \  
  --allocated-storage 20
```

Salida:

```
{  
  "DBInstance": {  
    "DBInstanceIdentifier": "test-mysql-instance",  
    "DBInstanceClass": "db.t3.micro",  
    "Engine": "mysql",  
    "DBInstanceStatus": "creating",  
    "MasterUsername": "admin",  
    "AllocatedStorage": 20,  
    "PreferredBackupWindow": "12:55-13:25",  
    "BackupRetentionPeriod": 1,  
    "DBSecurityGroups": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-12345abc",  
        "Status": "active"  
      }  
    ],  
    "DBParameterGroups": [  
      {  
        "DBParameterGroupName": "default.mysql5.7",  
        "ParameterApplyStatus": "in-sync"  
      }  
    ],  
    "DBSubnetGroup": {  
      "DBSubnetGroupName": "default",  
      "DBSubnetGroupDescription": "default",  
      "VpcId": "vpc-2ff2ff2f",  
      "SubnetGroupStatus": "Complete",  
      "Subnets": [  
        {  
          "SubnetIdentifier": "subnet-#####",  
          "SubnetAvailabilityZone": {  
            "Name": "us-west-2c"  
          }  
        }  
      ]  
    }  
  }  
}
```



```
    },
    "SubnetStatus": "Active"
  },
  {
    "SubnetIdentifier": "subnet-#####",
    "SubnetAvailabilityZone": {
      "Name": "us-west-2d"
    },
    "SubnetStatus": "Active"
  },
  {
    "SubnetIdentifier": "subnet-#####",
    "SubnetAvailabilityZone": {
      "Name": "us-west-2a"
    },
    "SubnetStatus": "Active"
  },
  {
    "SubnetIdentifier": "subnet-#####",
    "SubnetAvailabilityZone": {
      "Name": "us-west-2b"
    },
    "SubnetStatus": "Active"
  }
]
},
"PreferredMaintenanceWindow": "sun:08:07-sun:08:37",
"PendingModifiedValues": {
  "MasterUserPassword": "*****"
},
"MultiAZ": false,
"EngineVersion": "5.7.22",
"AutoMinorVersionUpgrade": true,
"ReadReplicaDBInstanceIdentifiers": [],
"LicenseModel": "general-public-license",
"OptionGroupMemberships": [
  {
    "OptionGroupName": "default:mysql-5-7",
    "Status": "in-sync"
  }
],
"PubliclyAccessible": true,
"StorageType": "gp2",
"DbInstancePort": 0,
```

```

    "StorageEncrypted": false,
    "DbiResourceId": "db-5555EXAMPLE44444444EXAMPLE",
    "CACertificateIdentifier": "rds-ca-2019",
    "DomainMemberships": [],
    "CopyTagsToSnapshot": false,
    "MonitoringInterval": 0,
    "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:test-mysql-
instance",
    "IAMDatabaseAuthenticationEnabled": false,
    "PerformanceInsightsEnabled": false,
    "DeletionProtection": false,
    "AssociatedRoles": []
  }
}

```

Para obtener más información, consulte [Creación de una RDS instancia](#) de base de datos de Amazon en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [C createDBInstance](#) en la referencia de AWS CLI comandos.

## create-db-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `create-db-parameter-group`.

### AWS CLI

Creación de un grupo de parámetros de base de datos

En el siguiente ejemplo `create-db-parameter-group`, se crea un grupo de parámetros de base de datos.

```

aws rds create-db-parameter-group \
  --db-parameter-group-name mydbparametergroup \
  --db-parameter-group-family MySQL5.6 \
  --description "My new parameter group"

```

Salida:

```

{
  "DBParameterGroup": {
    "DBParameterGroupName": "mydbparametergroup",

```

```

    "DBParameterGroupFamily": "mysql5.6",
    "Description": "My new parameter group",
    "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:mydbparametergroup"
  }
}

```

Para obtener más información, consulte [Creación de un grupo de parámetros de base de datos](#) de datos en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [reateDBParameterGrupo C](#) en la referencia de AWS CLI comandos.

## create-db-proxy-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `create-db-proxy-endpoint`.

### AWS CLI

Para crear un punto final proxy de base de datos para una RDS base de datos

El siguiente `create-db-proxy-endpoint` ejemplo crea un punto final de proxy de base de datos.

```

aws rds create-db-proxy-endpoint \
  --db-proxy-name proxyExample \
  --db-proxy-endpoint-name "proxyep1" \
  --vpc-subnet-ids subnetgroup1 subnetgroup2

```

Salida:

```

{
  "DBProxyEndpoint": {
    "DBProxyEndpointName": "proxyep1",
    "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-
endpoint:prx-endpoint-0123a01b12345c0ab",
    "DBProxyName": "proxyExample",
    "Status": "creating",
    "VpcId": "vpc-1234567",
    "VpcSecurityGroupIds": [
      "sg-1234",
      "sg-5678"
    ]
  }
}

```

```

    ],
    "VpcSubnetIds": [
        "subnetgroup1",
        "subnetgroup2"
    ],
    "Endpoint": "proxyep1.endpoint.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
    "CreateDate": "2023-04-05T16:09:33.452000+00:00",
    "TargetRole": "READ_WRITE",
    "IsDefault": false
  }
}

```

Para obtener más información, consulte [Creación de un punto de enlace proxy](#) en la Guía del RDS usuario de Amazon y [Creación de un punto de enlace proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [CreateDbProxyEndpoint](#) la Referencia de AWS CLI comandos.

## create-db-proxy

En el siguiente ejemplo de código, se muestra cómo usar `create-db-proxy`.

### AWS CLI

Para crear un proxy de base de datos para una RDS base de datos

El siguiente `create-db-proxy` ejemplo crea un proxy de base de datos.

```

aws rds create-db-proxy \
  --db-proxy-name proxyExample \
  --engine-family MYSQL \
  --auth
  Description="proxydescription1",AuthScheme="SECRETS",SecretArn="arn:aws:secretsmanager:us-
west-2:123456789123:secret:secretName-1234f",IAMAuth="DISABLED",ClientPasswordAuthType="MYSQ
  \
  --role-arn arn:aws:iam::123456789123:role/ProxyRole \
  --vpc-subnet-ids subnetgroup1 subnetgroup2

```

Salida:

```
{
```

```

"DBProxy": {
  "DBProxyName": "proxyExample",
  "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-
proxy:prx-0123a01b12345c0ab",
  "EngineFamily": "MYSQL",
  "VpcId": "vpc-1234567",
  "VpcSecuritytGroupIds": [
    "sg-1234",
    "sg-5678",
    "sg-9101"
  ],
  "VpcSubnetIds": [
    "subnetgroup1",
    "subnetgroup2"
  ],
  "Auth": "[
    {
      "Description": "proxydescription1",
      "AuthScheme": "SECRETS",
      "SecretArn": "arn:aws:secretsmanager:us-
west-2:123456789123:secret:proxyssecret1-Abcd1e",
      "IAMAuth": "DISABLED"
    }
  ]",
  "RoleArn": "arn:aws:iam::12345678912:role/ProxyRole",
  "Endpoint": "proxyExample.proxy-ab0cd1efghij.us-east-1.rds.amazonaws.com",
  "RequireTLS": false,
  "IdleClientTimeout": 1800,
  "DebuggingLogging": false,
  "CreateDate": "2023-04-05T16:09:33.452000+00:00",
  "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"
}
}

```

Para obtener más información, consulte [Creación de un RDS proxy](#) en la Guía del RDS usuario de Amazon y [Creación de un RDS proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [CreateDbProxy](#) la Referencia de AWS CLI comandos.

## create-db-security-group

En el siguiente ejemplo de código, se muestra cómo usar create-db-security-group.

## AWS CLI

Para crear un grupo de seguridad de Amazon RDS DB

El siguiente `create-db-security-group` comando crea un nuevo grupo de seguridad de Amazon RDS DB:

```
aws rds create-db-security-group --db-security-group-name mysecgroup --db-security-group-description "My Test Security Group"
```

En el ejemplo, el nuevo grupo de seguridad de base de datos recibe un nombre `mysecgroup` y tiene una descripción.

Salida:

```
{
  "DBSecurityGroup": {
    "OwnerId": "123456789012",
    "DBSecurityGroupName": "mysecgroup",
    "DBSecurityGroupDescription": "My Test Security Group",
    "VpcId": "vpc-a1b2c3d4",
    "EC2SecurityGroups": [],
    "IPRanges": [],
    "DBSecurityGroupArn": "arn:aws:rds:us-west-2:123456789012:secgrp:mysecgroup"
  }
}
```

- Para API obtener más información, consulte [CreateDbSecurityGroup](#) la Referencia de AWS CLI comandos.

## `create-db-shard-group`

En el siguiente ejemplo de código, se muestra cómo usar `create-db-shard-group`.

## AWS CLI

Ejemplo 1: Para crear un clúster de base de datos SQL principal de Aurora Postgre

El siguiente `create-db-cluster` ejemplo crea un clúster de base de datos SQL principal de Aurora Postgre que es compatible con Aurora Serverless v2 y Aurora Limitless Database.

```
aws rds create-db-cluster \  
  --db-cluster-identifier my-sv2-cluster \  
  --engine aurora-postgresql \  
  --engine-version 15.2-limitless \  
  --storage-type aurora-iopt1 \  
  --serverless-v2-scaling-configuration MinCapacity=2,MaxCapacity=16 \  
  --enable-limitless-database \  
  --master-username myuser \  
  --master-user-password mypassword \  
  --enable-cloudwatch-logs-exports postgresql
```

Salida:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 1,  
    "AvailabilityZones": [  
      "us-east-2b",  
      "us-east-2c",  
      "us-east-2a"  
    ],  
    "BackupRetentionPeriod": 1,  
    "DBClusterIdentifier": "my-sv2-cluster",  
    "DBClusterParameterGroup": "default.aurora-postgresql15",  
    "DBSubnetGroup": "default",  
    "Status": "creating",  
    "Endpoint": "my-sv2-cluster.cluster-cekyceexample.us-  
east-2.rds.amazonaws.com",  
    "ReaderEndpoint": "my-sv2-cluster.cluster-ro-cekyceexample.us-  
east-2.rds.amazonaws.com",  
    "MultiAZ": false,  
    "Engine": "aurora-postgresql",  
    "EngineVersion": "15.2-limitless",  
    "Port": 5432,  
    "MasterUsername": "myuser",  
    "PreferredBackupWindow": "06:05-06:35",  
    "PreferredMaintenanceWindow": "mon:08:25-mon:08:55",  
    "ReadReplicaIdentifiers": [],  
    "DBClusterMembers": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-#####",  
        "Status": "active"  
      }  
    ]  
  }  
}
```

```

    }
  ],
  "HostedZoneId": "Z2XHWR1EXAMPLE",
  "StorageEncrypted": false,
  "DbClusterResourceId": "cluster-XYEDT6ML6FHIXH4Q2J1EXAMPLE",
  "DBClusterArn": "arn:aws:rds:us-east-2:123456789012:cluster:my-sv2-cluster",
  "AssociatedRoles": [],
  "IAMDatabaseAuthenticationEnabled": false,
  "ClusterCreateTime": "2024-02-19T16:24:07.771000+00:00",
  "EnabledCloudwatchLogsExports": [
    "postgresql"
  ],
  "EngineMode": "provisioned",
  "DeletionProtection": false,
  "HttpEndpointEnabled": false,
  "CopyTagsToSnapshot": false,
  "CrossAccountClone": false,
  "DomainMemberships": [],
  "TagList": [],
  "StorageType": "aurora-iopt1",
  "AutoMinorVersionUpgrade": true,
  "ServerlessV2ScalingConfiguration": {
    "MinCapacity": 2.0,
    "MaxCapacity": 16.0
  },
  "NetworkType": "IPV4",
  "IOOptimizedNextAllowedModificationTime":
"2024-03-21T16:24:07.781000+00:00",
  "LimitlessDatabase": {
    "Status": "not-in-use",
    "MinRequiredACU": 96.0
  }
}
}
}

```

Ejemplo 2: Para crear la instancia de base de datos principal (de escritura)

En el siguiente `create-db-instance` ejemplo, se crea una instancia de base de datos principal (de escritura) de Aurora Serverless v2. Cuando utiliza la consola para crear un clúster de base de datos, Amazon crea RDS automáticamente la instancia de base de datos de escritura para su clúster de base de datos. Sin embargo, cuando utilice la AWS CLI para crear un clúster de base de datos, debe crear de forma explícita la instancia de base de datos de escritura para su clúster de base de datos mediante el `create-db-instance` AWS CLI comando.



```
aws rds create-db-instance \  
  --db-instance-identifier my-sv2-instance \  
  --db-cluster-identifier my-sv2-cluster \  
  --engine aurora-postgresql \  
  --db-instance-class db.serverless
```

Salida:

```
{  
  "DBInstance": {  
    "DBInstanceIdentifier": "my-sv2-instance",  
    "DBInstanceClass": "db.serverless",  
    "Engine": "aurora-postgresql",  
    "DBInstanceStatus": "creating",  
    "MasterUsername": "myuser",  
    "AllocatedStorage": 1,  
    "PreferredBackupWindow": "06:05-06:35",  
    "BackupRetentionPeriod": 1,  
    "DBSecurityGroups": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-#####",  
        "Status": "active"  
      }  
    ],  
    "DBParameterGroups": [  
      {  
        "DBParameterGroupName": "default.aurora-postgresql15",  
        "ParameterApplyStatus": "in-sync"  
      }  
    ],  
    "DBSubnetGroup": {  
      "DBSubnetGroupName": "default",  
      "DBSubnetGroupDescription": "default",  
      "VpcId": "vpc-#####",  
      "SubnetGroupStatus": "Complete",  
      "Subnets": [  
        {  
          "SubnetIdentifier": "subnet-#####",  
          "SubnetAvailabilityZone": {  
            "Name": "us-east-2c"  
          }  
        },  
        {  
          "SubnetIdentifier": "subnet-#####",  
          "SubnetAvailabilityZone": {  
            "Name": "us-east-2c"  
          }  
        }  
      ],  
      "SubnetOutpost": {}  
    }  
  }  
}
```

```

        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-#####",
        "SubnetAvailabilityZone": {
            "Name": "us-east-2a"
        },
        "SubnetOutpost": {},
        "SubnetStatus": "Active"
    },
    {
        "SubnetIdentifier": "subnet-#####",
        "SubnetAvailabilityZone": {
            "Name": "us-east-2b"
        },
        "SubnetOutpost": {},
        "SubnetStatus": "Active"
    }
]
},
"PreferredMaintenanceWindow": "fri:09:01-fri:09:31",
"PendingModifiedValues": {
    "PendingCloudwatchLogsExports": {
        "LogTypesToEnable": [
            "postgresql"
        ]
    }
},
"MultiAZ": false,
"EngineVersion": "15.2-limitless",
"AutoMinorVersionUpgrade": true,
"ReadReplicaDBInstanceIdentifiers": [],
"LicenseModel": "postgresql-license",
"OptionGroupMemberships": [
    {
        "OptionGroupName": "default:aurora-postgresql-15",
        "Status": "in-sync"
    }
],
"PubliclyAccessible": false,
"StorageType": "aurora-iopt1",
"DbInstancePort": 0,
"DBClusterIdentifier": "my-sv2-cluster",
"StorageEncrypted": false,

```

```

    "DbiResourceId": "db-BIQTE3B3K3RM7M74SK5EXAMPLE",
    "CACertificateIdentifier": "rds-ca-rsa2048-g1",
    "DomainMemberships": [],
    "CopyTagsToSnapshot": false,
    "MonitoringInterval": 0,
    "PromotionTier": 1,
    "DBInstanceArn": "arn:aws:rds:us-east-2:123456789012:db:my-sv2-instance",
    "IAMDatabaseAuthenticationEnabled": false,
    "PerformanceInsightsEnabled": false,
    "DeletionProtection": false,
    "AssociatedRoles": [],
    "TagList": [],
    "CustomerOwnedIpEnabled": false,
    "BackupTarget": "region",
    "NetworkType": "IPV4",
    "StorageThroughput": 0,
    "CertificateDetails": {
      "CAIdentifier": "rds-ca-rsa2048-g1"
    },
    "DedicatedLogVolume": false
  }
}

```

Ejemplo 3: Para crear el grupo de fragmentos de base de datos

En el siguiente `create-db-shard-group` ejemplo, se crea un grupo de fragmentos de base de datos en el clúster de base de datos SQL principal de Aurora Postgre.

```

aws rds create-db-shard-group \
  --db-shard-group-identifier my-db-shard-group \
  --db-cluster-identifier my-sv2-cluster \
  --max-acu 768

```

Salida:

```

{
  "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
  "DBShardGroupIdentifier": "my-db-shard-group",
  "DBClusterIdentifier": "my-sv2-cluster",
  "MaxACU": 768.0,
  "ComputeRedundancy": 0,
  "Status": "creating",
  "PubliclyAccessible": false,
}

```

```
"Endpoint": "my-sv2-cluster.limitless-cekyceexample.us-east-2.rds.amazonaws.com"
}
```

Para obtener más información, consulte [Uso de Aurora Serverless v2](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [CreateDbShardGroup](#) la Referencia de AWS CLI comandos.

## create-db-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `create-db-snapshot`.

### AWS CLI

Creación de una instantánea de base de datos

En el siguiente ejemplo `create-db-snapshot`, se crea una instantánea de base de datos.

```
aws rds create-db-snapshot \  
  --db-instance-identifier database-mysql \  
  --db-snapshot-identifier mydbsnapshot
```

Salida:

```
{  
  "DBSnapshot": {  
    "DBSnapshotIdentifier": "mydbsnapshot",  
    "DBInstanceIdentifier": "database-mysql",  
    "Engine": "mysql",  
    "AllocatedStorage": 100,  
    "Status": "creating",  
    "Port": 3306,  
    "AvailabilityZone": "us-east-1b",  
    "VpcId": "vpc-6594f31c",  
    "InstanceCreateTime": "2019-04-30T15:45:53.663Z",  
    "MasterUsername": "admin",  
    "EngineVersion": "5.6.40",  
    "LicenseModel": "general-public-license",  
    "SnapshotType": "manual",  
    "Iops": 1000,  
    "OptionGroupName": "default:mysql-5-6",  
    "PercentProgress": 0,  
  }  
}
```

```

    "StorageType": "io1",
    "Encrypted": true,
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
    "DBSnapshotArn": "arn:aws:rds:us-east-1:123456789012:snapshot:mydbsnapshot",
    "IAMDatabaseAuthenticationEnabled": false,
    "ProcessorFeatures": [],
    "DbiResourceId": "db-AKIAIOSFODNN7EXAMPLE"
  }
}

```

Para obtener más información, consulte [Creación de una instantánea de base](#) de datos en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [C reateDBSnapshot](#) en la referencia de AWS CLI comandos.

## create-db-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar create-db-subnet-group.

### AWS CLI

Para crear un grupo de subredes de base de datos

En el siguiente create-db-subnet-group ejemplo, se crea un grupo de subredes de base de datos denominado mysubnetgroup mediante subredes existentes.

```

aws rds create-db-subnet-group \
  --db-subnet-group-name mysubnetgroup \
  --db-subnet-group-description "test DB subnet group" \
  --subnet-ids
  ["subnet-0a1dc4e1a6f123456","subnet-070dd7ecb3aaaaaaa","subnet-00f5b198bc0abcdef"]

```

Salida:

```

{
  "DBSubnetGroup": {
    "DBSubnetGroupName": "mysubnetgroup",
    "DBSubnetGroupDescription": "test DB subnet group",
    "VpcId": "vpc-0f08e7610a1b2c3d4",
    "SubnetGroupStatus": "Complete",
    "Subnets": [

```

```
{
  "SubnetIdentifier": "subnet-070dd7ecb3aaaaaaa",
  "SubnetAvailabilityZone": {
    "Name": "us-west-2b"
  },
  "SubnetStatus": "Active"
},
{
  "SubnetIdentifier": "subnet-00f5b198bc0abcdef",
  "SubnetAvailabilityZone": {
    "Name": "us-west-2d"
  },
  "SubnetStatus": "Active"
},
{
  "SubnetIdentifier": "subnet-0a1dc4e1a6f123456",
  "SubnetAvailabilityZone": {
    "Name": "us-west-2b"
  },
  "SubnetStatus": "Active"
}
],
"DBSubnetGroupArn": "arn:aws:rds:us-
west-2:0123456789012:subgrp:mysubnetgroup"
}
```

Para obtener más información, consulte [Creación de una instancia](#) de base de datos VPC en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [CreateDbSubnetGroup](#) la Referencia de AWS CLI comandos.

## **create-event-subscription**

En el siguiente ejemplo de código, se muestra cómo usar `create-event-subscription`.

### AWS CLI

Para crear una suscripción a un evento

En el siguiente `create-event-subscription` ejemplo, se crea una suscripción para los eventos de copia de seguridad y recuperación de las instancias de base de datos de la AWS

cuenta corriente. Las notificaciones se envían a un tema de Amazon Simple Notification Service, especificado por `--sns-topic-arn`.

```
aws rds create-event-subscription \  
  --subscription-name my-instance-events \  
  --source-type db-instance \  
  --event-categories '["backup","recovery"]' \  
  --sns-topic-arn arn:aws:sns:us-east-1:123456789012:interesting-events
```

Salida:

```
{  
  "EventSubscription": {  
    "Status": "creating",  
    "CustSubscriptionId": "my-instance-events",  
    "SubscriptionCreationTime": "Tue Jul 31 23:22:01 UTC 2018",  
    "EventCategoriesList": [  
      "backup",  
      "recovery"  
    ],  
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",  
    "CustomerAwsId": "123456789012",  
    "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-  
events",  
    "SourceType": "db-instance",  
    "Enabled": true  
  }  
}
```

- Para API obtener más información, consulte [CreateEventSubscription](#) la Referencia de AWS CLI comandos.

## **create-global-cluster**

En el siguiente ejemplo de código, se muestra cómo usar `create-global-cluster`.

### AWS CLI

Para crear un clúster de base de datos global

El siguiente `create-global-cluster` ejemplo crea un nuevo clúster de base SQL de datos global compatible con Aurora My.

```
aws rds create-global-cluster \  
  --global-cluster-identifier myglobalcluster \  
  --engine aurora-mysql
```

Salida:

```
{  
  "GlobalCluster": {  
    "GlobalClusterIdentifier": "myglobalcluster",  
    "GlobalClusterResourceId": "cluster-f0e523bfe07aabb",  
    "GlobalClusterArn": "arn:aws:rds::123456789012:global-  
cluster:myglobalcluster",  
    "Status": "available",  
    "Engine": "aurora-mysql",  
    "EngineVersion": "5.7.mysql_aurora.2.07.2",  
    "StorageEncrypted": false,  
    "DeletionProtection": false,  
    "GlobalClusterMembers": []  
  }  
}
```

Para obtener más información, consulte [Creación de una base de datos global de Aurora](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [CreateGlobalCluster](#) la Referencia de AWS CLI comandos.

## create-option-group

En el siguiente ejemplo de código, se muestra cómo usar create-option-group.

### AWS CLI

Para crear un grupo de RDS opciones de Amazon

El siguiente create-option-group comando crea un nuevo grupo de RDS opciones de Amazon para Oracle Enterprise Edition la versión 11.2`, is named ``MyOptionGroup e incluye una descripción.

```
aws rds create-option-group \  
  --option-group-name MyOptionGroup \  
  --engine-name oracle-ee \  
  --description "MyOptionGroup description"
```



```
--major-engine-version 11.2 \  
--option-group-description "Oracle Database Manager Database Control"
```

Salida:

```
{  
  "OptionGroup": {  
    "OptionGroupName": "myoptiongroup",  
    "OptionGroupDescription": "Oracle Database Manager Database Control",  
    "EngineName": "oracle-ee",  
    "MajorEngineVersion": "11.2",  
    "Options": [],  
    "AllowsVpcAndNonVpcInstanceMemberships": true,  
    "OptionGroupArn": "arn:aws:rds:us-west-2:123456789012:og:myoptiongroup"  
  }  
}
```

- Para API obtener más información, consulte [CreateOptionGroup](#) la Referencia de AWS CLI comandos.

## delete-blue-green-deployment

En el siguiente ejemplo de código, se muestra cómo usar `delete-blue-green-deployment`.

AWS CLI

Ejemplo 1: Para eliminar recursos en un entorno verde RDS para una instancia de My SQL DB

En el siguiente `delete-blue-green-deployment` ejemplo, se eliminan los recursos de un entorno ecológico RDS para una instancia de My SQL DB.

```
aws rds delete-blue-green-deployment \  
  --blue-green-deployment-identifier bgd-v53303651eexfake \  
  --delete-target
```

Salida:

```
{  
  "BlueGreenDeployment": {  
    "BlueGreenDeploymentIdentifier": "bgd-v53303651eexfake",  
    "BlueGreenDeploymentName": "bgd-cli-test-instance",  
    "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",  
  }  
}
```

```
    "Target": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-green-
rkfbpe",
    "SwitchoverDetails": [
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-green-rkfbpe",
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1-green-j382ha",
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2-green-ejv4ao",
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3-green-vlpz3t",
        "Status": "AVAILABLE"
      }
    ],
    "Tasks": [
      {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
      },
      {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
      },
      {
        "Name": "CONFIGURE_BACKUPS",
        "Status": "COMPLETED"
      }
    ]
  }
}
```

```

    },
    {
      "Name": "CREATING_TOPOLOGY_OF_SOURCE",
      "Status": "COMPLETED"
    }
  ],
  "Status": "DELETING",
  "CreateTime": "2022-02-25T21:18:51.183000+00:00",
  "DeleteTime": "2022-02-25T22:25:31.331000+00:00"
}
}

```

Para obtener más información, consulta [Eliminar una implementación azul/verde](#) en la Guía del RDSusuario de Amazon.

Ejemplo 2: Para eliminar recursos en un entorno verde para un clúster Aurora My SQL DB

En el siguiente delete-blue-green-deployment ejemplo, se eliminan los recursos de un entorno verde para un clúster Aurora My SQL DB.

```

aws rds delete-blue-green-deployment \
  --blue-green-deployment-identifier bgd-wi89nwzglccsfake \
  --delete-target

```

Salida:

```

{
  "BlueGreenDeployment": {
    "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
    "BlueGreenDeploymentName": "my-blue-green-deployment",
    "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
    "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3rnukl",
    "SwitchoverDetails": [
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3rnukl",
        "Status": "AVAILABLE"
      }
    ],
  },
  {

```

```
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-1-green-gpmaxf",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-2",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-2-green-j2oajq",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-3",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-3-green-mkxies",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint-green-4sqjrq",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint-green-gwzlg",
        "Status": "AVAILABLE"
    }
],
"Tasks": [
    {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
    },
    {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
    }
],
```

```
{
  "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
  "Status": "COMPLETED"
},
{
  "Name": "CREATE_CUSTOM_ENDPOINTS",
  "Status": "COMPLETED"
}
],
"Status": "DELETING",
"CreateTime": "2022-02-25T21:12:00.288000+00:00",
>DeleteTime": "2022-02-25T22:29:11.336000+00:00"
}
}
```

Para obtener más información, consulte [Eliminar una implementación azul/verde](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte la Referencia [DeleteBlueGreenDeployment](#) de AWS CLI comandos.

## delete-db-cluster-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-cluster-endpoint`.

### AWS CLI

Para eliminar un punto final de clúster de base de datos personalizado

En el siguiente `delete-db-cluster-endpoint` ejemplo, se elimina el punto final del clúster de base de datos personalizado especificado.

```
aws rds delete-db-cluster-endpoint \
  --db-cluster-endpoint-identifier mycustomendpoint
```

Salida:

```
{
  "DBClusterEndpointIdentifier": "mycustomendpoint",
  "DBClusterIdentifier": "mydbcluster",
  "DBClusterEndpointResourceIdentifier": "cluster-endpoint-ANPAJ4AE5446DAEXAMPLE",
```

```

    "Endpoint": "mycustomendpoint.cluster-custom-cnpxample.us-
east-1.rds.amazonaws.com",
    "Status": "deleting",
    "EndpointType": "CUSTOM",
    "CustomEndpointType": "READER",
    "StaticMembers": [
        "dbinstance1",
        "dbinstance2",
        "dbinstance3"
    ],
    "ExcludedMembers": [],
    "DBClusterEndpointArn": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:mycustomendpoint"
}

```

Para obtener más información, consulte [Amazon Aurora Connection Management](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DeleteDbClusterEndpoint](#) la Referencia de AWS CLI comandos.

## delete-db-cluster-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-cluster-parameter-group`.

### AWS CLI

Para eliminar un grupo de parámetros de un clúster de base de datos

En el siguiente `delete-db-cluster-parameter-group` ejemplo, se elimina el grupo de parámetros del clúster de base de datos especificado.

```

aws rds delete-db-cluster-parameter-group \
  --db-cluster-parameter-group-name mydbclusterparametergroup

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajo con grupos de parámetros de bases de datos y grupos de parámetros de clústeres de bases de datos](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DeleteDbClusterParameterGroup](#) la Referencia de AWS CLI comandos.

## delete-db-cluster-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-cluster-snapshot`.

### AWS CLI

Para eliminar una instantánea de un clúster de base de datos

El siguiente `delete-db-cluster-snapshot` ejemplo elimina la instantánea del clúster de base de datos especificada.

```
aws rds delete-db-cluster-snapshot \  
--db-cluster-snapshot-identifier mydbclustersnapshot
```

Salida:

```
{  
  "DBClusterSnapshot": {  
    "AvailabilityZones": [  
      "us-east-1a",  
      "us-east-1b",  
      "us-east-1e"  
    ],  
    "DBClusterSnapshotIdentifier": "mydbclustersnapshot",  
    "DBClusterIdentifier": "mydbcluster",  
    "SnapshotCreateTime": "2019-06-18T21:21:00.469Z",  
    "Engine": "aurora-mysql",  
    "AllocatedStorage": 0,  
    "Status": "available",  
    "Port": 0,  
    "VpcId": "vpc-6594f31c",  
    "ClusterCreateTime": "2019-04-15T14:18:42.785Z",  
    "MasterUsername": "myadmin",  
    "EngineVersion": "5.7.mysql_aurora.2.04.2",  
    "LicenseModel": "aurora-mysql",  
    "SnapshotType": "manual",  
    "PercentProgress": 100,  
    "StorageEncrypted": true,  
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",  
    "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:123456789012:cluster-  
snapshot:mydbclustersnapshot",  
    "IAMDatabaseAuthenticationEnabled": false  
  }  
}
```

```
}
```

Para obtener más información, consulte [Eliminar una instantánea](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DeleteDbClusterSnapshot](#) la Referencia de AWS CLI comandos.

## delete-db-cluster

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-cluster`.

### AWS CLI

Ejemplo 1: Para eliminar una instancia de base de datos en un clúster de base de datos

En el siguiente `delete-db-instance` ejemplo, se elimina la última instancia de base de datos de un clúster de base de datos. No puede eliminar un clúster de base de datos si contiene instancias de base de datos que no están en el estado de eliminación. No puede tomar una instantánea final al eliminar una instancia de base de datos de un clúster de base de datos.

```
aws rds delete-db-instance \  
  --db-instance-identifier database-3
```

Salida:

```
{  
  "DBInstance": {  
    "DBInstanceIdentifier": "database-3",  
    "DBInstanceClass": "db.r4.large",  
    "Engine": "aurora-postgresql",  
    "DBInstanceStatus": "deleting",  
  
    ...output omitted...  
  }  
}
```

Para obtener más información, consulte [Eliminar una instancia de base de datos en un clúster de base de datos Aurora](#) en la Guía del usuario de Amazon Aurora.



## Ejemplo 2: Para eliminar un clúster de base de datos

En el siguiente `delete-db-cluster` ejemplo, se elimina el clúster de base de datos denominado `mycluster` y se toma una instantánea final con ese nombre `mycluster-final-snapshot`. El estado del clúster de base de datos está disponible mientras se realiza la instantánea. Para seguir el progreso de la eliminación, utilice el `describe-db-clusters` CLI comando.

```
aws rds delete-db-cluster \  
  --db-cluster-identifier mycluster \  
  --no-skip-final-snapshot \  
  --final-db-snapshot-identifier mycluster-final-snapshot
```

Salida:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 20,  
    "AvailabilityZones": [  
      "eu-central-1b",  
      "eu-central-1c",  
      "eu-central-1a"  
    ],  
    "BackupRetentionPeriod": 7,  
    "DBClusterIdentifier": "mycluster",  
    "DBClusterParameterGroup": "default.aurora-postgresql10",  
    "DBSubnetGroup": "default-vpc-aa11bb22",  
    "Status": "available",  
  
    ...output omitted...  
  }  
}
```

Para obtener más información, consulte [Clústeres Aurora con una sola instancia](#) de base de datos en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DeleteDbCluster](#) la Referencia de AWS CLI comandos.

## delete-db-instance-automated-backup

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-instance-automated-backup`.

### AWS CLI

Para eliminar una copia de seguridad automatizada replicada de una región

En el siguiente `delete-db-instance-automated-backup` ejemplo, se elimina la copia de seguridad automática con el nombre de recurso de Amazon (ARN) especificado.

```
aws rds delete-db-instance-automated-backup \
  --db-instance-automated-backups-arn "arn:aws:rds:us-west-2:123456789012:auto-
  backup:ab-jkib2gfq5rv7replzadausbrktni2bn4example"
```

Salida:

```
{
  "DBInstanceAutomatedBackup": {
    "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db",
    "DbiResourceId": "db-JKIB2GFQ5RV7REPLZA4EXAMPLE",
    "Region": "us-east-1",
    "DBInstanceIdentifier": "new-orcl-db",
    "RestoreWindow": {},
    "AllocatedStorage": 20,
    "Status": "deleting",
    "Port": 1521,
    "AvailabilityZone": "us-east-1b",
    "VpcId": "vpc-#####",
    "InstanceCreateTime": "2020-12-04T15:28:31Z",
    "MasterUsername": "admin",
    "Engine": "oracle-se2",
    "EngineVersion": "12.1.0.2.v21",
    "LicenseModel": "bring-your-own-license",
    "OptionGroupName": "default:oracle-se2-12-1",
    "Encrypted": false,
    "StorageType": "gp2",
    "IAMDatabaseAuthenticationEnabled": false,
    "BackupRetentionPeriod": 7,
    "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-west-2:123456789012:auto-
    backup:ab-jkib2gfq5rv7replzadausbrktni2bn4example"
  }
}
```

```
}
```

Para obtener más información, consulta [Eliminar copias de seguridad replicadas](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DeleteDbInstanceAutomatedBackup](#) la Referencia de AWS CLI comandos.

## delete-db-instance

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-instance`.

### AWS CLI

Eliminación de una instancia de base de datos

En el siguiente ejemplo `delete-db-instance`, se elimina la instancia de base de datos especificada tras crear una instantánea de base de datos final denominada `test-instance-final-snap`.

```
aws rds delete-db-instance \  
  --db-instance-identifier test-instance \  
  --final-db-snapshot-identifier test-instance-final-snap
```

Salida:

```
{  
  "DBInstance": {  
    "DBInstanceIdentifier": "test-instance",  
    "DBInstanceStatus": "deleting",  
    ...some output truncated...  
  }  
}
```

- Para API obtener más información, consulte [DeleteDBInstance](#) en la Referencia de AWS CLI comandos.

## delete-db-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-parameter-group`.

## AWS CLI

Eliminación de un grupo de parámetros de base de datos

En el siguiente ejemplo command, se elimina un grupo de parámetros de base de datos.

```
aws rds delete-db-parameter-group \  
  --db-parameter-group-name mydbparametergroup
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con grupos de parámetros de base de datos](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte el [deleteDBParametergrupo D](#) en la referencia de AWS CLI comandos.

## delete-db-proxy-endpoint

En el siguiente ejemplo de código, se muestra cómo usar delete-db-proxy-endpoint.

### AWS CLI

Para eliminar un punto final de proxy de base de datos para una RDS base de datos

En el siguiente delete-db-proxy-endpoint ejemplo, se elimina un punto final de proxy de base de datos de la base de datos de destino.

```
aws rds delete-db-proxy-endpoint \  
  --db-proxy-endpoint-name proxyEP1
```

Salida:

```
{  
  "DBProxyEndpoint":  
    {  
      "DBProxyEndpointName": "proxyEP1",  
      "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-  
endpoint:prx-endpoint-0123a01b12345c0ab",  
      "DBProxyName": "proxyExample",  
      "Status": "deleting",  
      "VpcId": "vpc-1234567",
```

```

    "VpcSecurityGroupIds": [
      "sg-1234",
      "sg-5678"
    ],
    "VpcSubnetIds": [
      "subnetgroup1",
      "subnetgroup2"
    ],
    "Endpoint": "proxyEP1.endpoint.proxy-ab0cd1efghij.us-east-1.rds.amazonaws.com",
    "CreateDate": "2023-04-13T01:49:38.568000+00:00",
    "TargetRole": "READ_ONLY",
    "IsDefault": false
  }
}

```

Para obtener más información, consulte [Eliminar un punto final proxy](#) en la Guía del RDS usuario de Amazon y [Eliminar un punto final proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DeleteDbProxyEndpoint](#) la Referencia de AWS CLI comandos.

## delete-db-proxy

En el siguiente ejemplo de código, se muestra cómo usar delete-db-proxy.

### AWS CLI

Para eliminar un proxy de base de datos para una RDS base de datos

El siguiente delete-db-proxy ejemplo elimina un proxy de base de datos.

```

aws rds delete-db-proxy \
  --db-proxy-name proxyExample

```

Salida:

```

{
  "DBProxy": {
    "DBProxyName": "proxyExample",
    "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-proxy:prx-0123a01b12345c0ab",
  }
}

```

```

    "Status": "deleting",
    "EngineFamily": "PostgreSQL",
    "VpcId": "vpc-1234567",
    "VpcSecurityGroupIds": [
      "sg-1234",
      "sg-5678"
    ],
    "VpcSubnetIds": [
      "subnetgroup1",
      "subnetgroup2"
    ],
    "Auth": "[
      {
        "Description": "proxydescription`",
        "AuthScheme": "SECRETS",
        "SecretArn": "arn:aws:secretsmanager:us-
west-2:123456789123:secret:proxysecret1-Abcd1e",
        "IAMAuth": "DISABLED"
      } ],
    "RoleArn": "arn:aws:iam::12345678912:role/ProxyPostgreSQLRole",
    "Endpoint": "proxyExample.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
    "RequireTLS": false,
    "IdleClientTimeout": 1800,
    "DebuggingLogging": false,
    "CreateDate": "2023-04-05T16:09:33.452000+00:00",
    "UpdateDate": "2023-04-13T01:49:38.568000+00:00"
  }
}

```

Para obtener más información, consulte [Eliminar un RDS proxy](#) en la Guía del RDS usuario de Amazon y [Eliminar un RDS proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DeleteDbProxy](#) la Referencia de AWS CLI comandos.

## delete-db-security-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-security-group`.

### AWS CLI

Para eliminar un grupo de seguridad de base de datos

En el siguiente `delete-db-security-group` ejemplo, se elimina un grupo de seguridad de base de datos denominado `mysecuritygroup`.

```
aws rds delete-db-security-group \  
  --db-security-group-name mysecuritygroup
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con grupos de seguridad de bases EC2 de datos \(plataforma clásica\)](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DeleteDbSecurityGroup](#) la Referencia de AWS CLI comandos.

## `delete-db-shard-group`

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-shard-group`.

### AWS CLI

Ejemplo 1: Eliminar un grupo de fragmentos de base de datos sin éxito

El siguiente `delete-db-shard-group` ejemplo muestra el error que se produce cuando se intenta eliminar un grupo de fragmentos de base de datos antes de eliminar todas las bases de datos y los esquemas.

```
aws rds delete-db-shard-group \  
  --db-shard-group-identifier limitless-test-shard-grp
```

Salida:

```
An error occurred (InvalidDBShardGroupState) when calling the DeleteDBShardGroup  
  operation: Unable to delete the DB shard group limitless-test-db-shard-group.  
  Delete all of your Limitless Database databases and schemas, then try again.
```

Ejemplo 2: Para eliminar correctamente un grupo de particiones de base de datos

En el siguiente `delete-db-shard-group` ejemplo, se elimina un grupo de fragmentos de base de datos después de eliminar todas las bases de datos y los esquemas, incluido el esquema `public`

```
aws rds delete-db-shard-group \  
  --db-shard-group-identifier limitless-test-shard-grp
```

Salida:

```
{  
  "DBShardGroupResourceId": "shardgroup-7bb446329da94788b3f957746example",  
  "DBShardGroupIdentifier": "limitless-test-shard-grp",  
  "DBClusterIdentifier": "limitless-test-cluster",  
  "MaxACU": 768.0,  
  "ComputeRedundancy": 0,  
  "Status": "deleting",  
  "PubliclyAccessible": true,  
  "Endpoint": "limitless-test-cluster.limitless-cekyexample.us-  
east-2.rds.amazonaws.com"  
}
```

Para obtener más información, consulte [Eliminar instancias y clústeres de base de datos Aurora](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DeleteDbShardGroup](#) la Referencia de AWS CLI comandos.

## delete-db-snapshot

En el siguiente ejemplo de código, se muestra cómo usar delete-db-snapshot.

AWS CLI

Para eliminar una instantánea de base de datos

El siguiente delete-db-snapshot ejemplo elimina la instantánea de base de datos especificada.

```
aws rds delete-db-snapshot \  
  --db-snapshot-identifier mydbsnapshot
```

Salida:

```
{  
  "DBSnapshot": {
```



```

    "DBSnapshotIdentifier": "mydbsnapshot",
    "DBInstanceIdentifier": "database-mysql",
    "SnapshotCreateTime": "2019-06-18T22:08:40.702Z",
    "Engine": "mysql",
    "AllocatedStorage": 100,
    "Status": "deleted",
    "Port": 3306,
    "AvailabilityZone": "us-east-1b",
    "VpcId": "vpc-6594f31c",
    "InstanceCreateTime": "2019-04-30T15:45:53.663Z",
    "MasterUsername": "admin",
    "EngineVersion": "5.6.40",
    "LicenseModel": "general-public-license",
    "SnapshotType": "manual",
    "Iops": 1000,
    "OptionGroupName": "default:mysql-5-6",
    "PercentProgress": 100,
    "StorageType": "io1",
    "Encrypted": true,
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
    "DBSnapshotArn": "arn:aws:rds:us-east-1:123456789012:snapshot:mydbsnapshot",
    "IAMDatabaseAuthenticationEnabled": false,
    "ProcessorFeatures": [],
    "DbiResourceId": "db-AKIAIOSFODNN7EXAMPLE"
  }
}

```

Para obtener más información, consulta [Eliminar una instantánea](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DeleteDbSnapshot](#) la Referencia de AWS CLI comandos.

## delete-db-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-db-subnet-group`.

### AWS CLI

Para eliminar un grupo de subredes de base de datos

El siguiente `delete-db-subnet-group` ejemplo elimina el grupo de subredes de base de datos llamado. `mysubnetgroup`

```
aws rds delete-db-subnet-group --db-subnet-group-name mysubnetgroup
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con una instancia](#) de base de datos VPC en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DeleteDbSubnetGroup](#) la Referencia de AWS CLI comandos.

## delete-event-subscription

En el siguiente ejemplo de código, se muestra cómo usar delete-event-subscription.

### AWS CLI

Para eliminar una suscripción a un evento

En el siguiente delete-event-subscription ejemplo, se elimina la suscripción al evento especificada.

```
aws rds delete-event-subscription --subscription-name my-instance-events
```

Salida:

```
{
  "EventSubscription": {
    "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-
events",
    "CustomerAwsId": "123456789012",
    "Enabled": false,
    "SourceIdsList": [
      "test-instance"
    ],
    "SourceType": "db-instance",
    "EventCategoriesList": [
      "backup",
      "recovery"
    ],
    "SubscriptionCreationTime": "2018-07-31 23:22:01.893",
    "CustSubscriptionId": "my-instance-events",
```

```

    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",
    "Status": "deleting"
  }
}

```

- Para API obtener más información, consulte [DeleteEventSubscription](#) la Referencia de AWS CLI comandos.

## delete-global-cluster

En el siguiente ejemplo de código, se muestra cómo usar `delete-global-cluster`.

### AWS CLI

Para eliminar un clúster de base de datos global

En el siguiente `delete-global-cluster` ejemplo, se elimina un clúster de base SQL de datos global compatible con Aurora My. El resultado muestra el clúster que va a eliminar, pero los `describe-global-clusters` comandos posteriores no muestran ese clúster de base de datos.

```

aws rds delete-global-cluster \
  --global-cluster-identifier myglobalcluster

```

Salida:

```

{
  "GlobalCluster": {
    "GlobalClusterIdentifier": "myglobalcluster",
    "GlobalClusterResourceId": "cluster-f0e523bfe07aabb",
    "GlobalClusterArn": "arn:aws:rds::123456789012:global-cluster:myglobalcluster",
    "Status": "available",
    "Engine": "aurora-mysql",
    "EngineVersion": "5.7.mysql_aurora.2.07.2",
    "StorageEncrypted": false,
    "DeletionProtection": false,
    "GlobalClusterMembers": []
  }
}

```

Para obtener más información, consulte [Eliminar una base de datos global de Aurora](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DeleteGlobalCluster](#) la Referencia de AWS CLI comandos.

## **delete-option-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-option-group`.

### AWS CLI

Para eliminar un grupo de opciones

En el siguiente `delete-option-group` ejemplo, se elimina el grupo de opciones especificado.

```
aws rds delete-option-group \  
  --option-group-name myoptiongroup
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Eliminar un grupo de opciones](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DeleteOptionGroup](#) la Referencia de AWS CLI comandos.

## **deregister-db-proxy-targets**

En el siguiente ejemplo de código, se muestra cómo usar `deregister-db-proxy-targets`.

### AWS CLI

Para anular el registro de un destino proxy de base de datos del grupo objetivo de la base de datos

El siguiente `deregister-db-proxy-targets` ejemplo elimina la asociación entre el proxy `proxyExample` y su objetivo.

```
aws rds deregister-db-proxy-targets \  
  --db-proxy-name proxyExample \  
  --db-instance-identifiers database-1
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar un RDS proxy](#) en la Guía del RDS usuario de Amazon y [Eliminar un RDS proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DeregisterDbProxyTargets](#) la Referencia de AWS CLI comandos.

## describe-account-attributes

En el siguiente ejemplo de código, se muestra cómo usar `describe-account-attributes`.

### AWS CLI

Descripción de atributos de cuenta

En el siguiente `describe-account-attributes` ejemplo, se recuperan los atributos de la AWS cuenta actual.

```
aws rds describe-account-attributes
```

Salida:

```
{
  "AccountQuotas": [
    {
      "Max": 40,
      "Used": 4,
      "AccountQuotaName": "DBInstances"
    },
    {
      "Max": 40,
      "Used": 0,
      "AccountQuotaName": "ReservedDBInstances"
    },
    {
      "Max": 100000,
      "Used": 40,
      "AccountQuotaName": "AllocatedStorage"
    },
    {
      "Max": 25,
      "Used": 0,

```

```
    "AccountQuotaName": "DBSecurityGroups"
  },
  {
    "Max": 20,
    "Used": 0,
    "AccountQuotaName": "AuthorizationsPerDBSecurityGroup"
  },
  {
    "Max": 50,
    "Used": 1,
    "AccountQuotaName": "DBParameterGroups"
  },
  {
    "Max": 100,
    "Used": 3,
    "AccountQuotaName": "ManualSnapshots"
  },
  {
    "Max": 20,
    "Used": 0,
    "AccountQuotaName": "EventSubscriptions"
  },
  {
    "Max": 50,
    "Used": 1,
    "AccountQuotaName": "DBSubnetGroups"
  },
  {
    "Max": 20,
    "Used": 1,
    "AccountQuotaName": "OptionGroups"
  },
  {
    "Max": 20,
    "Used": 6,
    "AccountQuotaName": "SubnetsPerDBSubnetGroup"
  },
  {
    "Max": 5,
    "Used": 0,
    "AccountQuotaName": "ReadReplicasPerMaster"
  },
  {
    "Max": 40,
```

```

        "Used": 1,
        "AccountQuotaName": "DBClusters"
    },
    {
        "Max": 50,
        "Used": 0,
        "AccountQuotaName": "DBClusterParameterGroups"
    },
    {
        "Max": 5,
        "Used": 0,
        "AccountQuotaName": "DBClusterRoles"
    }
]
}

```

- Para API obtener más información, consulte [DescribeAccountAttributes](#) la Referencia de AWS CLI comandos.

## describe-blue-green-deployments

En el siguiente ejemplo de código, se muestra cómo usar `describe-blue-green-deployments`.

### AWS CLI

Ejemplo 1: Para describir el despliegue azul/verde de una RDS instancia de base de datos una vez finalizada la creación

En el siguiente `describe-blue-green-deployment` ejemplo, se recuperan los detalles de una implementación azul/verde una vez finalizada la creación.

```
aws rds describe-blue-green-deployments \
  --blue-green-deployment-identifier bgd-v53303651eexfake
```

Salida:

```
{
  "BlueGreenDeployments": [
    {
      "BlueGreenDeploymentIdentifier": "bgd-v53303651eexfake",
      "BlueGreenDeploymentName": "bgd-cli-test-instance",
      "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",

```

```
    "Target": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-green-
rkfbpe",
    "SwitchoverDetails": [
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-green-rkfbpe",
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1-green-j382ha",
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2-green-ejv4ao",
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3-green-vlpz3t",
        "Status": "AVAILABLE"
      }
    ],
    "Tasks": [
      {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
      },
      {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
      },
      {
        "Name": "CONFIGURE_BACKUPS",
        "Status": "COMPLETED"
      }
    ]
  }
}
```



```

        },
        {
            "Name": "CREATING_TOPOLOGY_OF_SOURCE",
            "Status": "COMPLETED"
        }
    ],
    "Status": "AVAILABLE",
    "CreateTime": "2022-02-25T21:18:51.183000+00:00"
}
]
}

```

Para obtener más información, consulta [Visualización de una implementación azul/verde](#) en la Guía RDS del usuario de Amazon.

Ejemplo 2: Para describir una implementación azul/verde para un clúster Aurora My DB SQL

El siguiente `describe-blue-green-deployment` ejemplo recupera los detalles de una implementación azul/verde.

```

aws rds describe-blue-green-deployments \
  --blue-green-deployment-identifier bgd-wi89nwzglccsfake

```

Salida:

```

{
  "BlueGreenDeployments": [
    {
      "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
      "BlueGreenDeploymentName": "my-blue-green-deployment",
      "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
      "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3rnuk1",
      "SwitchoverDetails": [
        {
          "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
          "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3rnuk1",
          "Status": "AVAILABLE"
        }
      ],
    },
  ]
}

```

```
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-1-green-gpmaxf",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-2",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-2-green-j2oajq",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-3",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-3-green-mkxies",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-excluded-member-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-excluded-member-endpoint-green-4sqjrq",
        "Status": "AVAILABLE"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-reader-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-reader-endpoint-green-gwzlg",
        "Status": "AVAILABLE"
    }
],
"Tasks": [
    {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
    },
    {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
    }
],
```

```

        {
            "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
            "Status": "COMPLETED"
        },
        {
            "Name": "CREATE_CUSTOM_ENDPOINTS",
            "Status": "COMPLETED"
        }
    ],
    "Status": "AVAILABLE",
    "CreateTime": "2022-02-25T21:12:00.288000+00:00"
}
]
}

```

Para obtener más información, consulte [Visualización de una implementación azul/verde](#) en la Guía del usuario de Amazon Aurora.

Ejemplo 3: Para describir una implementación azul/verde para un SQL clúster Aurora My después de la conmutación

En el siguiente `describe-blue-green-deployment` ejemplo, se recuperan los detalles sobre una implementación azul/verde después de que el entorno verde se haya convertido en el entorno de producción.

```

aws rds describe-blue-green-deployments \
  --blue-green-deployment-identifier bgd-wi89nwzglccsfake

```

Salida:

```

{
  "BlueGreenDeployments": [
    {
      "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
      "BlueGreenDeploymentName": "my-blue-green-deployment",
      "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-old1",
      "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
      "SwitchoverDetails": [
        {

```

```
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
        "Status": "SWITCHOVER_COMPLETED"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-1-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-1",
        "Status": "SWITCHOVER_COMPLETED"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-2-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-2",
        "Status": "SWITCHOVER_COMPLETED"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-3-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-cluster-3",
        "Status": "SWITCHOVER_COMPLETED"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-excluded-member-endpoint-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-excluded-member-endpoint",
        "Status": "SWITCHOVER_COMPLETED"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-reader-endpoint-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-reader-endpoint",
        "Status": "SWITCHOVER_COMPLETED"
    }
],
"Tasks": [
    {
```

```

        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
    },
    {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
    },
    {
        "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
        "Status": "COMPLETED"
    },
    {
        "Name": "CREATE_CUSTOM_ENDPOINTS",
        "Status": "COMPLETED"
    }
],
"Status": "SWITCHOVER_COMPLETED",
"CreateTime": "2022-02-25T22:38:49.522000+00:00"
}
]
}

```

Para obtener más información, consulte [Visualización de una implementación azul/verde](#) en la Guía del usuario de Amazon Aurora.

Ejemplo 4: Para describir una implementación combinada azul y verde

En el siguiente describe-blue-green-deployment ejemplo, se recuperan los detalles de una implementación combinada azul y verde.

```
aws rds describe-blue-green-deployments
```

Salida:

```

{
  "BlueGreenDeployments": [
    {
      "BlueGreenDeploymentIdentifier": "bgd-wi89nwzgfakelccs",
      "BlueGreenDeploymentName": "my-blue-green-deployment",
      "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
      "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3rnukl",
    }
  ]
}

```

```
    "SwitchoverDetails": [  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-  
aurora-mysql-cluster",  
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-  
aurora-mysql-cluster-green-3rnukl",  
        "Status": "AVAILABLE"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-  
aurora-mysql-cluster-1",  
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-  
aurora-mysql-cluster-1-green-gpmaxf",  
        "Status": "AVAILABLE"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-  
aurora-mysql-cluster-2",  
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-  
aurora-mysql-cluster-2-green-j2oajq",  
        "Status": "AVAILABLE"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-  
aurora-mysql-cluster-3",  
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-  
aurora-mysql-cluster-3-green-mkxies",  
        "Status": "AVAILABLE"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-  
endpoint:my-excluded-member-endpoint",  
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-  
endpoint:my-excluded-member-endpoint-green-4sqjrq",  
        "Status": "AVAILABLE"  
      },  
      {  
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-  
endpoint:my-reader-endpoint",  
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-  
endpoint:my-reader-endpoint-green-gwwzlg",  
        "Status": "AVAILABLE"  
      }  
    ],  
  ],
```

```

    "Tasks": [
      {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
      },
      {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
      },
      {
        "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
        "Status": "COMPLETED"
      },
      {
        "Name": "CREATE_CUSTOM_ENDPOINTS",
        "Status": "COMPLETED"
      }
    ],
    "Status": "AVAILABLE",
    "CreateTime": "2022-02-25T21:12:00.288000+00:00"
  },
  {
    "BlueGreenDeploymentIdentifier": "bgd-v5330365fake1eex",
    "BlueGreenDeploymentName": "bgd-cli-test-instance",
    "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-old1",
    "Target": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
    "SwitchoverDetails": [
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance",
        "Status": "SWITCHOVER_COMPLETED"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1",
        "Status": "SWITCHOVER_COMPLETED"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2-old1",

```

```

        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2",
        "Status": "SWITCHOVER_COMPLETED"
    },
    {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3-old1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3",
        "Status": "SWITCHOVER_COMPLETED"
    }
],
"Tasks": [
    {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
    },
    {
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
    },
    {
        "Name": "CONFIGURE_BACKUPS",
        "Status": "COMPLETED"
    },
    {
        "Name": "CREATING_TOPOLOGY_OF_SOURCE",
        "Status": "COMPLETED"
    }
],
"Status": "SWITCHOVER_COMPLETED",
"CreateTime": "2022-02-25T22:33:22.225000+00:00"
}
]
}

```

Para obtener más información, consulte [Visualización de una implementación azul/verde](#) en la Guía RDS del usuario de Amazon y [Visualización de una implementación azul/verde](#) en la Guía del usuario de Amazon Aurora.

- Para obtener API más información, consulte la Referencia de comandos [DescribeBlueGreenDeployments](#).AWS CLI



## describe-certificates

En el siguiente ejemplo de código, se muestra cómo usar `describe-certificates`.

### AWS CLI

Para describir los certificados

En el siguiente `describe-certificates` ejemplo, se recuperan los detalles del certificado asociado a la región predeterminada del usuario.

```
aws rds describe-certificates
```

Salida:

```
{
  "Certificates": [
    {
      "CertificateIdentifier": "rds-ca-ecc384-g1",
      "CertificateType": "CA",
      "Thumbprint": "2ee3dcc06e50192559b13929e73484354f23387d",
      "ValidFrom": "2021-05-24T22:06:59+00:00",
      "ValidTill": "2121-05-24T23:06:59+00:00",
      "CertificateArn": "arn:aws:rds:us-west-2::cert:rds-ca-ecc384-g1",
      "CustomerOverride": false
    },
    {
      "CertificateIdentifier": "rds-ca-rsa4096-g1",
      "CertificateType": "CA",
      "Thumbprint": "19da4f2af579a8ae1f6a0fa77aa5befd874b4cab",
      "ValidFrom": "2021-05-24T22:03:20+00:00",
      "ValidTill": "2121-05-24T23:03:20+00:00",
      "CertificateArn": "arn:aws:rds:us-west-2::cert:rds-ca-rsa4096-g1",
      "CustomerOverride": false
    },
    {
      "CertificateIdentifier": "rds-ca-rsa2048-g1",
      "CertificateType": "CA",
      "Thumbprint": "7c40cb42714b6fdb2b296f9bbd0e8bb364436a76",
      "ValidFrom": "2021-05-24T21:59:00+00:00",
      "ValidTill": "2061-05-24T22:59:00+00:00",
      "CertificateArn": "arn:aws:rds:us-west-2::cert:rds-ca-rsa2048-g1",
      "CustomerOverride": true,
    }
  ]
}
```

```

    "CustomerOverrideValidTill": "2061-05-24T22:59:00+00:00"
  },
  {
    "CertificateIdentifier": "rds-ca-2019",
    "CertificateType": "CA",
    "Thumbprint": "d40ddb29e3750dffa671c3140bbf5f478d1c8096",
    "ValidFrom": "2019-08-22T17:08:50+00:00",
    "ValidTill": "2024-08-22T17:08:50+00:00",
    "CertificateArn": "arn:aws:rds:us-west-2::cert:rds-ca-2019",
    "CustomerOverride": false
  }
],
"DefaultCertificateForNewLaunches": "rds-ca-rsa2048-g1"
}

```

Para obtener más información, consulte [Uso deSSL/TLS para cifrar una conexión a una instancia de base de datos](#) en la Guía del RDS usuario de Amazon y [Uso deSSL/TLS para cifrar una conexión a un clúster](#) de base de datos en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte la Referencia [DescribeCertificates](#) de AWS CLI comandos.

## describe-db-cluster-backtracks

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-cluster-backtracks`.

### AWS CLI

Para describir los retrocesos de un clúster de base de datos

El siguiente `describe-db-cluster-backtracks` ejemplo recupera detalles sobre el clúster de base de datos especificado.

```

aws rds describe-db-cluster-backtracks \
  --db-cluster-identifier mydbcluster

```

Salida:

```

{
  "DBClusterBacktracks": [
    {

```

```

        "DBClusterIdentifier": "mydbcluster",
        "BacktrackIdentifier": "2f5f5294-0dd2-44c9-9f50-EXAMPLE",
        "BacktrackTo": "2021-02-12T04:59:22Z",
        "BacktrackedFrom": "2021-02-12T14:37:31.640Z",
        "BacktrackRequestCreationTime": "2021-02-12T14:36:18.819Z",
        "Status": "COMPLETED"
    },
    {
        "DBClusterIdentifier": "mydbcluster",
        "BacktrackIdentifier": "3c7a6421-af2a-4ea3-ae95-EXAMPLE",
        "BacktrackTo": "2021-02-11T22:53:46Z",
        "BacktrackedFrom": "2021-02-12T00:09:27.006Z",
        "BacktrackRequestCreationTime": "2021-02-12T00:07:53.487Z",
        "Status": "COMPLETED"
    }
]
}

```

Para obtener más información, consulte [Seguimiento de un clúster de base de datos Aurora](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbClusterBacktracks](#) la Referencia de AWS CLI comandos.

## describe-db-cluster-endpoints

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-cluster-endpoints`.

AWS CLI

Ejemplo 1: Para describir los puntos finales de un clúster de base de datos

El siguiente `describe-db-cluster-endpoints` ejemplo recupera los detalles de los puntos finales de su clúster de base de datos. Los tipos más comunes de cúmulos de Aurora tienen dos puntos finales. Un punto final tiene un tipo `WRITER`. Puede utilizar este punto final para todas las SQL sentencias. El otro punto final tiene el tipo `READER` Puede usar este punto final solo para `SELECT` y otras declaraciones de solo lectura SQL.

```
aws rds describe-db-cluster-endpoints
```

Salida:

```
{
  "DBClusterEndpoints": [
    {
      "DBClusterIdentifier": "my-database-1",
      "Endpoint": "my-database-1.cluster-cnpexample.us-east-1.rds.amazonaws.com",
      "Status": "creating",
      "EndpointType": "WRITER"
    },
    {
      "DBClusterIdentifier": "my-database-1",
      "Endpoint": "my-database-1.cluster-ro-cnpexample.us-east-1.rds.amazonaws.com",
      "Status": "creating",
      "EndpointType": "READER"
    },
    {
      "DBClusterIdentifier": "mydbcluster",
      "Endpoint": "mydbcluster.cluster-cnpexample.us-east-1.rds.amazonaws.com",
      "Status": "available",
      "EndpointType": "WRITER"
    },
    {
      "DBClusterIdentifier": "mydbcluster",
      "Endpoint": "mydbcluster.cluster-ro-cnpexample.us-east-1.rds.amazonaws.com",
      "Status": "available",
      "EndpointType": "READER"
    }
  ]
}
```

Ejemplo 2: Para describir los puntos finales de un clúster de base de datos único

El siguiente `describe-db-cluster-endpoints` ejemplo recupera los detalles de los puntos finales del clúster de base de datos de un único clúster de base de datos especificado. Los clústeres Aurora Serverless solo tienen un punto final con un tipo de `WRITER`.

```
aws rds describe-db-cluster-endpoints \
  --db-cluster-identifier serverless-cluster
```

Salida:

```
{
  "DBClusterEndpoints": [
    {
      "Status": "available",
      "Endpoint": "serverless-cluster.cluster-cnpexample.us-
east-1.rds.amazonaws.com",
      "DBClusterIdentifier": "serverless-cluster",
      "EndpointType": "WRITER"
    }
  ]
}
```

Para obtener más información, consulte [Amazon Aurora Connection Management](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbClusterEndpoints](#) la Referencia de AWS CLI comandos.

## describe-db-cluster-parameter-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-cluster-parameter-groups`.

### AWS CLI

Para describir los grupos de parámetros del clúster de bases de datos

En el siguiente `describe-db-cluster-parameter-groups` ejemplo, se recuperan los detalles de los grupos de parámetros del clúster de base de datos.

```
aws rds describe-db-cluster-parameter-groups
```

Salida:

```
{
  "DBClusterParameterGroups": [
    {
      "DBClusterParameterGroupName": "default.aurora-mysql5.7",
      "DBParameterGroupFamily": "aurora-mysql5.7",
      "Description": "Default cluster parameter group for aurora-mysql5.7",
      "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:default.aurora-mysql5.7"
    }
  ]
}
```

```

    },
    {
      "DBClusterParameterGroupName": "default.aurora-postgresql9.6",
      "DBParameterGroupFamily": "aurora-postgresql9.6",
      "Description": "Default cluster parameter group for aurora-
postgresql9.6",
      "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:default.aurora-postgresql9.6"
    },
    {
      "DBClusterParameterGroupName": "default.aurora5.6",
      "DBParameterGroupFamily": "aurora5.6",
      "Description": "Default cluster parameter group for aurora5.6",
      "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:default.aurora5.6"
    },
    {
      "DBClusterParameterGroupName": "mydbclusterpg",
      "DBParameterGroupFamily": "aurora-mysql5.7",
      "Description": "My DB cluster parameter group",
      "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:mydbclusterpg"
    },
    {
      "DBClusterParameterGroupName": "mydbclusterpgcopy",
      "DBParameterGroupFamily": "aurora-mysql5.7",
      "Description": "Copy of mydbclusterpg parameter group",
      "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:mydbclusterpgcopy"
    }
  ]
}

```

Para obtener más información, consulte [Trabajo con grupos de parámetros de bases de datos y grupos de parámetros de clústeres de bases de datos](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbClusterParameterGroups](#) la Referencia de AWS CLI comandos.

## describe-db-cluster-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-cluster-parameters`.

## AWS CLI

Ejemplo 1: Para describir los parámetros de un grupo de parámetros de un clúster de base de datos

El siguiente `describe-db-cluster-parameters` ejemplo recupera detalles sobre los parámetros de un grupo de parámetros de un clúster de base de datos.

```
aws rds describe-db-cluster-parameters \  
  --db-cluster-parameter-group-name mydbclusterpg
```

Salida:

```
{  
  "Parameters": [  
    {  
      "ParameterName": "allow-suspicious-udfs",  
      "Description": "Controls whether user-defined functions that have only  
an xxx symbol for the main function can be loaded",  
      "Source": "engine-default",  
      "ApplyType": "static",  
      "DataType": "boolean",  
      "AllowedValues": "0,1",  
      "IsModifiable": false,  
      "ApplyMethod": "pending-reboot",  
      "SupportedEngineModes": [  
        "provisioned"  
      ]  
    },  
    {  
      "ParameterName": "aurora_lab_mode",  
      "ParameterValue": "0",  
      "Description": "Enables new features in the Aurora engine.",  
      "Source": "engine-default",  
      "ApplyType": "static",  
      "DataType": "boolean",  
      "AllowedValues": "0,1",  
      "IsModifiable": true,  
      "ApplyMethod": "pending-reboot",  
      "SupportedEngineModes": [  
        "provisioned"  
      ]  
    },  
  ],  
}
```

```

    ...some output truncated...
  ]
}

```

Ejemplo 2: Mostrar solo los nombres de los parámetros de un grupo de parámetros de un clúster de base de datos

El siguiente `describe-db-cluster-parameters` ejemplo recupera solo los nombres de los parámetros de un grupo de parámetros de un clúster de base de datos.

```

aws rds describe-db-cluster-parameters \
  --db-cluster-parameter-group-name default.aurora-mysql5.7 \
  --query 'Parameters[].{ParameterName:ParameterName}'

```

Salida:

```

[
  {
    "ParameterName": "allow-suspicious-udfs"
  },
  {
    "ParameterName": "aurora_binlog_read_buffer_size"
  },
  {
    "ParameterName": "aurora_binlog_replication_max_yield_seconds"
  },
  {
    "ParameterName": "aurora_binlog_use_large_read_buffer"
  },
  {
    "ParameterName": "aurora_lab_mode"
  },
  ...some output truncated...
]

```

Ejemplo 3: Para describir únicamente los parámetros modificables de un grupo de parámetros de un clúster de base de datos

El siguiente `describe-db-cluster-parameters` ejemplo recupera solo los nombres de los parámetros que puede modificar en un grupo de parámetros de un clúster de base de datos.



```
aws rds describe-db-cluster-parameters \
  --db-cluster-parameter-group-name default.aurora-mysql5.7 \
  --query 'Parameters[].{ParameterName:ParameterName,IsModifiable:IsModifiable} |
[?IsModifiable == `true`]'
```

Salida:

```
[
  {
    "ParameterName": "aurora_binlog_read_buffer_size",
    "IsModifiable": true
  },
  {
    "ParameterName": "aurora_binlog_replication_max_yield_seconds",
    "IsModifiable": true
  },
  {
    "ParameterName": "aurora_binlog_use_large_read_buffer",
    "IsModifiable": true
  },
  {
    "ParameterName": "aurora_lab_mode",
    "IsModifiable": true
  },
  ...some output truncated...
]
```

Ejemplo 4: Para describir únicamente los parámetros booleanos modificables de un grupo de parámetros de un clúster de base de datos

El siguiente `describe-db-cluster-parameters` ejemplo recupera solo los nombres de los parámetros que puede modificar en un grupo de parámetros de un clúster de base de datos y que tienen un tipo de datos booleano.

```
aws rds describe-db-cluster-parameters \
  --db-cluster-parameter-group-name default.aurora-mysql5.7 \
  --query 'Parameters[].
{ParameterName:ParameterName,DataType:DataType,IsModifiable:IsModifiable} | [?
DataType == `boolean`] | [?IsModifiable == `true`]'
```

Salida:

```
[
  {
    "DataType": "boolean",
    "ParameterName": "aurora_binlog_use_large_read_buffer",
    "IsModifiable": true
  },
  {
    "DataType": "boolean",
    "ParameterName": "aurora_lab_mode",
    "IsModifiable": true
  },
  {
    "DataType": "boolean",
    "ParameterName": "autocommit",
    "IsModifiable": true
  },
  {
    "DataType": "boolean",
    "ParameterName": "automatic_sp_privileges",
    "IsModifiable": true
  },
  ...some output truncated...
]
```

Para obtener más información, consulte [Trabajo con grupos de parámetros de bases de datos y grupos de parámetros de clústeres de bases de datos](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbClusterParameters](#) la Referencia de AWS CLI comandos.

## **describe-db-cluster-snapshot-attributes**

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-cluster-snapshot-attributes`.

### AWS CLI

Para describir los nombres y valores de los atributos de una instantánea de un clúster de base de datos

El siguiente `describe-db-cluster-snapshot-attributes` ejemplo recupera los detalles de los nombres y valores de los atributos de la instantánea del clúster de base de datos especificada.

```
aws rds describe-db-cluster-snapshot-attributes \  
  --db-cluster-snapshot-identifier myclustersnapshot
```

Salida:

```
{  
  "DBClusterSnapshotAttributesResult": {  
    "DBClusterSnapshotIdentifier": "myclustersnapshot",  
    "DBClusterSnapshotAttributes": [  
      {  
        "AttributeName": "restore",  
        "AttributeValues": [  
          "123456789012"  
        ]  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [Compartir una instantánea de un clúster](#) de base de datos en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbClusterSnapshotAttributes](#) la Referencia de AWS CLI comandos.

## **describe-db-cluster-snapshots**

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-cluster-snapshots`.

AWS CLI

Para describir una instantánea de un clúster de base de datos para un clúster de base de datos

El siguiente `describe-db-cluster-snapshots` ejemplo recupera los detalles de las instantáneas del clúster de base de datos del clúster de base de datos especificado.

```
aws rds describe-db-cluster-snapshots \  
  --db-cluster-snapshot-identifier myclustersnapshot
```

```
--db-cluster-identifier mydbcluster
```

Salida:

```
{
  "DBClusterSnapshots": [
    {
      "AvailabilityZones": [
        "us-east-1a",
        "us-east-1b",
        "us-east-1e"
      ],
      "DBClusterSnapshotIdentifier": "myclustersnapshotcopy",
      "DBClusterIdentifier": "mydbcluster",
      "SnapshotCreateTime": "2019-06-04T09:16:42.649Z",
      "Engine": "aurora-mysql",
      "AllocatedStorage": 0,
      "Status": "available",
      "Port": 0,
      "VpcId": "vpc-6594f31c",
      "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
      "MasterUsername": "myadmin",
      "EngineVersion": "5.7.mysql_aurora.2.04.2",
      "LicenseModel": "aurora-mysql",
      "SnapshotType": "manual",
      "PercentProgress": 100,
      "StorageEncrypted": true,
      "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/
AKIAIOSFODNN7EXAMPLE",
      "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:814387698303:cluster-
snapshot:myclustersnapshotcopy",
      "IAMDatabaseAuthenticationEnabled": false
    },
    {
      "AvailabilityZones": [
        "us-east-1a",
        "us-east-1b",
        "us-east-1e"
      ],
      "DBClusterSnapshotIdentifier": "rds:mydbcluster-2019-06-20-09-16",
      "DBClusterIdentifier": "mydbcluster",
      "SnapshotCreateTime": "2019-06-20T09:16:26.569Z",
      "Engine": "aurora-mysql",

```

```

    "AllocatedStorage": 0,
    "Status": "available",
    "Port": 0,
    "VpcId": "vpc-6594f31c",
    "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
    "MasterUsername": "myadmin",
    "EngineVersion": "5.7.mysql_aurora.2.04.2",
    "LicenseModel": "aurora-mysql",
    "SnapshotType": "automated",
    "PercentProgress": 100,
    "StorageEncrypted": true,
    "KmsKeyId": "arn:aws:kms:us-east-1:814387698303:key/
AKIAIOSFODNN7EXAMPLE",
    "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:123456789012:cluster-
snapshot:rds:mydbcluster-2019-06-20-09-16",
    "IAMDatabaseAuthenticationEnabled": false
  }
]
}

```

Para obtener más información, consulte [Creación de una instantánea de clúster](#) de base de datos en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbClusterSnapshots](#) la Referencia de AWS CLI comandos.

## describe-db-clusters

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-clusters`.

### AWS CLI

Ejemplo 1: Para describir un clúster de base de datos

El siguiente `describe-db-clusters` ejemplo recupera los detalles del clúster de base de datos especificado.

```
aws rds describe-db-clusters \
  --db-cluster-identifier mydbcluster
```

Salida:

```
{
```

```
"DBClusters": [  
  {  
    "AllocatedStorage": 1,  
    "AvailabilityZones": [  
      "us-east-1a",  
      "us-east-1b",  
      "us-east-1e"  
    ],  
    "BackupRetentionPeriod": 1,  
    "DatabaseName": "mydbcluster",  
    "DBClusterIdentifier": "mydbcluster",  
    "DBClusterParameterGroup": "default.aurora-mysql5.7",  
    "DBSubnetGroup": "default",  
    "Status": "available",  
    "EarliestRestorableTime": "2019-06-19T09:16:28.210Z",  
    "Endpoint": "mydbcluster.cluster-cnpeexample.us-  
east-1.rds.amazonaws.com",  
    "ReaderEndpoint": "mydbcluster.cluster-ro-cnpeexample.us-  
east-1.rds.amazonaws.com",  
    "MultiAZ": true,  
    "Engine": "aurora-mysql",  
    "EngineVersion": "5.7.mysql_aurora.2.04.2",  
    "LatestRestorableTime": "2019-06-20T22:38:14.908Z",  
    "Port": 3306,  
    "MasterUsername": "myadmin",  
    "PreferredBackupWindow": "09:09-09:39",  
    "PreferredMaintenanceWindow": "sat:04:09-sat:04:39",  
    "ReadReplicaIdentifiers": [],  
    "DBClusterMembers": [  
      {  
        "DBInstanceIdentifier": "dbinstance3",  
        "IsClusterWriter": false,  
        "DBClusterParameterGroupStatus": "in-sync",  
        "PromotionTier": 1  
      },  
      {  
        "DBInstanceIdentifier": "dbinstance1",  
        "IsClusterWriter": false,  
        "DBClusterParameterGroupStatus": "in-sync",  
        "PromotionTier": 1  
      },  
      {  
        "DBInstanceIdentifier": "dbinstance2",  
        "IsClusterWriter": false,
```

```

        "DBClusterParameterGroupStatus": "in-sync",
        "PromotionTier": 1
    },
    {
        "DBInstanceIdentifier": "mydbcluster",
        "IsClusterWriter": false,
        "DBClusterParameterGroupStatus": "in-sync",
        "PromotionTier": 1
    },
    {
        "DBInstanceIdentifier": "mydbcluster-us-east-1b",
        "IsClusterWriter": false,
        "DBClusterParameterGroupStatus": "in-sync",
        "PromotionTier": 1
    },
    {
        "DBInstanceIdentifier": "mydbcluster",
        "IsClusterWriter": true,
        "DBClusterParameterGroupStatus": "in-sync",
        "PromotionTier": 1
    }
],
"VpcSecurityGroups": [
    {
        "VpcSecurityGroupId": "sg-0b9130572daf3dc16",
        "Status": "active"
    }
],
"HostedZoneId": "Z2R2ITUGPM61AM",
"StorageEncrypted": true,
"KmsKeyId": "arn:aws:kms:us-east-1:814387698303:key/
AKIAIOSFODNN7EXAMPLE",
"DbClusterResourceId": "cluster-AKIAIOSFODNN7EXAMPLE",
"DBClusterArn": "arn:aws:rds:us-
east-1:123456789012:cluster:mydbcluster",
"AssociatedRoles": [],
"IAMDatabaseAuthenticationEnabled": false,
"ClusterCreateTime": "2019-04-15T14:18:42.785Z",
"EngineMode": "provisioned",
"DeletionProtection": false,
"HttpEndpointEnabled": false
}
]

```

```
}

```

Ejemplo 2: Para enumerar determinados atributos de todos los clústeres de bases de datos

El siguiente `describe-db-clusters` ejemplo recupera solo los `ReaderEndpoint` atributos `DBClusterIdentifierEndpoint`, y de todos los clústeres de base de datos de la AWS región actual.

```
aws rds describe-db-clusters \
  --query 'DBClusters[.
  {DBClusterIdentifier:DBClusterIdentifier,Endpoint:Endpoint,ReaderEndpoint:ReaderEndpoint}]'
```

Salida:

```
[
  {
    "Endpoint": "cluster-57-2020-05-01-2270.cluster-cnpxexample.us-
east-1.rds.amazonaws.com",
    "ReaderEndpoint": "cluster-57-2020-05-01-2270.cluster-ro-cnpxexample.us-
east-1.rds.amazonaws.com",
    "DBClusterIdentifier": "cluster-57-2020-05-01-2270"
  },
  {
    "Endpoint": "cluster-57-2020-05-01-4615.cluster-cnpxexample.us-
east-1.rds.amazonaws.com",
    "ReaderEndpoint": "cluster-57-2020-05-01-4615.cluster-ro-cnpxexample.us-
east-1.rds.amazonaws.com",
    "DBClusterIdentifier": "cluster-57-2020-05-01-4615"
  },
  {
    "Endpoint": "pg2-cluster.cluster-cnpxexample.us-east-1.rds.amazonaws.com",
    "ReaderEndpoint": "pg2-cluster.cluster-ro-cnpxexample.us-
east-1.rds.amazonaws.com",
    "DBClusterIdentifier": "pg2-cluster"
  },
  ...output omitted...
]
```

Ejemplo 3: Para enumerar los clústeres de bases de datos con un atributo específico



El siguiente `describe-db-clusters` ejemplo recupera solo los Engine atributos `DBClusterIdentifier` y los atributos de los clústeres de base de datos que utilizan el motor de `aurora-postgresql` base de datos.

```
aws rds describe-db-clusters \  
  --query 'DBClusters[].{DBClusterIdentifier:DBClusterIdentifier,Engine:Engine} |  
  [?Engine == `aurora-postgresql`]
```

Salida:

```
[  
  {  
    "Engine": "aurora-postgresql",  
    "DBClusterIdentifier": "pg2-cluster"  
  }  
]
```

Para obtener más información, consulte los [clústeres de base](#) de datos de Amazon Aurora en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbClusters](#) la Referencia de AWS CLI comandos.

## `describe-db-engine-versions`

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-engine-versions`.

AWS CLI

Para describir las versiones del motor de base de datos del motor My SQL DB

El siguiente ejemplo `describe-db-engine-versions` se muestran detalles sobre cada una de las versiones del motor de base de datos especificado.

```
aws rds describe-db-engine-versions \  
  --engine mysql
```

Salida:

```
{
```

```
"DBEngineVersions": [
  {
    "Engine": "mysql",
    "EngineVersion": "5.5.46",
    "DBParameterGroupFamily": "mysql5.5",
    "DBEngineDescription": "MySQL Community Edition",
    "DBEngineVersionDescription": "MySQL 5.5.46",
    "ValidUpgradeTarget": [
      {
        "Engine": "mysql",
        "EngineVersion": "5.5.53",
        "Description": "MySQL 5.5.53",
        "AutoUpgrade": false,
        "IsMajorVersionUpgrade": false
      },
      {
        "Engine": "mysql",
        "EngineVersion": "5.5.54",
        "Description": "MySQL 5.5.54",
        "AutoUpgrade": false,
        "IsMajorVersionUpgrade": false
      },
      {
        "Engine": "mysql",
        "EngineVersion": "5.5.57",
        "Description": "MySQL 5.5.57",
        "AutoUpgrade": false,
        "IsMajorVersionUpgrade": false
      },
      ...some output truncated...
    ]
  }
]
```

Para obtener más información, consulte [¿Qué es Amazon Relational Database Service \(RDSAmazon\)?](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [escribeDBEnginelas versiones D](#) en la Referencia de AWS CLI comandos.

## **describe-db-instance-automated-backups**

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-instance-automated-backups`.

## AWS CLI

Para describir las copias de seguridad automatizadas de una instancia de base de datos

En el siguiente `describe-db-instance-automated-backups` ejemplo, se muestran detalles sobre las copias de seguridad automatizadas de la instancia de base de datos especificada. Los detalles incluyen las copias de seguridad automatizadas replicadas en otras AWS regiones.

```
aws rds describe-db-instance-automated-backups \
  --db-instance-identifier new-orcl-db
```

Salida:

```
{
  "DBInstanceAutomatedBackups": [
    {
      "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db",
      "DbiResourceId": "db-JKIB2GFQ5RV7REPLZA4EXAMPLE",
      "Region": "us-east-1",
      "DBInstanceIdentifier": "new-orcl-db",
      "RestoreWindow": {
        "EarliestTime": "2020-12-07T21:05:20.939Z",
        "LatestTime": "2020-12-07T21:05:20.939Z"
      },
      "AllocatedStorage": 20,
      "Status": "replicating",
      "Port": 1521,
      "InstanceCreateTime": "2020-12-04T15:28:31Z",
      "MasterUsername": "admin",
      "Engine": "oracle-se2",
      "EngineVersion": "12.1.0.2.v21",
      "LicenseModel": "bring-your-own-license",
      "OptionGroupName": "default:oracle-se2-12-1",
      "Encrypted": false,
      "StorageType": "gp2",
      "IAMDatabaseAuthenticationEnabled": false,
      "BackupRetentionPeriod": 14,
      "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-
west-2:123456789012:auto-backup:ab-jkib2gfg5rv7replzadabrktni2bn4example"
    }
  ]
}
```

Para obtener más información, consulte [Búsqueda de información sobre copias de seguridad replicadas](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DescribeDbInstanceAutomatedBackups](#) la Referencia de AWS CLI comandos.

## describe-db-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-instances`.

### AWS CLI

Descripción de una instancia de base de datos

En siguiente ejemplo `describe-db-instances`, se recuperan los detalles sobre la instancia de base de datos especificada.

```
aws rds describe-db-instances \
  --db-instance-identifier mydbinstancecf
```

Salida:

```
{
  "DBInstances": [
    {
      "DBInstanceIdentifier": "mydbinstancecf",
      "DBInstanceClass": "db.t3.small",
      "Engine": "mysql",
      "DBInstanceStatus": "available",
      "MasterUsername": "masterawsuser",
      "Endpoint": {
        "Address": "mydbinstancecf.abcxample.us-east-1.rds.amazonaws.com",
        "Port": 3306,
        "HostedZoneId": "Z2R2ITUGPM61AM"
      },
      "...some output truncated..."
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeDBInstances](#) en la Referencia de AWS CLI comandos.

## describe-db-log-files

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-log-files`.

### AWS CLI

Para describir los archivos de registro de una instancia de base de datos

El siguiente `describe-db-log-files` ejemplo recupera detalles sobre los archivos de registro de la instancia de base de datos especificada.

```
aws rds describe-db-log-files -\
  -db-instance-identifier test-instance
```

Salida:

```
{
  "DescribeDBLogFiles": [
    {
      "Size": 0,
      "LastWritten": 1533060000000,
      "LogFileName": "error/mysql-error-running.log"
    },
    {
      "Size": 2683,
      "LastWritten": 1532994300000,
      "LogFileName": "error/mysql-error-running.log.0"
    },
    {
      "Size": 107,
      "LastWritten": 1533057300000,
      "LogFileName": "error/mysql-error-running.log.18"
    },
    {
      "Size": 13105,
      "LastWritten": 1532991000000,
      "LogFileName": "error/mysql-error-running.log.23"
    },
    {
      "Size": 0,
      "LastWritten": 1533061200000,
      "LogFileName": "error/mysql-error.log"
    },
  ],
}
```

```

    {
      "Size": 3519,
      "LastWritten": 1532989252000,
      "LogFileName": "mysqlUpgrade"
    }
  ]
}

```

- Para API obtener más información, consulte [DescribeDbLogFiles](#) la Referencia de AWS CLI comandos.

## describe-db-parameter-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-parameter-groups`.

AWS CLI

Descripción del grupo de parámetros de base de datos

En el siguiente ejemplo `describe-db-parameter-groups`, se recuperan los detalles sobre los grupos de parámetros de base de datos.

```
aws rds describe-db-parameter-groups
```

Salida:

```

{
  "DBParameterGroups": [
    {
      "DBParameterGroupName": "default.aurora-mysql5.7",
      "DBParameterGroupFamily": "aurora-mysql5.7",
      "Description": "Default parameter group for aurora-mysql5.7",
      "DBParameterGroupArn": "arn:aws:rds:us-east-1:123456789012:pg:default.aurora-mysql5.7"
    },
    {
      "DBParameterGroupName": "default.aurora-postgresql9.6",
      "DBParameterGroupFamily": "aurora-postgresql9.6",
      "Description": "Default parameter group for aurora-postgresql9.6",
      "DBParameterGroupArn": "arn:aws:rds:us-east-1:123456789012:pg:default.aurora-postgresql9.6"
    }
  ]
}

```

```

    },
    {
      "DBParameterGroupName": "default.aurora5.6",
      "DBParameterGroupFamily": "aurora5.6",
      "Description": "Default parameter group for aurora5.6",
      "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.aurora5.6"
    },
    {
      "DBParameterGroupName": "default.mariadb10.1",
      "DBParameterGroupFamily": "mariadb10.1",
      "Description": "Default parameter group for mariadb10.1",
      "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.mariadb10.1"
    },
    ...some output truncated...
  ]
}

```

Para obtener más información, consulte [Trabajar con grupos de parámetros de base](#) de datos en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [escribeDBParameterlos grupos D](#) en la referencia de AWS CLI comandos.

## describe-db-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-parameters`.

### AWS CLI

Descripción de los parámetros de un grupo de parámetros de base de datos

En el siguiente ejemplo `describe-db-parameters`, se recuperan los detalles del grupo de parámetros de base de datos especificado.

```
aws rds describe-db-parameters \
  --db-parameter-group-name mydbpg
```

Salida:

```
{
```

```

"Parameters": [
  {
    "ParameterName": "allow-suspicious-udfs",
    "Description": "Controls whether user-defined functions that have only
an xxx symbol for the main function can be loaded",
    "Source": "engine-default",
    "ApplyType": "static",
    "DataType": "boolean",
    "AllowedValues": "0,1",
    "IsModifiable": false,
    "ApplyMethod": "pending-reboot"
  },
  {
    "ParameterName": "auto_generate_certs",
    "Description": "Controls whether the server autogenerates SSL key and
certificate files in the data directory, if they do not already exist.",
    "Source": "engine-default",
    "ApplyType": "static",
    "DataType": "boolean",
    "AllowedValues": "0,1",
    "IsModifiable": false,
    "ApplyMethod": "pending-reboot"
  },
  ...some output truncated...
]
}

```

Para obtener más información, consulte [Trabajar con grupos de parámetros de base](#) de datos en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [D escribeDBParameters](#) en la referencia de AWS CLI comandos.

## describe-db-proxies

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-proxies`.

### AWS CLI

Para describir un proxy de base de datos para una RDS base de datos

El siguiente `describe-db-proxies` ejemplo devuelve información sobre los proxies de base de datos.



**aws rds describe-db-proxies**

Salida:

```
{
  "DBProxies": [
    {
      "DBProxyName": "proxyExample1",
      "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-proxy:prx-0123a01b12345c0ab",
      "Status": "available",
      "EngineFamily": "PostgreSQL",
      "VpcId": "vpc-1234567",
      "VpcSecurityGroupIds": [
        "sg-1234"
      ],
      "VpcSubnetIds": [
        "subnetgroup1",
        "subnetgroup2"
      ],
      "Auth": "[
        {
          "Description": "proxydescription1"
          "AuthScheme": "SECRETS",
          "SecretArn": "arn:aws:secretsmanager:us-west-2:123456789123:secret:secretName-1234f",
          "IAMAuth": "DISABLED"
        }
      ]",
      "RoleArn": "arn:aws:iam::12345678912???:role/ProxyPostgreSQLRole",
      "Endpoint": "proxyExample1.proxy-ab0cd1efghij.us-east-1.rds.amazonaws.com",
      "RequireTLS": false,
      "IdleClientTimeout": 1800,
      "DebuggingLogging": false,
      "CreateDate": "2023-04-05T16:09:33.452000+00:00",
      "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"
    },
    {
      "DBProxyName": "proxyExample2",
      "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-proxy:prx-1234a12b23456c1ab",
      "Status": "available",
```

```

    "EngineFamily": "PostgreSQL",
    "VpcId": "sg-1234567",
    "VpcSecurityGroupIds": [
      "sg-1234"
    ],
    "VpcSubnetIds": [
      "subnetgroup1",
      "subnetgroup2"
    ],
    "Auth": "[
      {
        "Description": "proxydescription2"
        "AuthScheme": "SECRETS",
        "SecretArn": "aarn:aws:secretsmanager:us-
west-2:123456789123:secret:secretName-1234f",
        "IAMAuth": "DISABLED"
      }
    ]",
    "RoleArn": "arn:aws:iam::12345678912:role/ProxyPostgreSQLRole",
    "Endpoint": "proxyExample2.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
    "RequireTLS": false,
    "IdleClientTimeout": 1800,
    "DebuggingLogging": false,
    "CreateDate": "2022-01-05T16:19:33.452000+00:00",
    "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"
  }
]
}

```

Para obtener más información, consulte [Visualización de un RDS proxy](#) en la Guía del RDS usuario de Amazon y [Visualización de un RDS proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbProxies](#) la Referencia de AWS CLI comandos.

## describe-db-proxy-endpoints

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-proxy-endpoints`.

### AWS CLI

Para describir los puntos finales de un proxy de base de datos

El siguiente `describe-db-proxy-endpoints` ejemplo devuelve información sobre los puntos finales del proxy de base de datos.

```
aws rds describe-db-proxy-endpoints
```

Salida:

```
{
  "DBProxyEndpoints": [
    {
      "DBProxyEndpointName": "proxyEndpoint1",
      "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-
endpoint:prx-endpoint-0123a01b12345c0ab",
      "DBProxyName": "proxyExample",
      "Status": "available",
      "VpcId": "vpc-1234567",
      "VpcSecurityGroupIds": [
        "sg-1234"
      ],
      "VpcSubnetIds": [
        "subnetgroup1",
        "subnetgroup2"
      ],
      "Endpoint": "proxyEndpoint1.endpoint.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
      "CreateDate": "2023-04-05T16:09:33.452000+00:00",
      "TargetRole": "READ_WRITE",
      "IsDefault": false
    },
    {
      "DBProxyEndpointName": "proxyEndpoint2",
      "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-
endpoint:prx-endpoint-4567a01b12345c0ab",
      "DBProxyName": "proxyExample2",
      "Status": "available",
      "VpcId": "vpc1234567",
      "VpcSecurityGroupIds": [
        "sg-5678"
      ],
      "VpcSubnetIds": [
        "subnetgroup1",
        "subnetgroup2"
      ],
    }
  ]
}
```

```

        "Endpoint": "proxyEndpoint2.endpoint.proxy-cd1ef2klmnop.us-
east-1.rds.amazonaws.com",
        "CreateDate": "2023-04-05T16:09:33.452000+00:00",
        "TargetRole": "READ_WRITE",
        "IsDefault": false
    }
]
}

```

Para obtener más información, consulte [Visualización de un punto de enlace proxy](#) en la Guía del RDS usuario de Amazon y [Creación de un punto de enlace proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbProxyEndpoints](#) la Referencia de AWS CLI comandos.

## describe-db-proxy-target-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-proxy-target-groups`.

### AWS CLI

Para describir los puntos finales de un proxy de base de datos

El siguiente `describe-db-proxy-target-groups` ejemplo devuelve información sobre los grupos objetivo del proxy de base de datos.

```

aws rds describe-db-proxy-target-groups \
  --db-proxy-name proxyExample

```

Salida:

```

{
  "TargetGroups":
    {
      "DBProxyName": "proxyExample",
      "TargetGroupName": "default",
      "TargetGroupArn": "arn:aws:rds:us-east-1:123456789012:target-group:prx-
tg-0123a01b12345c0ab",
      "IsDefault": true,
      "Status": "available",
      "ConnectionPoolConfig": {

```

```

        "MaxConnectionsPercent": 100,
        "MaxIdleConnectionsPercent": 50,
        "ConnectionBorrowTimeout": 120,
        "SessionPinningFilters": []
    },
    "CreateDate": "2023-05-02T18:41:19.495000+00:00",
    "UpdatedDate": "2023-05-02T18:41:21.762000+00:00"
}
}

```

Para obtener más información, consulte [Visualización de un RDS proxy](#) en la Guía del RDS usuario de Amazon y [Visualización de un RDS proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbProxyTargetGroups](#) la Referencia de AWS CLI comandos.

## describe-db-proxy-targets

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-proxy-targets`.

### AWS CLI

Para describir los objetivos del proxy de base de datos

El siguiente `describe-db-proxy-targets` ejemplo devuelve información sobre los objetivos del proxy de base de datos.

```

aws rds describe-db-proxy-targets \
  --db-proxy-name proxyExample

```

Salida:

```

{
  "Targets": [
    {
      "Endpoint": "database1.ab0cd1efghij.us-east-1.rds.amazonaws.com",
      "TrackedClusterId": "database1",
      "RdsResourceId": "database1-instance-1",
      "Port": 3306,
      "Type": "RDS_INSTANCE",
      "Role": "READ_WRITE",
      "TargetHealth": {
        "State": "UNAVAILABLE",

```

```

        "Reason": "PENDING_PROXY_CAPACITY",
        "Description": "DBProxy Target is waiting for proxy to scale to
desired capacity"
    }
}
]
}

```

Para obtener más información, consulte [Visualización de un RDS proxy](#) en la Guía del RDS usuario de Amazon y [Visualización de un RDS proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbProxyTargets](#) la Referencia de AWS CLI comandos.

## describe-db-recommendations

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-recommendations`.

### AWS CLI

Ejemplo 1: Para enumerar todas las recomendaciones de la base de datos

En el siguiente `describe-db-recommendations` ejemplo, se enumeran todas las recomendaciones de bases de datos de su AWS cuenta.

```
aws rds describe-db-recommendations
```

Salida:

```

{
  "DBRecommendations": [
    {
      "RecommendationId": "12ab3cde-f456-7g8h-9012-i3j45678k9lm",
      "TypeId": "config_recommendation:old_minor_version",
      "Severity": "informational",
      "ResourceArn": "arn:aws:rds:us-west-2:111122223333:db:database-1",
      "Status": "active",
      "CreatedTime": "2024-02-21T23:14:19.292000+00:00",
      "UpdatedTime": "2024-02-21T23:14:19+00:00",
      "Detection": "***[resource-name]** is not running the latest minor DB
engine version",
      "Recommendation": "Upgrade to latest engine version",
    }
  ]
}

```

```
"Description": "Your database resources aren't running the latest minor
DB engine version. The latest minor version contains the latest security fixes and
other improvements.",
  "RecommendedActions": [
    {
      "ActionId": "12ab34c5de6fg7h89i0jk1lm234n5678",
      "Operation": "modifyDbInstance",
      "Parameters": [
        {
          "Key": "EngineVersion",
          "Value": "5.7.44"
        },
        {
          "Key": "DBInstanceIdentifier",
          "Value": "database-1"
        }
      ],
      "ApplyModes": [
        "immediately",
        "next-maintenance-window"
      ],
      "Status": "ready",
      "ContextAttributes": [
        {
          "Key": "Recommended value",
          "Value": "5.7.44"
        },
        {
          "Key": "Current engine version",
          "Value": "5.7.42"
        }
      ]
    }
  ],
  "Category": "security",
  "Source": "RDS",
  "TypeDetection": "**[resource-count] resources** are not running the
latest minor DB engine version",
  "TypeRecommendation": "Upgrade to latest engine version",
  "Impact": "Reduced database performance and data security at risk",
  "AdditionalInfo": "We recommend that you maintain your database with the
latest DB engine minor version as this version includes the latest security and
functionality fixes. The DB engine minor version upgrades contain only the changes
```

```

which are backward-compatible with earlier minor versions of the same major version
of the DB engine.",
    "Links": [
        {
            "Text": "Upgrading an RDS DB instance engine version",
            "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
USER_UpgradeDBInstance.Upgrading.html"
        },
        {
            "Text": "Using Amazon RDS Blue/Green Deployments for database
updates for Amazon Aurora",
            "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/
AuroraUserGuide/blue-green-deployments.html"
        },
        {
            "Text": "Using Amazon RDS Blue/Green Deployments for database
updates for Amazon RDS",
            "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
blue-green-deployments.html"
        }
    ]
}
]
}
}

```

Para obtener más información, consulte [Visualización y respuesta a RDS las recomendaciones de Amazon](#) en la Guía del RDS usuario de Amazon y [Visualización y respuesta a RDS las recomendaciones de Amazon](#) en la Guía del usuario de Amazon Aurora.

Ejemplo 2: Para enumerar las recomendaciones de bases de datos de alta gravedad

En el siguiente `describe-db-recommendations` ejemplo, se enumeran las recomendaciones de bases de datos de alta gravedad de su AWS cuenta.

```

aws rds describe-db-recommendations \
  --filters Name=severity,Values=high

```

Salida:

```

{
  "DBRecommendations": [
    {
      "RecommendationId": "12ab3cde-f456-7g8h-9012-i3j45678k9lm",

```



```
"TypeId": "config_recommendation::rds_extended_support",
"Severity": "high",
"ResourceArn": "arn:aws:rds:us-west-2:111122223333:db:database-1",
"Status": "active",
"CreatedTime": "2024-02-21T23:14:19.392000+00:00",
"UpdatedTime": "2024-02-21T23:14:19+00:00",
"Detection": "Your databases will be auto-enrolled to RDS Extended
Support on February 29",
"Recommendation": "Upgrade your major version before February 29, 2024
to avoid additional charges",
"Description": "Your PostgreSQL 11 and MySQL 5.7 databases will be
automatically enrolled into RDS Extended Support on February 29, 2024. To avoid
the increase in charges due to RDS Extended Support, we recommend upgrading your
databases to a newer major engine version before February 29, 2024.\n\nTo learn more
about the RDS Extended Support pricing, refer to the pricing page.",
"RecommendedActions": [
  {
    "ActionId": "12ab34c5de6fg7h89i0jk1lm234n5678",
    "Parameters": [],
    "ApplyModes": [
      "manual"
    ],
    "Status": "ready",
    "ContextAttributes": []
  }
],
"Category": "cost optimization",
"Source": "RDS",
"TypeDetection": "Your database will be auto-enrolled to RDS Extended
Support on February 29",
"TypeRecommendation": "Upgrade your major version before February 29,
2024 to avoid additional charges",
"Impact": "Increase in charges due to RDS Extended Support",
"AdditionalInfo": "With Amazon RDS Extended Support, you can continue
running your database on a major engine version past the RDS end of standard
support date for an additional cost. This paid feature gives you more time to
upgrade to a supported major engine version.\n\nDuring Extended Support, Amazon RDS
will supply critical CVE patches and bug fixes.",
"Links": [
  {
    "Text": "Amazon RDS Extended Support pricing for RDS for MySQL",
    "Url": "https://aws.amazon.com/rds/mysql/pricing/"
  }
]
```

```

        "Text": "Amazon RDS Extended Support for RDS for MySQL and
PostgreSQL databases",
        "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
extended-support.html"
    },
    {
        "Text": "Amazon RDS Extended Support pricing for Amazon Aurora
PostgreSQL",
        "Url": "https://aws.amazon.com/rds/aurora/pricing/"
    },
    {
        "Text": "Amazon RDS Extended Support for Aurora PostgreSQL
databases",
        "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/
AuroraUserGuide/extended-support.html"
    },
    {
        "Text": "Amazon RDS Extended Support pricing for RDS for
PostgreSQL",
        "Url": "https://aws.amazon.com/rds/postgresql/pricing/"
    }
]
}
]
}

```

Para obtener más información, consulte [Visualización y respuesta a RDS las recomendaciones de Amazon](#) en la Guía del RDS usuario de Amazon y [Visualización y respuesta a RDS las recomendaciones de Amazon](#) en la Guía del usuario de Amazon Aurora.

Ejemplo 3: Para enumerar las recomendaciones de base de datos para una instancia de base de datos específica

En el siguiente `describe-db-recommendations` ejemplo, se enumeran todas las recomendaciones de base de datos para una instancia de base de datos específica.

```

aws rds describe-db-recommendations \
  --filters Name=dbi-resource-id,Values=database-1

```

Salida:

```
{
```

```
"DBRecommendations": [
  {
    "RecommendationId": "12ab3cde-f456-7g8h-9012-i3j45678k9lm",
    "TypeId": "config_recommendation::old_minor_version",
    "Severity": "informational",
    "ResourceArn": "arn:aws:rds:us-west-2:111122223333:db:database-1",
    "Status": "active",
    "CreatedTime": "2024-02-21T23:14:19.292000+00:00",
    "UpdatedTime": "2024-02-21T23:14:19+00:00",
    "Detection": "***[resource-name]** is not running the latest minor DB
engine version",
    "Recommendation": "Upgrade to latest engine version",
    "Description": "Your database resources aren't running the latest minor
DB engine version. The latest minor version contains the latest security fixes and
other improvements.",
    "RecommendedActions": [
      {
        "ActionId": "12ab34c5de6fg7h89i0jk1lm234n5678",
        "Operation": "modifyDbInstance",
        "Parameters": [
          {
            "Key": "EngineVersion",
            "Value": "5.7.44"
          },
          {
            "Key": "DBInstanceIdentifier",
            "Value": "database-1"
          }
        ],
        "ApplyModes": [
          "immediately",
          "next-maintenance-window"
        ],
        "Status": "ready",
        "ContextAttributes": [
          {
            "Key": "Recommended value",
            "Value": "5.7.44"
          },
          {
            "Key": "Current engine version",
            "Value": "5.7.42"
          }
        ]
      }
    ]
  }
]
```

```

    }
  ],
  "Category": "security",
  "Source": "RDS",
  "TypeDetection": "**[resource-count] resources** are not running the
latest minor DB engine version",
  "TypeRecommendation": "Upgrade to latest engine version",
  "Impact": "Reduced database performance and data security at risk",
  "AdditionalInfo": "We recommend that you maintain your database with the
latest DB engine minor version as this version includes the latest security and
functionality fixes. The DB engine minor version upgrades contain only the changes
which are backward-compatible with earlier minor versions of the same major version
of the DB engine.",
  "Links": [
    {
      "Text": "Upgrading an RDS DB instance engine version",
      "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
USER_UpgradeDBInstance.Upgrading.html"
    },
    {
      "Text": "Using Amazon RDS Blue/Green Deployments for database
updates for Amazon Aurora",
      "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/
AuroraUserGuide/blue-green-deployments.html"
    },
    {
      "Text": "Using Amazon RDS Blue/Green Deployments for database
updates for Amazon RDS",
      "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
blue-green-deployments.html"
    }
  ]
}
]
}

```

Para obtener más información, consulte [Visualización y respuesta a RDS las recomendaciones de Amazon](#) en la Guía del RDS usuario de Amazon y [Visualización y respuesta a RDS las recomendaciones de Amazon](#) en la Guía del usuario de Amazon Aurora.

Ejemplo 4: Para enumerar todas las recomendaciones de bases de datos activas

En el siguiente describe-db-recommendations ejemplo, se enumeran todas las recomendaciones de bases de datos activas de su AWS cuenta.

```
aws rds describe-db-recommendations \  
  --filters Name=status,Values=active
```

Salida:

```
{  
  "DBRecommendations": [  
    {  
      "RecommendationId": "12ab3cde-f456-7g8h-9012-i3j45678k9lm",  
      "TypeId": "config_recommendation::old_minor_version",  
      "Severity": "informational",  
      "ResourceArn": "arn:aws:rds:us-west-2:111122223333:db:database-1",  
      "Status": "active",  
      "CreatedTime": "2024-02-21T23:14:19.292000+00:00",  
      "UpdatedTime": "2024-02-21T23:14:19+00:00",  
      "Detection": "***[resource-name]** is not running the latest minor DB  
engine version",  
      "Recommendation": "Upgrade to latest engine version",  
      "Description": "Your database resources aren't running the latest minor  
DB engine version. The latest minor version contains the latest security fixes and  
other improvements.",  
      "RecommendedActions": [  
        {  
          "ActionId": "12ab34c5de6fg7h89i0jk1lm234n5678",  
          "Operation": "modifyDbInstance",  
          "Parameters": [  
            {  
              "Key": "EngineVersion",  
              "Value": "5.7.44"  
            },  
            {  
              "Key": "DBInstanceIdentifier",  
              "Value": "database-1"  
            }  
          ],  
          "ApplyModes": [  
            "immediately",  
            "next-maintenance-window"  
          ],  
          "Status": "ready",
```

```
        "ContextAttributes": [
            {
                "Key": "Recommended value",
                "Value": "5.7.44"
            },
            {
                "Key": "Current engine version",
                "Value": "5.7.42"
            }
        ]
    },
    "Category": "security",
    "Source": "RDS",
    "TypeDetection": "***[resource-count] resources** are not running the
latest minor DB engine version",
    "TypeRecommendation": "Upgrade to latest engine version",
    "Impact": "Reduced database performance and data security at risk",
    "AdditionalInfo": "We recommend that you maintain your database with the
latest DB engine minor version as this version includes the latest security and
functionality fixes. The DB engine minor version upgrades contain only the changes
which are backward-compatible with earlier minor versions of the same major version
of the DB engine.",
    "Links": [
        {
            "Text": "Upgrading an RDS DB instance engine version",
            "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
USER_UpgradeDBInstance.Upgrading.html"
        },
        {
            "Text": "Using Amazon RDS Blue/Green Deployments for database
updates for Amazon Aurora",
            "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/
AuroraUserGuide/blue-green-deployments.html"
        },
        {
            "Text": "Using Amazon RDS Blue/Green Deployments for database
updates for Amazon RDS",
            "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
blue-green-deployments.html"
        }
    ]
}
]
```

```
}
```

Para obtener más información, consulte [Visualización y respuesta a RDS las recomendaciones de Amazon](#) en la Guía del RDS usuario de Amazon y [Visualización y respuesta a RDS las recomendaciones de Amazon](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbRecommendations](#) la Referencia de AWS CLI comandos.

## describe-db-security-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-security-groups`.

### AWS CLI

Para enumerar los grupos de seguridad de base de datos

En el siguiente `describe-db-security-groups` ejemplo, se enumeran los grupos de seguridad de base de datos.

```
aws rds describe-db-security-groups
```

Salida:

```
{
  "DBSecurityGroups": [
    {
      "OwnerId": "123456789012",
      "DBSecurityGroupName": "default",
      "DBSecurityGroupDescription": "default",
      "EC2SecurityGroups": [],
      "IPRanges": [],
      "DBSecurityGroupArn": "arn:aws:rds:us-
west-1:111122223333:secgrp:default"
    },
    {
      "OwnerId": "123456789012",
      "DBSecurityGroupName": "mysecgroup",
      "DBSecurityGroupDescription": "My Test Security Group",
      "VpcId": "vpc-1234567f",
      "EC2SecurityGroups": [],
      "IPRanges": [],

```

```

        "DBSecurityGroupArn": "arn:aws:rds:us-
west-1:111122223333:secgrp:mysecgroup"
    }
]
}

```

Para obtener más información, consulte la [lista de los grupos de seguridad de base de datos disponibles](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DescribeDbSecurityGroups](#) la Referencia de AWS CLI comandos.

## describe-db-shard-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-shard-groups`.

### AWS CLI

Ejemplo 1: Para describir los grupos de fragmentos de base de datos

En el siguiente `describe-db-shard-groups` ejemplo, se recuperan los detalles de los grupos de fragmentos de base de datos.

```
aws rds describe-db-shard-groups
```

Salida:

```

{
  "DBShardGroups": [
    {
      "DBShardGroupResourceId": "shardgroup-7bb446329da94788b3f957746example",
      "DBShardGroupIdentifier": "limitless-test-shard-grp",
      "DBClusterIdentifier": "limitless-test-cluster",
      "MaxACU": 768.0,
      "ComputeRedundancy": 0,
      "Status": "available",
      "PubliclyAccessible": true,
      "Endpoint": "limitless-test-cluster.limitless-cekyexample.us-
east-2.rds.amazonaws.com"
    },
    {
      "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",

```



```

        "DBShardGroupIdentifier": "my-db-shard-group",
        "DBClusterIdentifier": "my-sv2-cluster",
        "MaxACU": 768.0,
        "ComputeRedundancy": 0,
        "Status": "available",
        "PubliclyAccessible": false,
        "Endpoint": "my-sv2-cluster.limitless-cekyceexample.us-
east-2.rds.amazonaws.com"
    }
]
}

```

Para obtener más información, consulte los [clústeres de base](#) de datos de Amazon Aurora en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeDbShardGroups](#) la Referencia de AWS CLI comandos.

## describe-db-snapshot-attributes

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-snapshot-attributes`.

### AWS CLI

Para describir los nombres y valores de los atributos de una instantánea de base de datos

El siguiente `describe-db-snapshot-attributes` ejemplo describe los nombres y valores de los atributos de una instantánea de base de datos.

```

aws rds describe-db-snapshot-attributes \
  --db-snapshot-identifier mydbsnapshot

```

Salida:

```

{
  "DBSnapshotAttributesResult": {
    "DBSnapshotIdentifier": "mydbsnapshot",
    "DBSnapshotAttributes": [
      {
        "AttributeName": "restore",
        "AttributeValues": [
          "123456789012",
          "210987654321"
        ]
      }
    ]
  }
}

```

```

    ]
  }
]
}
}

```

Para obtener más información, consulte [Compartir una instantánea de base](#) de datos en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DescribeDbSnapshotAttributes](#) la Referencia de AWS CLI comandos.

## describe-db-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-snapshots`.

### AWS CLI

Ejemplo 1: Descripción de una instantánea de base de datos para una instancia de base de datos

En el siguiente ejemplo `describe-db-snapshots`, se recuperan los detalles de una instantánea de base de datos de una instancia de base de datos.

```

aws rds describe-db-snapshots \
  --db-snapshot-identifier mydbsnapshot

```

Salida:

```

{
  "DBSnapshots": [
    {
      "DBSnapshotIdentifier": "mydbsnapshot",
      "DBInstanceIdentifier": "mysqldb",
      "SnapshotCreateTime": "2018-02-08T22:28:08.598Z",
      "Engine": "mysql",
      "AllocatedStorage": 20,
      "Status": "available",
      "Port": 3306,
      "AvailabilityZone": "us-east-1f",
      "VpcId": "vpc-6594f31c",
      "InstanceCreateTime": "2018-02-08T22:24:55.973Z",
      "MasterUsername": "mysqladmin",
      "EngineVersion": "5.6.37",
    }
  ]
}

```

```

        "LicenseModel": "general-public-license",
        "SnapshotType": "manual",
        "OptionGroupName": "default:mysql-5-6",
        "PercentProgress": 100,
        "StorageType": "gp2",
        "Encrypted": false,
        "DBSnapshotArn": "arn:aws:rds:us-
east-1:123456789012:snapshot:mydbsnapshot",
        "IAMDatabaseAuthenticationEnabled": false,
        "ProcessorFeatures": [],
        "DbiResourceId": "db-AKIAIOSFODNN7EXAMPLE"
    }
]
}

```

Para obtener más información, consulte [Creación de una instantánea de base](#) de datos en la Guía del RDS usuario de Amazon.

Ejemplo 2: Búsqueda del número de instantáneas manuales realizadas

En el siguiente `describe-db-snapshots` ejemplo, se utiliza el `length` operador como `--query` opción para devolver el número de instantáneas manuales que se han realizado en una AWS región determinada.

```

aws rds describe-db-snapshots \
  --snapshot-type manual \
  --query "Length(*[].{DBSnapshots:SnapshotType})" \
  --region eu-central-1

```

Salida:

```
35
```

Para obtener más información, consulte [Creación de una instantánea de base](#) de datos en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DescribeDBSnapshots](#) en la referencia de AWS CLI comandos.

## describe-db-subnet-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-db-subnet-groups`.

## AWS CLI

Para describir un grupo de subredes de base de datos

El siguiente `describe-db-subnet-groups` ejemplo recupera los detalles del grupo de subredes de base de datos especificado.

```
aws rds describe-db-subnet-groups
```

Salida:

```
{
  "DBSubnetGroups": [
    {
      "DBSubnetGroupName": "mydbsubnetgroup",
      "DBSubnetGroupDescription": "My DB Subnet Group",
      "VpcId": "vpc-971c12ee",
      "SubnetGroupStatus": "Complete",
      "Subnets": [
        {
          "SubnetIdentifier": "subnet-d8c8e7f4",
          "SubnetAvailabilityZone": {
            "Name": "us-east-1a"
          },
          "SubnetStatus": "Active"
        },
        {
          "SubnetIdentifier": "subnet-718fdc7d",
          "SubnetAvailabilityZone": {
            "Name": "us-east-1f"
          },
          "SubnetStatus": "Active"
        },
        {
          "SubnetIdentifier": "subnet-cbc8e7e7",
          "SubnetAvailabilityZone": {
            "Name": "us-east-1a"
          },
          "SubnetStatus": "Active"
        },
        {
          "SubnetIdentifier": "subnet-0ccde220",
          "SubnetAvailabilityZone": {
```

```

        "Name": "us-east-1a"
      },
      "SubnetStatus": "Active"
    }
  ],
  "DBSubnetGroupArn": "arn:aws:rds:us-
east-1:123456789012:subgrp:mydbsubnetgroup"
}
]
}

```

Para obtener más información, consulte [Amazon Virtual Private Cloud VPCs y Amazon RDS](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DescribeDbSubnetGroups](#) la Referencia de AWS CLI comandos.

## describe-engine-default-cluster-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-engine-default-cluster-parameters`.

### AWS CLI

Para describir la información de los parámetros predeterminados del motor y del sistema para el motor de base de datos Aurora

En el siguiente `describe-engine-default-cluster-parameters` ejemplo, se recuperan los detalles de la información de los parámetros predeterminados del motor y del sistema para los clústeres de bases de datos Aurora compatibles con My SQL 5.7.

```

aws rds describe-engine-default-cluster-parameters \
  --db-parameter-group-family aurora-mysql5.7

```

Salida:

```

{
  "EngineDefaults": {
    "Parameters": [
      {
        "ParameterName": "aurora_load_from_s3_role",
        "Description": "IAM role ARN used to load data from AWS S3",

```

```

        "Source": "engine-default",
        "ApplyType": "dynamic",
        "DataType": "string",
        "IsModifiable": true,
        "SupportedEngineModes": [
            "provisioned"
        ]
    },
    ...some output truncated...
]
}
}

```

Para obtener más información, consulte [Trabajo con grupos de parámetros de bases de datos y grupos de parámetros de clústeres de bases de datos](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeEngineDefaultClusterParameters](#) la Referencia de AWS CLI comandos.

## describe-engine-default-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-engine-default-parameters`.

### AWS CLI

Para describir el motor predeterminado y la información sobre los parámetros del sistema para el motor de base de datos

En el siguiente `describe-engine-default-parameters` ejemplo, se recuperan los detalles de la información de los parámetros predeterminados del motor y del sistema para las instancias de base de datos My SQL 5.7.

```
aws rds describe-engine-default-parameters \
  --db-parameter-group-family mysql5.7
```

Salida:

```
{
  "EngineDefaults": {
    "Parameters": [
      {
```

```

        "ParameterName": "allow-suspicious-udfs",
        "Description": "Controls whether user-defined functions that have
only an xxx symbol for the main function can be loaded",
        "Source": "engine-default",
        "ApplyType": "static",
        "DataType": "boolean",
        "AllowedValues": "0,1",
        "IsModifiable": false
    },
    ...some output truncated...
]
}
}

```

Para obtener más información, consulte [Trabajar con grupos de parámetros de base](#) de datos en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DescribeEngineDefaultParameters](#) la Referencia de AWS CLI comandos.

## describe-event-categories

En el siguiente ejemplo de código, se muestra cómo usar `describe-event-categories`.

### AWS CLI

Para describir las categorías de eventos

El siguiente `describe-event-categories` ejemplo recupera detalles sobre las categorías de eventos de todas las fuentes de eventos disponibles.

```
aws rds describe-event-categories
```

Salida:

```

{
  "EventCategoriesMapList": [
    {
      "SourceType": "db-instance",
      "EventCategories": [
        "deletion",
        "read replica",
        "failover",

```

```
        "restoration",
        "maintenance",
        "low storage",
        "configuration change",
        "backup",
        "creation",
        "availability",
        "recovery",
        "failure",
        "backtrack",
        "notification"
    ]
},
{
    "SourceType": "db-security-group",
    "EventCategories": [
        "configuration change",
        "failure"
    ]
},
{
    "SourceType": "db-parameter-group",
    "EventCategories": [
        "configuration change"
    ]
},
{
    "SourceType": "db-snapshot",
    "EventCategories": [
        "deletion",
        "creation",
        "restoration",
        "notification"
    ]
},
{
    "SourceType": "db-cluster",
    "EventCategories": [
        "failover",
        "failure",
        "notification"
    ]
},
{
```



```

        "SourceType": "db-cluster-snapshot",
        "EventCategories": [
            "backup"
        ]
    }
]
}

```

- Para API obtener más información, consulte [DescribeEventCategories](#) la Referencia de AWS CLI comandos.

## describe-event-subscriptions

En el siguiente ejemplo de código, se muestra cómo usar `describe-event-subscriptions`.

### AWS CLI

Para describir las suscripciones a eventos

En este ejemplo se describen todas las suscripciones a RDS eventos de Amazon de la AWS cuenta corriente.

```
aws rds describe-event-subscriptions
```

Salida:

```

{
  "EventSubscriptionsList": [
    {
      "EventCategoriesList": [
        "backup",
        "recovery"
      ],
      "Enabled": true,
      "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-
instance-events",
      "Status": "creating",
      "SourceType": "db-instance",
      "CustomerAwsId": "123456789012",
      "SubscriptionCreationTime": "2018-07-31 23:22:01.893",
      "CustSubscriptionId": "my-instance-events",
      "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events"
    }
  ]
}

```

```

    },
    ...some output truncated...
  ]
}

```

- Para API obtener más información, consulte [DescribeEventSubscriptions](#) la Referencia de AWS CLI comandos.

## describe-events

En el siguiente ejemplo de código, se muestra cómo usar `describe-events`.

### AWS CLI

Para describir los eventos

En el siguiente `describe-events` ejemplo, se recuperan los detalles de los eventos que se han producido en la instancia de base de datos especificada.

```

aws rds describe-events \
  --source-identifier test-instance \
  --source-type db-instance

```

Salida:

```

{
  "Events": [
    {
      "SourceType": "db-instance",
      "SourceIdentifier": "test-instance",
      "EventCategories": [
        "backup"
      ],
      "Message": "Backing up DB instance",
      "Date": "2018-07-31T23:09:23.983Z",
      "SourceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance"
    },
    {
      "SourceType": "db-instance",
      "SourceIdentifier": "test-instance",
      "EventCategories": [
        "backup"
      ]
    }
  ]
}

```

```

    ],
    "Message": "Finished DB Instance backup",
    "Date": "2018-07-31T23:15:13.049Z",
    "SourceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance"
  }
]
}

```

- Para API obtener más información, consulte [DescribeEvents](#) la Referencia de AWS CLI comandos.

## describe-export-tasks

En el siguiente ejemplo de código, se muestra cómo usar `describe-export-tasks`.

### AWS CLI

Para describir las tareas de exportación de instantáneas

El siguiente `describe-export-tasks` ejemplo devuelve información sobre las exportaciones de instantáneas a Amazon S3.

```
aws rds describe-export-tasks
```

Salida:

```

{
  "ExportTasks": [
    {
      "ExportTaskIdentifier": "test-snapshot-export",
      "SourceArn": "arn:aws:rds:us-west-2:123456789012:snapshot:test-
snapshot",
      "SnapshotTime": "2020-03-02T18:26:28.163Z",
      "TaskStartTime": "2020-03-02T18:57:56.896Z",
      "TaskEndTime": "2020-03-02T19:10:31.985Z",
      "S3Bucket": "mybucket",
      "S3Prefix": "",
      "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/ExportRole",
      "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
abcd0000-7fca-4128-82f2-aabbccddeeff",
      "Status": "COMPLETE",
      "PercentProgress": 100,

```

```

        "TotalExtractedDataInGB": 0
    },
    {
        "ExportTaskIdentifier": "my-s3-export",
        "SourceArn": "arn:aws:rds:us-west-2:123456789012:snapshot:db5-snapshot-
test",
        "SnapshotTime": "2020-03-27T20:48:42.023Z",
        "S3Bucket": "mybucket",
        "S3Prefix": "",
        "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/ExportRole",
        "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
abcd0000-7fca-4128-82f2-aabbccddeeff",
        "Status": "STARTING",
        "PercentProgress": 0,
        "TotalExtractedDataInGB": 0
    }
]
}

```

Para obtener más información, consulta [Cómo monitorizar las exportaciones de instantáneas](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DescribeExportTasks](#) la Referencia de AWS CLI comandos.

## describe-global-clusters

En el siguiente ejemplo de código, se muestra cómo usar `describe-global-clusters`.

### AWS CLI

Para describir los clústeres de bases de datos globales

El siguiente `describe-global-clusters` ejemplo muestra los clústeres de bases de datos globales de Aurora en la AWS región actual.

```
aws rds describe-global-clusters
```

Salida:

```

{
  "GlobalClusters": [
    {

```

```

    "GlobalClusterIdentifier": "myglobalcluster",
    "GlobalClusterResourceId": "cluster-f5982077e3b5aabb",
    "GlobalClusterArn": "arn:aws:rds::123456789012:global-
cluster:myglobalcluster",
    "Status": "available",
    "Engine": "aurora-mysql",
    "EngineVersion": "5.7.mysql_aurora.2.07.2",
    "StorageEncrypted": false,
    "DeletionProtection": false,
    "GlobalClusterMembers": []
  }
]
}

```

Para obtener más información, consulte [Administrar una base de datos global de Aurora](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [DescribeGlobalClusters](#) la Referencia de AWS CLI comandos.

## describe-option-group-options

En el siguiente ejemplo de código, se muestra cómo usar `describe-option-group-options`.

### AWS CLI

Para describir todas las opciones disponibles

El siguiente `describe-option-group-options` ejemplo muestra dos opciones para una instancia de Oracle Database 19c.

```

aws rds describe-option-group-options \
  --engine-name oracle-ee \
  --major-engine-version 19 \
  --max-items 2

```

Salida:

```

{
  "OptionGroupOptions": [
    {
      "Name": "APEX",
      "Description": "Oracle Application Express Runtime Environment",

```

```

    "EngineName": "oracle-ee",
    "MajorEngineVersion": "19",
    "MinimumRequiredMinorEngineVersion": "0.0.0.ru-2019-07.rur-2019-07.r1",
    "PortRequired": false,
    "OptionsDependedOn": [],
    "OptionsConflictsWith": [],
    "Persistent": false,
    "Permanent": false,
    "RequiresAutoMinorEngineVersionUpgrade": false,
    "VpcOnly": false,
    "SupportsOptionVersionDowngrade": false,
    "OptionGroupOptionSettings": [],
    "OptionGroupOptionVersions": [
      {
        "Version": "19.1.v1",
        "IsDefault": true
      },
      {
        "Version": "19.2.v1",
        "IsDefault": false
      }
    ]
  },
  {
    "Name": "APEX-DEV",
    "Description": "Oracle Application Express Development Environment",
    "EngineName": "oracle-ee",
    "MajorEngineVersion": "19",
    "MinimumRequiredMinorEngineVersion": "0.0.0.ru-2019-07.rur-2019-07.r1",
    "PortRequired": false,
    "OptionsDependedOn": [
      "APEX"
    ],
    "OptionsConflictsWith": [],
    "Persistent": false,
    "Permanent": false,
    "RequiresAutoMinorEngineVersionUpgrade": false,
    "VpcOnly": false,
    "OptionGroupOptionSettings": []
  }
],
"NextToken": "eyJNYXJrZXIiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}

```

Para obtener más información, consulta Cómo [enumerar las opciones y los ajustes de opciones de un grupo de opciones](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DescribeOptionGroupOptions](#) la Referencia de AWS CLI comandos.

## describe-option-groups

En el siguiente ejemplo de código, se muestra cómo usar describe-option-groups.

### AWS CLI

Para describir los grupos de opciones disponibles

En el siguiente describe-option-groups ejemplo, se enumeran los grupos de opciones de una instancia de Oracle Database 19c.

```
aws rds describe-option-groups \  
  --engine-name oracle-ee \  
  --major-engine-version 19
```

Salida:

```
{  
  "OptionGroupsList": [  
    {  
      "OptionGroupName": "default:oracle-ee-19",  
      "OptionGroupDescription": "Default option group for oracle-ee 19",  
      "EngineName": "oracle-ee",  
      "MajorEngineVersion": "19",  
      "Options": [],  
      "AllowsVpcAndNonVpcInstanceMemberships": true,  
      "OptionGroupArn": "arn:aws:rds:us-west-1:111122223333:og:default:oracle-  
ee-19"  
    }  
  ]  
}
```

Para obtener más información, consulta Cómo [enumerar las opciones y los ajustes de opciones de un grupo de opciones](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DescribeOptionGroups](#) la Referencia de AWS CLI comandos.

## describe-orderable-db-instance-options

En el siguiente ejemplo de código, se muestra cómo usar `describe-orderable-db-instance-options`.

### AWS CLI

Descripción de las opciones de instancias de base de datos ordenables

En el siguiente `describe-orderable-db-instance-options` ejemplo, se obtienen detalles sobre las opciones de ordenación de las instancias de base de datos que ejecutan el motor MySQL DB.

```
aws rds describe-orderable-db-instance-options \  
  --engine mysql
```

Salida:

```
{  
  "OrderableDBInstanceOptions": [  
    {  
      "MinStorageSize": 5,  
      "ReadReplicaCapable": true,  
      "MaxStorageSize": 6144,  
      "AvailabilityZones": [  
        {  
          "Name": "us-east-1a"  
        },  
        {  
          "Name": "us-east-1b"  
        },  
        {  
          "Name": "us-east-1c"  
        },  
        {  
          "Name": "us-east-1d"  
        }  
      ],  
      "SupportsIops": false,  
    }  
  ]  
}
```



```
    "AvailableProcessorFeatures": [],
    "MultiAZCapable": true,
    "DBInstanceClass": "db.m1.large",
    "Vpc": true,
    "StorageType": "gp2",
    "LicenseModel": "general-public-license",
    "EngineVersion": "5.5.46",
    "SupportsStorageEncryption": false,
    "SupportsEnhancedMonitoring": true,
    "Engine": "mysql",
    "SupportsIAMDatabaseAuthentication": false,
    "SupportsPerformanceInsights": false
  }
]
...some output truncated...
}
```

- Para API obtener más información, consulte la Referencia [DescribeOrderableDBInstanceOptions](#) de AWS CLI comandos.

## describe-pending-maintenance-actions

En el siguiente ejemplo de código, se muestra cómo usar `describe-pending-maintenance-actions`.

AWS CLI

Para enumerar los recursos con al menos una acción de mantenimiento pendiente

En el siguiente `describe-pending-maintenance-actions` ejemplo, se muestra la acción de mantenimiento pendiente de una instancia de base de datos.

```
aws rds describe-pending-maintenance-actions
```

Salida:

```
{
  "PendingMaintenanceActions": [
    {
      "ResourceIdentifier": "arn:aws:rds:us-
west-2:123456789012:cluster:global-db1-cl1",
```

```

    "PendingMaintenanceActionDetails": [
      {
        "Action": "system-update",
        "Description": "Upgrade to Aurora PostgreSQL 2.4.2"
      }
    ]
  }
]
}

```

Para obtener más información, consulte [Mantenimiento de una instancia](#) de base de datos en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DescribePendingMaintenanceActions](#) la Referencia de AWS CLI comandos.

## describe-reserved-db-instances-offerings

En el siguiente ejemplo de código, se muestra cómo usar `describe-reserved-db-instances-offerings`.

### AWS CLI

Para describir las ofertas de instancias de base de datos reservadas

El siguiente `describe-reserved-db-instances-offerings` ejemplo recupera detalles sobre las opciones de instancias de base de datos reservadas para `oracle`.

```

aws rds describe-reserved-db-instances-offerings \
  --product-description oracle

```

Salida:

```

{
  "ReservedDBInstancesOfferings": [
    {
      "CurrencyCode": "USD",
      "UsagePrice": 0.0,
      "ProductDescription": "oracle-se2(li)",
      "ReservedDBInstancesOfferingId": "005bdee3-9ef4-4182-aa0c-58ef7cb6c2f8",
      "MultiAZ": true,

```

```
    "DBInstanceClass": "db.m4.xlarge",
    "OfferingType": "Partial Upfront",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": 0.594,
        "RecurringChargeFrequency": "Hourly"
      }
    ],
    "FixedPrice": 4089.0,
    "Duration": 31536000
  },
  ...some output truncated...
}
```

- Para API obtener más información, consulte [DescribeReservedDbInstancesOfferings](#) la Referencia de AWS CLI comandos.

## describe-reserved-db-instances

En el siguiente ejemplo de código, se muestra cómo usar `describe-reserved-db-instances`.

AWS CLI

Para describir las instancias de base de datos reservadas

En el siguiente `describe-reserved-db-instances` ejemplo, se recuperan detalles sobre cualquier instancia de base de datos reservada de la AWS cuenta actual.

```
aws rds describe-reserved-db-instances
```

Salida:

```
{
  "ReservedDBInstances": [
    {
      "ReservedDBInstanceId": "myreservedinstance",
      "ReservedDBInstancesOfferingId": "12ab34cd-59af-4b2c-a660-1abcdef23456",
      "DBInstanceClass": "db.t3.micro",
      "StartTime": "2020-06-01T13:44:21.436Z",
      "Duration": 31536000,
      "FixedPrice": 0.0,

```

```

    "UsagePrice": 0.0,
    "CurrencyCode": "USD",
    "DBInstanceCount": 1,
    "ProductDescription": "sqlserver-ex(li)",
    "OfferingType": "No Upfront",
    "MultiAZ": false,
    "State": "payment-pending",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": 0.014,
        "RecurringChargeFrequency": "Hourly"
      }
    ],
    "ReservedDBInstanceArn": "arn:aws:rds:us-
west-2:123456789012:ri:myreservedinstance",
    "LeaseId": "a1b2c3d4-6b69-4a59-be89-5e11aa446666"
  }
]
}

```

Para obtener más información, consulte [Instancias de bases de datos reservadas para Amazon RDS](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DescribeReservedDbInstances](#) la Referencia de AWS CLI comandos.

## describe-source-regions

En el siguiente ejemplo de código, se muestra cómo usar `describe-source-regions`.

### AWS CLI

Para describir las regiones de origen

El siguiente `describe-source-regions` ejemplo recupera detalles sobre todas las AWS regiones de origen. También muestra que las copias de seguridad automatizadas solo se pueden replicar desde EE. UU. Oeste (Oregón) hasta la AWS región de destino, EE. UU. Este (Norte de Virginia).

```

aws rds describe-source-regions \
  --region us-east-1

```

**Salida:**

```
{
  "SourceRegions": [
    {
      "RegionName": "af-south-1",
      "Endpoint": "https://rds.af-south-1.amazonaws.com",
      "Status": "available",
      "SupportsDBInstanceAutomatedBackupsReplication": false
    },
    {
      "RegionName": "ap-east-1",
      "Endpoint": "https://rds.ap-east-1.amazonaws.com",
      "Status": "available",
      "SupportsDBInstanceAutomatedBackupsReplication": false
    },
    {
      "RegionName": "ap-northeast-1",
      "Endpoint": "https://rds.ap-northeast-1.amazonaws.com",
      "Status": "available",
      "SupportsDBInstanceAutomatedBackupsReplication": true
    },
    {
      "RegionName": "ap-northeast-2",
      "Endpoint": "https://rds.ap-northeast-2.amazonaws.com",
      "Status": "available",
      "SupportsDBInstanceAutomatedBackupsReplication": true
    },
    {
      "RegionName": "ap-northeast-3",
      "Endpoint": "https://rds.ap-northeast-3.amazonaws.com",
      "Status": "available",
      "SupportsDBInstanceAutomatedBackupsReplication": false
    },
    {
      "RegionName": "ap-south-1",
      "Endpoint": "https://rds.ap-south-1.amazonaws.com",
      "Status": "available",
      "SupportsDBInstanceAutomatedBackupsReplication": true
    },
    {
      "RegionName": "ap-southeast-1",
      "Endpoint": "https://rds.ap-southeast-1.amazonaws.com",
      "Status": "available",

```

```
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "ap-southeast-2",
    "Endpoint": "https://rds.ap-southeast-2.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "ap-southeast-3",
    "Endpoint": "https://rds.ap-southeast-3.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": false
  },
  {
    "RegionName": "ca-central-1",
    "Endpoint": "https://rds.ca-central-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "eu-north-1",
    "Endpoint": "https://rds.eu-north-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "eu-south-1",
    "Endpoint": "https://rds.eu-south-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": false
  },
  {
    "RegionName": "eu-west-1",
    "Endpoint": "https://rds.eu-west-1.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  {
    "RegionName": "eu-west-2",
    "Endpoint": "https://rds.eu-west-2.amazonaws.com",
    "Status": "available",
    "SupportsDBInstanceAutomatedBackupsReplication": true
  },
  },
}
```

```
{
  "RegionName": "eu-west-3",
  "Endpoint": "https://rds.eu-west-3.amazonaws.com",
  "Status": "available",
  "SupportsDBInstanceAutomatedBackupsReplication": true
},
{
  "RegionName": "me-central-1",
  "Endpoint": "https://rds.me-central-1.amazonaws.com",
  "Status": "available",
  "SupportsDBInstanceAutomatedBackupsReplication": false
},
{
  "RegionName": "me-south-1",
  "Endpoint": "https://rds.me-south-1.amazonaws.com",
  "Status": "available",
  "SupportsDBInstanceAutomatedBackupsReplication": false
},
{
  "RegionName": "sa-east-1",
  "Endpoint": "https://rds.sa-east-1.amazonaws.com",
  "Status": "available",
  "SupportsDBInstanceAutomatedBackupsReplication": true
},
{
  "RegionName": "us-east-2",
  "Endpoint": "https://rds.us-east-2.amazonaws.com",
  "Status": "available",
  "SupportsDBInstanceAutomatedBackupsReplication": true
},
{
  "RegionName": "us-west-1",
  "Endpoint": "https://rds.us-west-1.amazonaws.com",
  "Status": "available",
  "SupportsDBInstanceAutomatedBackupsReplication": true
},
{
  "RegionName": "us-west-2",
  "Endpoint": "https://rds.us-west-2.amazonaws.com",
  "Status": "available",
  "SupportsDBInstanceAutomatedBackupsReplication": true
}
]
```

```
}
```

Para obtener más información, consulte [Búsqueda de información sobre copias de seguridad replicadas](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [DescribeSourceRegions](#) la Referencia de AWS CLI comandos.

## **describe-valid-db-instance-modifications**

En el siguiente ejemplo de código, se muestra cómo usar `describe-valid-db-instance-modifications`.

### AWS CLI

Para describir las modificaciones válidas de una instancia de base de datos

El siguiente `describe-valid-db-instance-modifications` ejemplo recupera detalles sobre las modificaciones válidas para la instancia de base de datos especificada.

```
aws rds describe-valid-db-instance-modifications \  
--db-instance-identifier test-instance
```

Salida:

```
{  
  "ValidDBInstanceModificationsMessage": {  
    "ValidProcessorFeatures": [],  
    "Storage": [  
      {  
        "StorageSize": [  
          {  
            "Step": 1,  
            "To": 20,  
            "From": 20  
          },  
          {  
            "Step": 1,  
            "To": 6144,  
            "From": 22  
          }  
        ]  
      }  
    ],  
  }  
}
```



```
    "ProvisionedIops": [
      {
        "Step": 1,
        "To": 0,
        "From": 0
      }
    ],
    "IopsToStorageRatio": [
      {
        "To": 0.0,
        "From": 0.0
      }
    ],
    "StorageType": "gp2"
  },
  {
    "StorageSize": [
      {
        "Step": 1,
        "To": 6144,
        "From": 100
      }
    ],
    "ProvisionedIops": [
      {
        "Step": 1,
        "To": 40000,
        "From": 1000
      }
    ],
    "IopsToStorageRatio": [
      {
        "To": 50.0,
        "From": 1.0
      }
    ],
    "StorageType": "io1"
  },
  {
    "StorageSize": [
      {
        "Step": 1,
        "To": 20,
        "From": 20
      }
    ]
  }
}
```

```

        },
        {
            "Step": 1,
            "To": 3072,
            "From": 22
        }
    ],
    "ProvisionedIops": [
        {
            "Step": 1,
            "To": 0,
            "From": 0
        }
    ],
    "IopsToStorageRatio": [
        {
            "To": 0.0,
            "From": 0.0
        }
    ],
    "StorageType": "magnetic"
}
]
}
}

```

- Para API obtener más información, consulte [DescribeValidDbInstanceModifications](#) la Referencia de AWS CLI comandos.

## download-db-log-file-portion

En el siguiente ejemplo de código, se muestra cómo usar `download-db-log-file-portion`.

### AWS CLI

Para descargar un archivo de registro de base de datos

El siguiente `download-db-log-file-portion` ejemplo descarga solo la parte más reciente del archivo de registro y la guarda en un archivo local denominado `tail.txt`.

```

aws rds download-db-log-file-portion \
  --db-instance-identifier test-instance \

```

```
--log-file-name log.txt \  
--output text > tail.txt
```

Para descargar el archivo completo, debe incluir el `--starting-token 0` parámetro. El siguiente ejemplo guarda el resultado en un archivo local denominado `full.txt`.

```
aws rds download-db-log-file-portion \  
--db-instance-identifier test-instance \  
--log-file-name log.txt \  
--starting-token 0 \  
--output text > full.txt
```

El archivo guardado puede contener líneas en blanco. Aparecen al final de cada parte del archivo de registro durante la descarga. Por lo general, esto no causa ningún problema en el análisis del archivo de registro.

- Para API obtener más información, consulte [DownloadDbLogFilePortion](#) la Referencia de AWS CLI comandos.

## generate-auth-token

En el siguiente ejemplo de código, se muestra cómo usar `generate-auth-token`.

### AWS CLI

Para generar un token de autenticación

El siguiente `generate-db-auth-token` ejemplo genera un token de autenticación para usarlo con la autenticación de la IAM base de datos.

```
aws rds generate-db-auth-token \  
--hostname aurmysql-test.cdgmuqiadpid.us-west-2.rds.amazonaws.com \  
--port 3306 \  
--region us-east-1 \  
--username jane_doe
```

Salida:

```
aurmysql-test.cdgmuqiadpid.us-west-2.rds.amazonaws.com:3306/?  
Action=connect&DBUser=jane_doe&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-  
Credential=AKIAIESZCNJ30EXAMPLE%2F20180731%2Fus-east-1%2Frds-db%2Faws4_request&X-
```

```
Amz-Date=20180731T235209Z&X-Amz-Expires=900&X-Amz-SignedHeaders=host&X-Amz-Signature=5a8753ebEXAMPLEa2c724e5667797EXAMPLE9d6ec6e3f427191fa41aeEXAMPLE
```

- Para API obtener más información, consulte [GenerateAuthToken](#) la Referencia de AWS CLI comandos.

## generate-db-auth-token

En el siguiente ejemplo de código, se muestra cómo usar `generate-db-auth-token`.

### AWS CLI

Para generar un token IAM de autenticación

El siguiente `generate-db-auth-token` ejemplo genera un token de IAM autenticación para conectarse a una base de datos.

```
aws rds generate-db-auth-token \  
  --hostname mydb.123456789012.us-east-1.rds.amazonaws.com \  
  --port 3306 \  
  --region us-east-1 \  
  --username db_user
```

Salida:

```
mydb.123456789012.us-east-1.rds.amazonaws.com:3306/?  
Action=connect&DBUser=db_user&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIEXAMPLE%2Fus-east-1%2Frds-db%2Faws4_request&X-Amz-Date=20210123T011543Z&X-Amz-Expires=900&X-Amz-SignedHeaders=host&X-Amz-Signature=88987EXAMPLE1EXAMPLE2EXAMPLE3EXAMPLE4EXAMPLE5EXAMPLE6
```

Para obtener más información, consulte [Conexión a su instancia de base de datos mediante IAM autenticación](#) en la Guía del RDS usuario de Amazon y [Conexión a su clúster de base de datos mediante IAM autenticación](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [GenerateDbAuthToken](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

## AWS CLI

Para enumerar etiquetas en un RDS recurso de Amazon

En el siguiente `list-tags-for-resource` ejemplo, se enumeran todas las etiquetas de una instancia de base de datos.

```
aws rds list-tags-for-resource \  
  --resource-name arn:aws:rds:us-east-1:123456789012:db:orc11
```

Salida:

```
{  
  "TagList": [  
    {  
      "Key": "Environment",  
      "Value": "test"  
    },  
    {  
      "Key": "Name",  
      "Value": "MyDatabase"  
    }  
  ]  
}
```

Para obtener más información, consulta Cómo [etiquetar RDS los recursos de Amazon](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## **modify-certificates**

En el siguiente ejemplo de código, se muestra cómo usar `modify-certificates`.

## AWS CLI

Para anular temporalmente el TLS certificadoSSL/predeterminado del sistema para las nuevas instancias de base de datos

El siguiente `modify-certificates` ejemplo anula temporalmente el certificadoSSL/TLSpredeterminado del sistema para las nuevas instancias de base de datos.

```
aws rds modify-certificates \  
  --certificate-identifier rds-ca-2019
```

Salida:

```
{  
  "Certificate": {  
    "CertificateIdentifier": "rds-ca-2019",  
    "CertificateType": "CA",  
    "Thumbprint": "EXAMPLE123456789012",  
    "ValidFrom": "2019-09-19T18:16:53Z",  
    "ValidTill": "2024-08-22T17:08:50Z",  
    "CertificateArn": "arn:aws:rds:us-east-1::cert:rds-ca-2019",  
    "CustomerOverride": true,  
    "CustomerOverrideValidTill": "2024-08-22T17:08:50Z"  
  }  
}
```

Para obtener más información, [consulte Rotación del TLS certificadoSSL](#) en la Guía del RDS usuario de Amazon y [Rotación del TLS certificadoSSL](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ModifyCertificates](#) la Referencia de AWS CLI comandos.

## modify-current-db-cluster-capacity

En el siguiente ejemplo de código, se muestra cómo usar `modify-current-db-cluster-capacity`.

AWS CLI

Para escalar la capacidad de un clúster de base de datos Aurora Serverless

El siguiente `modify-current-db-cluster-capacity` ejemplo escala la capacidad de un clúster de base de datos Aurora Serverless a 8.

```
aws rds modify-current-db-cluster-capacity \  
  --db-cluster-identifier mydbcluster \  
  --capacity 8
```

Salida:

```
{
  "DBClusterIdentifier": "mydbcluster",
  "PendingCapacity": 8,
  "CurrentCapacity": 1,
  "SecondsBeforeTimeout": 300,
  "TimeoutAction": "ForceApplyCapacityChange"
}
```

Para obtener más información, consulte [Escalar manualmente la capacidad del clúster de base de datos Aurora Serverless v1](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ModifyCurrentDbClusterCapacity](#) la Referencia de AWS CLI comandos.

## modify-db-cluster-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-cluster-endpoint`.

### AWS CLI

Para modificar un punto final de clúster de base de datos personalizado

El siguiente `modify-db-cluster-endpoint` ejemplo modifica el punto final del clúster de base de datos personalizado especificado.

```
aws rds modify-db-cluster-endpoint \
  --db-cluster-endpoint-identifier mycustomendpoint \
  --static-members dbinstance1 dbinstance2 dbinstance3
```

Salida:

```
{
  "DBClusterEndpointIdentifier": "mycustomendpoint",
  "DBClusterIdentifier": "mydbcluster",
  "DBClusterEndpointResourceIdentifier": "cluster-endpoint-ANPAJ4AE5446DAEXAMPLE",
  "Endpoint": "mycustomendpoint.cluster-custom-cnpxample.us-east-1.rds.amazonaws.com",
  "Status": "modifying",
  "EndpointType": "CUSTOM",
  "CustomEndpointType": "READER",
  "StaticMembers": [
```

```

        "dbinstance1",
        "dbinstance2",
        "dbinstance3"
    ],
    "ExcludedMembers": [],
    "DBClusterEndpointArn": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:mycustomendpoint"
}

```

Para obtener más información, consulte [Amazon Aurora Connection Management](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ModifyDbClusterEndpoint](#) la Referencia de AWS CLI comandos.

## modify-db-cluster-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-cluster-parameter-group`.

### AWS CLI

Para modificar los parámetros de un grupo de parámetros de un clúster de base de datos

El siguiente `modify-db-cluster-parameter-group` ejemplo modifica los valores de los parámetros de un grupo de parámetros de un clúster de base de datos.

```

aws rds modify-db-cluster-parameter-group \
  --db-cluster-parameter-group-name mydbclusterpg \
  --
parameters "ParameterName=server_audit_logging,ParameterValue=1,ApplyMethod=immediate"
\
ParameterName=server_audit_logs_upload,ParameterValue=1,ApplyMethod=immediate

```

Salida:

```

{
  "DBClusterParameterGroupName": "mydbclusterpg"
}

```



Para obtener más información, consulte [Trabajar con grupos de parámetros de bases de datos y grupos de parámetros de clústeres de bases de datos](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ModifyDbClusterParameterGroup](#) la Referencia de AWS CLI comandos.

## modify-db-cluster-snapshot-attribute

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-cluster-snapshot-attribute`.

### AWS CLI

Para modificar un atributo de instantánea de un clúster de base de datos

En el siguiente `modify-db-cluster-snapshot-attribute` ejemplo, se realizan cambios en el atributo de instantánea del clúster de base de datos especificado.

```
aws rds modify-db-cluster-snapshot-attribute \  
  --db-cluster-snapshot-identifier myclustersnapshot \  
  --attribute-name restore \  
  --values-to-add 123456789012
```

Salida:

```
{  
  "DBClusterSnapshotAttributesResult": {  
    "DBClusterSnapshotIdentifier": "myclustersnapshot",  
    "DBClusterSnapshotAttributes": [  
      {  
        "AttributeName": "restore",  
        "AttributeValues": [  
          "123456789012"  
        ]  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [Restauración desde una instantánea de un clúster de base de datos](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ModifyDbClusterSnapshotAttribute](#) la Referencia de AWS CLI comandos.

## modify-db-cluster

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-cluster`.

### AWS CLI

Ejemplo 1: Para modificar un clúster de base de datos

El siguiente `modify-db-cluster` ejemplo cambia la contraseña del usuario maestro del clúster de base de datos denominado `cluster-2` y establece el período de retención de la copia de seguridad en 14 días. El `--apply-immediately` parámetro hace que los cambios se realicen de forma inmediata, en lugar de esperar hasta el siguiente período de mantenimiento.

```
aws rds modify-db-cluster \  
  --db-cluster-identifier cluster-2 \  
  --backup-retention-period 14 \  
  --master-user-password newpassword99 \  
  --apply-immediately
```

Salida:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 1,  
    "AvailabilityZones": [  
      "eu-central-1b",  
      "eu-central-1c",  
      "eu-central-1a"  
    ],  
    "BackupRetentionPeriod": 14,  
    "DatabaseName": "",  
    "DBClusterIdentifier": "cluster-2",  
    "DBClusterParameterGroup": "default.aurora5.6",  
    "DBSubnetGroup": "default-vpc-2305ca49",  
    "Status": "available",  
    "EarliestRestorableTime": "2020-06-03T02:07:29.637Z",  
    "Endpoint": "cluster-2.cluster-#####.eu-central-1.rds.amazonaws.com",  
    "ReaderEndpoint": "cluster-2.cluster-ro-#####.eu-  
central-1.rds.amazonaws.com",
```

```

    "MultiAZ": false,
    "Engine": "aurora",
    "EngineVersion": "5.6.10a",
    "LatestRestorableTime": "2020-06-04T15:11:25.748Z",
    "Port": 3306,
    "MasterUsername": "admin",
    "PreferredBackupWindow": "01:55-02:25",
    "PreferredMaintenanceWindow": "thu:21:14-thu:21:44",
    "ReadReplicaIdentifiers": [],
    "DBClusterMembers": [
      {
        "DBInstanceIdentifier": "cluster-2-instance-1",
        "IsClusterWriter": true,
        "DBClusterParameterGroupStatus": "in-sync",
        "PromotionTier": 1
      }
    ],
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-20a5c047",
        "Status": "active"
      }
    ],
    "HostedZoneId": "Z1RLNU0EXAMPLE",
    "StorageEncrypted": true,
    "KmsKeyId": "arn:aws:kms:eu-central-1:123456789012:key/d1bd7c8f-5cdb-49ca-8a62-a1b2c3d4e5f6",
    "DbClusterResourceId": "cluster-AGJ7XI77XVIS6FUXHU1EXAMPLE",
    "DBClusterArn": "arn:aws:rds:eu-central-1:123456789012:cluster:cluster-2",
    "AssociatedRoles": [],
    "IAMDatabaseAuthenticationEnabled": false,
    "ClusterCreateTime": "2020-04-03T14:44:02.764Z",
    "EngineMode": "provisioned",
    "DeletionProtection": false,
    "HttpEndpointEnabled": false,
    "CopyTagsToSnapshot": true,
    "CrossAccountClone": false,
    "DomainMemberships": []
  }
}

```

Para obtener más información, consulte [Modificación de un clúster de base de datos de Amazon Aurora](#) en la Guía del usuario de Amazon Aurora.

## Ejemplo 2: Para asociar un grupo VPC de seguridad a un clúster de base de datos

El siguiente `modify-db-instance` ejemplo asocia un grupo de VPC seguridad específico y elimina los grupos de seguridad de base de datos de un clúster de base de datos.

```
aws rds modify-db-cluster \  
  --db-cluster-identifier dbName \  
  --vpc-security-group-ids sg-ID
```

Salida:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 1,  
    "AvailabilityZones": [  
      "us-west-2c",  
      "us-west-2b",  
      "us-west-2a"  
    ],  
    "BackupRetentionPeriod": 1,  
    "DBClusterIdentifier": "dbName",  
    "DBClusterParameterGroup": "default.aurora-mysql8.0",  
    "DBSubnetGroup": "default",  
    "Status": "available",  
    "EarliestRestorableTime": "2024-02-15T01:12:13.966000+00:00",  
    "Endpoint": "dbName.cluster-abcdefghji.us-west-2.rds.amazonaws.com",  
    "ReaderEndpoint": "dbName.cluster-ro-abcdefghji.us-  
west-2.rds.amazonaws.com",  
    "MultiAZ": false,  
    "Engine": "aurora-mysql",  
    "EngineVersion": "8.0.mysql_aurora.3.04.1",  
    "LatestRestorableTime": "2024-02-15T02:25:33.696000+00:00",  
    "Port": 3306,  
    "MasterUsername": "admin",  
    "PreferredBackupWindow": "10:59-11:29",  
    "PreferredMaintenanceWindow": "thu:08:54-thu:09:24",  
    "ReadReplicaIdentifiers": [],  
    "DBClusterMembers": [  
      {  
        "DBInstanceIdentifier": "dbName-instance-1",  
        "IsClusterWriter": true,  
        "DBClusterParameterGroupStatus": "in-sync",  
        "PromotionTier": 1
```

```

    }
  ],
  "VpcSecurityGroups": [
    {
      "VpcSecurityGroupId": "sg-ID",
      "Status": "active"
    }
  ],
  ...output omitted...
}
}

```

Para obtener más información, consulte [Control del acceso con grupos de seguridad](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ModifyDbCluster](#) la Referencia de AWS CLI comandos.

## modify-db-instance

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-instance`.

### AWS CLI

Ejemplo 1: para modificar una instancia de base de datos

El siguiente `modify-db-instance` ejemplo asocia un grupo de opciones y un grupo de parámetros a una instancia de base de datos de Microsoft SQL Server compatible. El parámetro `--apply-immediately` hace que los grupos de opciones y parámetros se asocien de inmediato, en lugar de esperar hasta el siguiente período de mantenimiento.

```

aws rds modify-db-instance \
  --db-instance-identifier database-2 \
  --option-group-name test-se-2017 \
  --db-parameter-group-name test-sqlserver-se-2017 \
  --apply-immediately

```

Salida:

```

{
  "DBInstance": {

```

```
"DBInstanceIdentifier": "database-2",
"DBInstanceClass": "db.r4.large",
"Engine": "sqlserver-se",
"DBInstanceStatus": "available",

...output omitted...

"DBParameterGroups": [
  {
    "DBParameterGroupName": "test-sqlserver-se-2017",
    "ParameterApplyStatus": "applying"
  }
],
"AvailabilityZone": "us-west-2d",

...output omitted...

"MultiAZ": true,
"EngineVersion": "14.00.3281.6.v1",
"AutoMinorVersionUpgrade": false,
"ReadReplicaDBInstanceIdentifiers": [],
"LicenseModel": "license-included",
"OptionGroupMemberships": [
  {
    "OptionGroupName": "test-se-2017",
    "Status": "pending-apply"
  }
],
"CharacterSetName": "SQL_Latin1_General_CP1_CI_AS",
"SecondaryAvailabilityZone": "us-west-2c",
"PubliclyAccessible": true,
"StorageType": "gp2",

...output omitted...

"DeletionProtection": false,
"AssociatedRoles": [],
"MaxAllocatedStorage": 1000
}
}
```

Para obtener más información, consulte [Modificación de una RDS instancia](#) de base de datos de Amazon en la Guía del RDS usuario de Amazon.

## Ejemplo 2: Para asociar un grupo VPC de seguridad a una instancia de base de datos

El siguiente `modify-db-instance` ejemplo asocia un grupo de VPC seguridad específico y elimina los grupos de seguridad de base de datos de una instancia de base de datos:

```
aws rds modify-db-instance \  
  --db-instance-identifier dbName \  
  --vpc-security-group-ids sg-ID
```

Salida:

```
{  
  "DBInstance": {  
    "DBInstanceIdentifier": "dbName",  
    "DBInstanceClass": "db.t3.micro",  
    "Engine": "mysql",  
    "DBInstanceStatus": "available",  
    "MasterUsername": "admin",  
    "Endpoint": {  
      "Address": "dbName.abcdefghijkl.us-west-2.rds.amazonaws.com",  
      "Port": 3306,  
      "HostedZoneId": "ABCDEFGHIJK1234"  
    },  
    "AllocatedStorage": 20,  
    "InstanceCreateTime": "2024-02-15T00:37:58.793000+00:00",  
    "PreferredBackupWindow": "11:57-12:27",  
    "BackupRetentionPeriod": 7,  
    "DBSecurityGroups": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sg-ID",  
        "Status": "active"  
      }  
    ],  
    ... output omitted ...  
    "MultiAZ": false,  
    "EngineVersion": "8.0.35",  
    "AutoMinorVersionUpgrade": true,  
    "ReadReplicaDBInstanceIdentifiers": [],  
    "LicenseModel": "general-public-license",  
    ... output omitted ...  
  }  
}
```

```
}
```

Para obtener más información, consulte [Controlar el acceso con grupos de seguridad](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [ModifyDBInstance](#) en la referencia de AWS CLI comandos.

## **modify-db-parameter-group**

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-parameter-group`.

### AWS CLI

Modificación de un grupo de parámetros de base de datos

En el siguiente ejemplo `modify-db-parameter-group`, se cambia el valor del parámetro `clr_enabled` en un grupo de parámetros de base de datos. El parámetro `--apply-immediately` hace que el grupo de parámetros de la base de datos se modifique de inmediato, en lugar de esperar hasta el siguiente período de mantenimiento.

```
aws rds modify-db-parameter-group \  
  --db-parameter-group-name test-sqlserver-se-2017 \  
  --parameters "ParameterName='clr_enabled',ParameterValue=1,ApplyMethod=immediate"
```

Salida:

```
{  
  "DBParameterGroupName": "test-sqlserver-se-2017"  
}
```

Para obtener más información, consulte [Modificación de parámetros en un grupo de parámetros de base de datos](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [ModifyDBParameter Group](#) en la referencia de AWS CLI comandos.

## **modify-db-proxy-endpoint**

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-proxy-endpoint`.



## AWS CLI

Para modificar un punto final de proxy de base de datos para una RDS base de datos

El siguiente `modify-db-proxy-endpoint` ejemplo modifica un punto final de proxy de base de datos `proxyEndpoint` para establecer el tiempo de espera de lectura en 65 segundos.

```
aws rds modify-db-proxy-endpoint \  
  --db-proxy-endpoint-name proxyEndpoint \  
  --cli-read-timeout 65
```

Salida:

```
{  
  "DBProxyEndpoint":  
    {  
      "DBProxyEndpointName": "proxyEndpoint",  
      "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-  
endpoint:prx-endpoint-0123a01b12345c0ab",  
      "DBProxyName": "proxyExample",  
      "Status": "available",  
      "VpcId": "vpc-1234567",  
      "VpcSecurityGroupIds": [  
        "sg-1234"  
      ],  
      "VpcSubnetIds": [  
        "subnetgroup1",  
        "subnetgroup2"  
      ],  
      "Endpoint": "proxyEndpoint.endpoint.proxyExample-ab0cd1efghij.us-  
east-1.rds.amazonaws.com",  
      "CreateDate": "2023-04-05T16:09:33.452000+00:00",  
      "TargetRole": "READ_WRITE",  
      "IsDefault": "false"  
    }  
}
```

Para obtener más información, consulte [Modificación de un punto final proxy](#) en la Guía del RDS usuario de Amazon y [Modificación de un punto final proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ModifyDbProxyEndpoint](#) la Referencia de AWS CLI comandos.

## modify-db-proxy-target-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-proxy-target-group`.

### AWS CLI

Para modificar los puntos finales de un proxy de base de datos

El siguiente `modify-db-proxy-target-group` ejemplo modifica un grupo objetivo de proxy de base de datos para establecer el número máximo de conexiones en un 80 por ciento y el máximo de conexiones inactivas en un 10 por ciento.

```
aws rds modify-db-proxy-target-group \  
  --target-group-name default \  
  --db-proxy-name proxyExample \  
  --connection-pool-config MaxConnectionsPercent=80,MaxIdleConnectionsPercent=10
```

Salida:

```
{  
  "DBProxyTargetGroup":  
    {  
      "DBProxyName": "proxyExample",  
      "TargetGroupName": "default",  
      "TargetGroupArn": "arn:aws:rds:us-east-1:123456789012:target-group:prx-  
tg-0123a01b12345c0ab",  
      "IsDefault": true,  
      "Status": "available",  
      "ConnectionPoolConfig": {  
        "MaxConnectionsPercent": 80,  
        "MaxIdleConnectionsPercent": 10,  
        "ConnectionBorrowTimeout": 120,  
        "SessionPinningFilters": []  
      },  
      "CreateDate": "2023-05-02T18:41:19.495000+00:00",  
      "UpdateDate": "2023-05-02T18:41:21.762000+00:00"  
    }  
}
```

Para obtener más información, consulte [Modificación de un RDS proxy](#) en la Guía del RDS usuario de Amazon y [Modificación de un RDS proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ModifyDbProxyTargetGroup](#) la Referencia de AWS CLI comandos.

## modify-db-proxy

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-proxy`.

### AWS CLI

Para modificar un proxy de base de datos para una RDS base de datos

En el siguiente `modify-db-proxy` ejemplo, se modifica un proxy de base de datos cuyo nombre es obligatorio `proxyExample` SSL para sus conexiones.

```
aws rds modify-db-proxy \  
  --db-proxy-name proxyExample \  
  --require-tls
```

Salida:

```
{  
  "DBProxy":  
    {  
      "DBProxyName": "proxyExample",  
      "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-  
proxy:prx-0123a01b12345c0ab",  
      "Status": "modifying"  
      "EngineFamily": "PostgreSQL",  
      "VpcId": "sg-1234567",  
      "VpcSecurityGroupIds": [  
        "sg-1234"  
      ],  
      "VpcSubnetIds": [  
        "subnetgroup1",  
        "subnetgroup2"  
      ],  
      "Auth": "[  
        {  
          "Description": "proxydescription1",  
          "AuthScheme": "SECRETS",  
          "SecretArn": "arn:aws:secretsmanager:us-  
west-2:123456789123:secret:proxysecret1-Abcd1e",
```

```

        "IAMAuth": "DISABLED"
      }
    ],
    "RoleArn": "arn:aws:iam::12345678912:role/ProxyPostgreSQLRole",
    "Endpoint": "proxyExample.proxy-ab0cd1efghij.us-east-1.rds.amazonaws.com",
    "RequireTLS": true,
    "IdleClientTimeout": 1800,
    "DebuggingLogging": false,
    "CreateDate": "2023-04-05T16:09:33.452000+00:00",
    "UpdatedAt": "2023-04-13T01:49:38.568000+00:00"
  }
}

```

Para obtener más información, consulte [Modificación de un RDS proxy](#) en la Guía del RDS usuario de Amazon y [Creación de un RDS proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ModifyDbProxy](#) la Referencia de AWS CLI comandos.

## modify-db-shard-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-shard-group`.

### AWS CLI

Ejemplo 1: Para modificar un grupo de fragmentos de base de datos

El siguiente `modify-db-shard-group` ejemplo cambia la capacidad máxima de un grupo de particiones de base de datos.

```

aws rds modify-db-shard-group \
  --db-shard-group-identifier my-db-shard-group \
  --max-acu 1000

```

Salida:

```

{
  "DBShardGroups": [
    {
      "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
      "DBShardGroupIdentifier": "my-db-shard-group",
      "DBClusterIdentifier": "my-sv2-cluster",

```

```

        "MaxACU": 768.0,
        "ComputeRedundancy": 0,
        "Status": "available",
        "PubliclyAccessible": false,
        "Endpoint": "my-sv2-cluster.limitless-cekyceexample.us-
east-2.rds.amazonaws.com"
    }
]
}

```

Para obtener más información, consulte los [clústeres de base](#) de datos de Amazon Aurora en la Guía del usuario de Amazon Aurora.

Ejemplo 2: Para describir sus grupos de fragmentos de base de datos

En el siguiente `describe-db-shard-groups` ejemplo, se recuperan los detalles de los grupos de fragmentos de base de datos después de ejecutar el comando `modify-db-shard-group`. La capacidad máxima del grupo de particiones de base de datos `my-db-shard-group` es ahora de 1000 unidades de capacidad Aurora (ACUs).

```
aws rds describe-db-shard-groups
```

Salida:

```

{
  "DBShardGroups": [
    {
      "DBShardGroupResourceId": "shardgroup-7bb446329da94788b3f957746example",
      "DBShardGroupIdentifier": "limitless-test-shard-grp",
      "DBClusterIdentifier": "limitless-test-cluster",
      "MaxACU": 768.0,
      "ComputeRedundancy": 0,
      "Status": "available",
      "PubliclyAccessible": true,
      "Endpoint": "limitless-test-cluster.limitless-cekyceexample.us-
east-2.rds.amazonaws.com"
    },
    {
      "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
      "DBShardGroupIdentifier": "my-db-shard-group",
      "DBClusterIdentifier": "my-sv2-cluster",
      "MaxACU": 1000.0,

```

```

        "ComputeRedundancy": 0,
        "Status": "available",
        "PubliclyAccessible": false,
        "Endpoint": "my-sv2-cluster.limitless-cekyceexample.us-
east-2.rds.amazonaws.com"
    }
]
}

```

Para obtener más información, consulte los [clústeres de base](#) de datos de Amazon Aurora en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ModifyDbShardGroup](#) la Referencia de AWS CLI comandos.

## modify-db-snapshot-attribute

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-snapshot-attribute`.

### AWS CLI

Ejemplo 1: Para permitir que dos AWS cuentas restauren una instantánea de base de datos

El siguiente `modify-db-snapshot-attribute` ejemplo concede permiso a dos AWS cuentas, con los identificadores 111122223333 y 444455556666, para restaurar la instantánea de base de datos con un nombre `mydbsnapshot`.

```

aws rds modify-db-snapshot-attribute \
  --db-snapshot-identifier mydbsnapshot \
  --attribute-name restore \
  --values-to-add {"111122223333","444455556666"}

```

Salida:

```

{
  "DBSnapshotAttributesResult": {
    "DBSnapshotIdentifier": "mydbsnapshot",
    "DBSnapshotAttributes": [
      {
        "AttributeName": "restore",
        "AttributeValues": [

```

```

        "111122223333",
        "444455556666"
    ]
}
]
}
}

```

Para obtener más información, consulta [Compartir una instantánea](#) en la Guía del RDS usuario de Amazon.

Ejemplo 2: Para evitar que una AWS cuenta restaure una instantánea de base de datos

El siguiente `modify-db-snapshot-attribute` ejemplo elimina el permiso de una AWS cuenta concreta para restaurar la instantánea de base de datos denominada `mydbsnapshot`. Al especificar una sola cuenta, el identificador de la cuenta no puede estar entre comillas o corchetes.

```

aws rds modify-db-snapshot-attribute \
  --db-snapshot-identifier mydbsnapshot \
  --attribute-name restore \
  --values-to-remove 444455556666

```

Salida:

```

{
  "DBSnapshotAttributesResult": {
    "DBSnapshotIdentifier": "mydbsnapshot",
    "DBSnapshotAttributes": [
      {
        "AttributeName": "restore",
        "AttributeValues": [
          "111122223333"
        ]
      }
    ]
  }
}

```

Para obtener más información, consulta [Compartir una instantánea](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [ModifyDbSnapshotAttribute](#) la Referencia de AWS CLI comandos.

## modify-db-snapshot-attributes

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-snapshot-attributes`.

### AWS CLI

Para modificar un atributo de instantánea de base de datos

El siguiente `modify-db-snapshot-attribute` ejemplo permite utilizar dos identificadores de AWS cuenta 111122223333 y 444455556666 restaurar la instantánea de base de datos denominada `mydbsnapshot`.

```
aws rds modify-db-snapshot-attribute \  
  --db-snapshot-identifier mydbsnapshot \  
  --attribute-name restore \  
  --values-to-add '["111122223333", "444455556666"]'
```

Salida:

```
{  
  "DBSnapshotAttributesResult": {  
    "DBSnapshotIdentifier": "mydbsnapshot",  
    "DBSnapshotAttributes": [  
      {  
        "AttributeName": "restore",  
        "AttributeValues": [  
          "111122223333",  
          "444455556666"  
        ]  
      }  
    ]  
  }  
}
```

Para obtener más información, consulta [Compartir una instantánea](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [ModifyDbSnapshotAttributes](#) la Referencia de AWS CLI comandos.



## modify-db-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-snapshot`.

### AWS CLI

Para modificar una instantánea de base de datos

En el siguiente `modify-db-snapshot` ejemplo, se actualiza una instantánea de PostgreSQL 10.6 denominada `SQL Postgre db5-snapshot-upg-test` 11.7. La nueva versión del motor de base de datos se muestra cuando la instantánea ha terminado de actualizarse y su estado está disponible.

```
aws rds modify-db-snapshot \  
  --db-snapshot-identifier db5-snapshot-upg-test \  
  --engine-version 11.7
```

Salida:

```
{  
  "DBSnapshot": {  
    "DBSnapshotIdentifier": "db5-snapshot-upg-test",  
    "DBInstanceIdentifier": "database-5",  
    "SnapshotCreateTime": "2020-03-27T20:49:17.092Z",  
    "Engine": "postgres",  
    "AllocatedStorage": 20,  
    "Status": "upgrading",  
    "Port": 5432,  
    "AvailabilityZone": "us-west-2a",  
    "VpcId": "vpc-2ff27557",  
    "InstanceCreateTime": "2020-03-27T19:59:04.735Z",  
    "MasterUsername": "postgres",  
    "EngineVersion": "10.6",  
    "LicenseModel": "postgresql-license",  
    "SnapshotType": "manual",  
    "OptionGroupName": "default:postgres-11",  
    "PercentProgress": 100,  
    "StorageType": "gp2",  
    "Encrypted": false,  
    "DBSnapshotArn": "arn:aws:rds:us-west-2:123456789012:snapshot:db5-snapshot-upg-test",  
    "IAMDatabaseAuthenticationEnabled": false,  
    "ProcessorFeatures": [],
```

```

    "DbiResourceId": "db-GJMF75LM42IL6BTFRE4UZJ5YM4"
  }
}

```

Para obtener más información, consulte [Actualización de una instantánea de SQL base de datos de Postgre](#) en la Guía RDS del usuario de Amazon.

- Para API obtener más información, consulte la Referencia [ModifyDbSnapshot](#) de AWS CLI comandos.

## modify-db-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-db-subnet-group`.

### AWS CLI

Para modificar un grupo de subredes de base de datos

El siguiente `modify-db-subnet-group` ejemplo agrega una subred con el ID `subnet-08e41f9e230222222` al grupo de subredes de base de datos denominado `mysubnetgroup`. Para mantener las subredes existentes en el grupo de subredes, incluya sus valores IDs como en la opción `--subnet-ids`. Asegúrese de tener subredes con al menos dos zonas de disponibilidad diferentes en el grupo de subredes de base de datos.

```

aws rds modify-db-subnet-group \
  --db-subnet-group-name mysubnetgroup \
  --subnet-ids
  '["subnet-0a1dc4e1a6f123456", "subnet-070dd7ecb3aaaaaaa", "subnet-00f5b198bc0abcdef", "subnet-

```

Salida:

```

{
  "DBSubnetGroup": {
    "DBSubnetGroupName": "mysubnetgroup",
    "DBSubnetGroupDescription": "test DB subnet group",
    "VpcId": "vpc-0f08e7610a1b2c3d4",
    "SubnetGroupStatus": "Complete",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-08e41f9e230222222",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2a"
        }
      }
    ]
  }
}

```

```

    },
    "SubnetStatus": "Active"
  },
  {
    "SubnetIdentifier": "subnet-070dd7ecb3aaaaaaaa",
    "SubnetAvailabilityZone": {
      "Name": "us-west-2b"
    },
    "SubnetStatus": "Active"
  },
  {
    "SubnetIdentifier": "subnet-00f5b198bc0abcdef",
    "SubnetAvailabilityZone": {
      "Name": "us-west-2d"
    },
    "SubnetStatus": "Active"
  },
  {
    "SubnetIdentifier": "subnet-0a1dc4e1a6f123456",
    "SubnetAvailabilityZone": {
      "Name": "us-west-2b"
    },
    "SubnetStatus": "Active"
  }
],
  "DBSubnetGroupArn": "arn:aws:rds:us-
west-2:534026745191:subgrp:mysubnetgroup"
}
}

```

Para obtener más información, consulte el [paso 3: Crear un grupo de subredes de base](#) de datos en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [ModifyDbSubnetGroup](#) la Referencia de AWS CLI comandos.

## modify-event-subscription

En el siguiente ejemplo de código, se muestra cómo usar `modify-event-subscription`.

### AWS CLI

Para modificar una suscripción a un evento

El siguiente `modify-event-subscription` ejemplo deshabilita la suscripción al evento especificada, de modo que ya no publique notificaciones en el tema especificado de Amazon Simple Notification Service.

```
aws rds modify-event-subscription \  
  --subscription-name my-instance-events \  
  --no-enabled
```

Salida:

```
{  
  "EventSubscription": {  
    "EventCategoriesList": [  
      "backup",  
      "recovery"  
    ],  
    "CustomerAwsId": "123456789012",  
    "SourceType": "db-instance",  
    "SubscriptionCreationTime": "Tue Jul 31 23:22:01 UTC 2018",  
    "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-  
events",  
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",  
    "CustSubscriptionId": "my-instance-events",  
    "Status": "modifying",  
    "Enabled": false  
  }  
}
```

- Para API obtener más información, consulte [ModifyEventSubscription](#) la Referencia de AWS CLI comandos.

## **modify-global-cluster**

En el siguiente ejemplo de código, se muestra cómo usar `modify-global-cluster`.

### AWS CLI

Para modificar un clúster de base de datos global

El siguiente `modify-global-cluster` ejemplo habilita la protección contra la eliminación de un clúster de base SQL de datos global compatible con Aurora My.

```
aws rds modify-global-cluster \  
  --global-cluster-identifier myglobalcluster \  
  --deletion-protection
```

Salida:

```
{  
  "GlobalCluster": {  
    "GlobalClusterIdentifier": "myglobalcluster",  
    "GlobalClusterResourceId": "cluster-f0e523bfe07aabb",  
    "GlobalClusterArn": "arn:aws:rds::123456789012:global-  
cluster:myglobalcluster",  
    "Status": "available",  
    "Engine": "aurora-mysql",  
    "EngineVersion": "5.7.mysql_aurora.2.07.2",  
    "StorageEncrypted": false,  
    "DeletionProtection": true,  
    "GlobalClusterMembers": []  
  }  
}
```

Para obtener más información, consulte [Administrar una base de datos global de Aurora](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ModifyGlobalCluster](#) la Referencia de AWS CLI comandos.

## **promote-read-replica-db-cluster**

En el siguiente ejemplo de código, se muestra cómo usar `promote-read-replica-db-cluster`.

AWS CLI

Para promover un clúster de base de datos, lea réplica

El siguiente `promote-read-replica-db-cluster` ejemplo promueve que la réplica de lectura especificada se convierta en un clúster de base de datos independiente.

```
aws rds promote-read-replica-db-cluster \  
  --db-cluster-identifier mydbcluster-1
```

Salida:

```
{
  "DBCluster": {
    "AllocatedStorage": 1,
    "AvailabilityZones": [
      "us-east-1a",
      "us-east-1b",
      "us-east-1c"
    ],
    "BackupRetentionPeriod": 1,
    "DatabaseName": "",
    "DBClusterIdentifier": "mydbcluster-1",
    ...some output truncated...
  }
}
```

Para obtener más información, consulte Cómo [convertir una réplica de lectura en un clúster](#) de base de datos en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [PromoteReadReplicaDbCluster](#) la Referencia de AWS CLI comandos.

## promote-read-replica

En el siguiente ejemplo de código, se muestra cómo usar `promote-read-replica`.

### AWS CLI

Para promover una réplica de lectura

El siguiente `promote-read-replica` ejemplo promueve que la réplica de lectura especificada se convierta en una instancia de base de datos independiente.

```
aws rds promote-read-replica \
  --db-instance-identifier test-instance-repl
```

Salida:

```
{
  "DBInstance": {
    "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance-repl",
    "StorageType": "standard",
```

```
    "ReadReplicaSourceDBInstanceIdentifier": "test-instance",
    "DBInstanceStatus": "modifying",
    ...some output truncated...
  }
}
```

- Para API obtener más información, consulte [PromoteReadReplica](#) la Referencia de AWS CLI comandos.

## **purchase-reserved-db-instance**

En el siguiente ejemplo de código, se muestra cómo usar `purchase-reserved-db-instance`.

### AWS CLI

Para comprar una oferta de instancia de base de datos reservada

En el siguiente `purchase-reserved-db-instances-offering` ejemplo, se compra una oferta de instancia de base de datos reservada. `reserved-db-instances-offering-id` Debe ser un identificador de oferta válido, tal como lo devuelve el `describe-reserved-db-instances-offering` comando.

```
aws rds purchase-reserved-db-instances -offering -- reserved-db-instances-offering -id
438012d3-4a52-4cc7-b2e3-8dff72e0e706
```

- Para API obtener más información [PurchaseReservedDbInstance](#), AWS CLI consulte la Referencia de comandos.

## **purchase-reserved-db-instances-offerings**

En el siguiente ejemplo de código, se muestra cómo usar `purchase-reserved-db-instances-offerings`.

### AWS CLI

Ejemplo 1: Para buscar una instancia de base de datos reservada y comprarla

En el siguiente `describe-reserved-db-instances-offerings` ejemplo, se enumeran las instancias de My SQL DB reservadas disponibles con la clase de instancia `db.t2.micro` y una duración de un año. El ID de la oferta es obligatorio para comprar una instancia de base de datos reservada.

```
aws rds describe-reserved-db-instances-offerings \  
  --product-description mysql \  
  --db-instance-class db.t2.micro \  
  --duration 1
```

Salida:

```
{  
  "ReservedDBInstancesOfferings": [  
    {  
      "ReservedDBInstancesOfferingId": "8ba30be1-b9ec-447f-8f23-6114e3f4c7b4",  
      "DBInstanceClass": "db.t2.micro",  
      "Duration": 31536000,  
      "FixedPrice": 51.0,  
      "UsagePrice": 0.0,  
      "CurrencyCode": "USD",  
      "ProductDescription": "mysql",  
      "OfferingType": "Partial Upfront",  
      "MultiAZ": false,  
      "RecurringCharges": [  
        {  
          "RecurringChargeAmount": 0.006,  
          "RecurringChargeFrequency": "Hourly"  
        }  
      ]  
    },  
    ... some output truncated ...  
  ]  
}
```

Para obtener más información, consulte [Instancias de bases de datos reservadas para Amazon RDS](#) en la Guía del RDS usuario de Amazon.

Ejemplo 2: Para comprar una instancia de base de datos reservada

El siguiente `purchase-reserved-db-instances-offering` ejemplo muestra cómo comprar la oferta de instancias de base de datos reservadas del ejemplo anterior.

```
aws rds purchase-reserved-db-instances -offering --id 8ba30be1-b9ec-447f-8f23-6114e3f4c7b4  
reserved-db-instances-offering
```

Salida:



```
{
  "ReservedDBInstance": {
    "ReservedDBInstanceId": "ri-2020-06-29-16-54-57-670",
    "ReservedDBInstancesOfferingId": "8ba30be1-b9ec-447f-8f23-6114e3f4c7b4",
    "DBInstanceClass": "db.t2.micro",
    "StartTime": "2020-06-29T16:54:57.670Z",
    "Duration": 31536000,
    "FixedPrice": 51.0,
    "UsagePrice": 0.0,
    "CurrencyCode": "USD",
    "DBInstanceCount": 1,
    "ProductDescription": "mysql",
    "OfferingType": "Partial Upfront",
    "MultiAZ": false,
    "State": "payment-pending",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": 0.006,
        "RecurringChargeFrequency": "Hourly"
      }
    ],
    "ReservedDBInstanceArn": "arn:aws:rds:us-west-2:123456789012:ri:ri-2020-06-29-16-54-57-670"
  }
}
```

Para obtener más información, consulte [Instancias de bases de datos reservadas para Amazon RDS](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [PurchaseReservedDbInstancesOfferings](#) la Referencia de AWS CLI comandos.

## reboot-db-instance

En el siguiente ejemplo de código, se muestra cómo usar `reboot-db-instance`.

### AWS CLI

Reinicio de una instancia de base de datos

En el siguiente ejemplo `reboot-db-instance`, se inicia un reinicio de la instancia de base de datos especificada.

```
aws rds reboot-db-instance \  
  --db-instance-identifier test-mysql-instance
```

Salida:

```
{  
  "DBInstance": {  
    "DBInstanceIdentifier": "test-mysql-instance",  
    "DBInstanceClass": "db.t3.micro",  
    "Engine": "mysql",  
    "DBInstanceStatus": "rebooting",  
    "MasterUsername": "admin",  
    "Endpoint": {  
      "Address": "test-mysql-instance.#####.us-  
west-2.rds.amazonaws.com",  
      "Port": 3306,  
      "HostedZoneId": "Z1PVIF0EXAMPLE"  
    },  
    ... output omitted...  
  }  
}
```

Para obtener más información, consulte [Reiniciar una instancia](#) de base de datos en la Guía RDS del usuario de Amazon.

- Para API obtener más información, consulte [RebootDBInstance](#) en la Referencia de AWS CLI comandos.

## reboot-db-shard-group

En el siguiente ejemplo de código, se muestra cómo usar `reboot-db-shard-group`.

### AWS CLI

Ejemplo 1: Para reiniciar un grupo de fragmentos de base de datos

En el siguiente `reboot-db-shard-group` ejemplo, se reinicia un grupo de fragmentos de base de datos.

```
aws rds reboot-db-shard-group \  
  --db-shard-group-identifier
```

```
--db-shard-group-identifier my-db-shard-group
```

Salida:

```
{
  "DBShardGroups": [
    {
      "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
      "DBShardGroupIdentifier": "my-db-shard-group",
      "DBClusterIdentifier": "my-sv2-cluster",
      "MaxACU": 1000.0,
      "ComputeRedundancy": 0,
      "Status": "available",
      "PubliclyAccessible": false,
      "Endpoint": "my-sv2-cluster.limitless-cekyceexample.us-east-2.rds.amazonaws.com"
    }
  ]
}
```

Para obtener más información, consulte [Reiniciar un clúster de base de datos de Amazon Aurora o una instancia de base de datos de Amazon Aurora](#) en la Guía del usuario de Amazon Aurora.

Ejemplo 2: Para describir sus grupos de fragmentos de base de datos

En el siguiente `describe-db-shard-groups` ejemplo, se recuperan los detalles de los grupos de fragmentos de base de datos después de ejecutar el comando. `reboot-db-shard-group` El grupo de fragmentos de base de datos `my-db-shard-group` se está reiniciando ahora.

```
aws rds describe-db-shard-groups
```

Salida:

```
{
  "DBShardGroups": [
    {
      "DBShardGroupResourceId": "shardgroup-7bb446329da94788b3f957746example",
      "DBShardGroupIdentifier": "limitless-test-shard-grp",
      "DBClusterIdentifier": "limitless-test-cluster",
      "MaxACU": 768.0,
      "ComputeRedundancy": 0,
      "Status": "available",
    }
  ]
}
```

```

        "PubliclyAccessible": true,
        "Endpoint": "limitless-test-cluster.limitless-cekyexample.us-
east-2.rds.amazonaws.com"
    },
    {
        "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
        "DBShardGroupIdentifier": "my-db-shard-group",
        "DBClusterIdentifier": "my-sv2-cluster",
        "MaxACU": 1000.0,
        "ComputeRedundancy": 0,
        "Status": "rebooting",
        "PubliclyAccessible": false,
        "Endpoint": "my-sv2-cluster.limitless-cekyexample.us-
east-2.rds.amazonaws.com"
    }
]
}

```

Para obtener más información, consulte [Reiniciar un clúster de base de datos de Amazon Aurora o una instancia de base de datos de Amazon Aurora](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte la Referencia [RebootDbShardGroup](#) de AWS CLI comandos.

## register-db-proxy-targets

En el siguiente ejemplo de código, se muestra cómo usar `register-db-proxy-targets`.

### AWS CLI

Para registrar un proxy de base de datos en una base de datos

El siguiente `register-db-proxy-targets` ejemplo crea la asociación entre una base de datos y un proxy.

```

aws rds register-db-proxy-targets \
  --db-proxy-name proxyExample \
  --db-cluster-identifiers database-5

```

Salida:

```
{
```

```
"DBProxyTargets": [  
  {  
    "RdsResourceId": "database-5",  
    "Port": 3306,  
    "Type": "TRACKED_CLUSTER",  
    "TargetHealth": {  
      "State": "REGISTERING"  
    }  
  },  
  {  
    "Endpoint": "database-5instance-1.ab0cd1efghij.us-  
east-1.rds.amazonaws.com",  
    "RdsResourceId": "database-5",  
    "Port": 3306,  
    "Type": "RDS_INSTANCE",  
    "TargetHealth": {  
      "State": "REGISTERING"  
    }  
  }  
]  
}
```

Para obtener más información, consulte [Creación de un RDS proxy](#) en la Guía del RDS usuario de Amazon y [Creación de un RDS proxy](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [RegisterDbProxyTargets](#) la Referencia de AWS CLI comandos.

## **remove-from-global-cluster**

En el siguiente ejemplo de código, se muestra cómo usar `remove-from-global-cluster`.

### AWS CLI

Para separar un clúster secundario de Aurora de un clúster de base de datos global de Aurora

En el siguiente `remove-from-global-cluster` ejemplo, se separa un clúster secundario de Aurora de un clúster de base de datos global de Aurora. El clúster pasa de ser de solo lectura a ser un clúster independiente con capacidad de lectura-escritura.

```
aws rds remove-from-global-cluster \  
  --region us-west-2 \  
  \
```

```
--global-cluster-identifier myglobalcluster \  
--db-cluster-identifier arn:aws:rds:us-west-2:123456789012:cluster:DB-1
```

Salida:

```
{  
  "GlobalCluster": {  
    "GlobalClusterIdentifier": "myglobalcluster",  
    "GlobalClusterResourceId": "cluster-abc123def456gh",  
    "GlobalClusterArn": "arn:aws:rds::123456789012:global-  
cluster:myglobalcluster",  
    "Status": "available",  
    "Engine": "aurora-postgresql",  
    "EngineVersion": "10.11",  
    "StorageEncrypted": true,  
    "DeletionProtection": false,  
    "GlobalClusterMembers": [  
      {  
        "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:js-  
global-cluster",  
        "Readers": [  
          "arn:aws:rds:us-west-2:123456789012:cluster:DB-1"  
        ],  
        "IsWriter": true  
      },  
      {  
        "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:DB-1",  
        "Readers": [],  
        "IsWriter": false,  
        "GlobalWriteForwardingStatus": "disabled"  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [Eliminar un clúster de una base de datos global de Amazon Aurora](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [RemoveFromGlobalCluster](#) la Referencia de AWS CLI comandos.

## remove-option-from-option-group

En el siguiente ejemplo de código, se muestra cómo usar `remove-option-from-option-group`.

### AWS CLI

Para eliminar una opción de un grupo de opciones

En el siguiente `remove-option-from-option-group` ejemplo, se elimina la OEM opción `demyoptiongroup`.

```
aws rds remove-option-from-option-group \  
  --option-group-name myoptiongroup \  
  --options OEM \  
  --apply-immediately
```

Salida:

```
{  
  "OptionGroup": {  
    "OptionGroupName": "myoptiongroup",  
    "OptionGroupDescription": "Test",  
    "EngineName": "oracle-ee",  
    "MajorEngineVersion": "19",  
    "Options": [],  
    "AllowsVpcAndNonVpcInstanceMemberships": true,  
    "OptionGroupArn": "arn:aws:rds:us-east-1:123456789012:og:myoptiongroup"  
  }  
}
```

Para obtener más información, consulte [Eliminar una opción de un grupo de opciones](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [RemoveOptionFromOptionGroup](#) la Referencia de AWS CLI comandos.

## remove-role-from-db-cluster

En el siguiente ejemplo de código, se muestra cómo usar `remove-role-from-db-cluster`.

## AWS CLI

Para desasociar una función de AWS Identity and Access Management (IAM) de un clúster de base de datos

En el siguiente `remove-role-from-db-cluster` ejemplo, se elimina un rol de un clúster de base de datos.

```
aws rds remove-role-from-db-cluster \  
  --db-cluster-identifier mydbcluster \  
  --role-arn arn:aws:iam::123456789012:role/RDSLoadFromS3
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Asociación de un IAM rol a un clúster de Amazon Aurora My SQL DB](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [RemoveRoleFromDbCluster](#) la Referencia de AWS CLI comandos.

## `remove-role-from-db-instance`

En el siguiente ejemplo de código, se muestra cómo usar `remove-role-from-db-instance`.

## AWS CLI

Para desasociar un rol de AWS Identity and Access Management (IAM) de una instancia de base de datos

En el siguiente `remove-role-from-db-instance` ejemplo, se elimina el rol nombrado `rds-s3-integration-role` de una instancia de base de datos de Oracle denominada `test-instance`.

```
aws rds remove-role-from-db-instance \  
  --db-instance-identifier test-instance \  
  --feature-name S3_INTEGRATION \  
  --role-arn arn:aws:iam::111122223333:role/rds-s3-integration-role
```

Este comando no genera ninguna salida.

Para obtener más información, consulta Cómo [deshabilitar la integración RDS SQL del servidor con S3](#) en la Guía del RDS usuario de Amazon.



- Para API obtener más información, consulte [RemoveRoleFromDbInstance](#) la Referencia de AWS CLI comandos.

## remove-source-identifier-from-subscription

En el siguiente ejemplo de código, se muestra cómo usar `remove-source-identifier-from-subscription`.

### AWS CLI

Para eliminar un identificador de origen de una suscripción

En el siguiente `remove-source-identifier` ejemplo, se quita el identificador de origen especificado de una suscripción existente.

```
aws rds remove-source-identifier-from-subscription \  
  --subscription-name my-instance-events \  
  --source-identifier test-instance-repl
```

Salida:

```
{  
  "EventSubscription": {  
    "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-  
events",  
    "SubscriptionCreationTime": "Tue Jul 31 23:22:01 UTC 2018",  
    "EventCategoriesList": [  
      "backup",  
      "recovery"  
    ],  
    "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",  
    "Status": "modifying",  
    "CustSubscriptionId": "my-instance-events",  
    "CustomerAwsId": "123456789012",  
    "SourceIdsList": [  
      "test-instance"  
    ],  
    "SourceType": "db-instance",  
    "Enabled": false  
  }  
}
```

- Para API obtener más información, consulte [RemoveSourceIdentifierFromSubscription](#) la Referencia de AWS CLI comandos.

## **remove-tags-from-resource**

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags-from-resource`.

### AWS CLI

Cómo eliminar etiquetas de un recurso

En el siguiente `remove-tags-from-resource` ejemplo, se eliminan las etiquetas de un recurso.

```
aws rds remove-tags-from-resource \  
  --resource-name arn:aws:rds:us-east-1:123456789012:db:mydbinstance \  
  --tag-keys Name Environment
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de RDS los recursos de Amazon](#) en la Guía del RDS usuario de Amazon y [Etiquetado de RDS los recursos de Amazon](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte la Referencia [RemoveTagsFromResource](#) de AWS CLI comandos.

## **reset-db-cluster-parameter-group**

En el siguiente ejemplo de código, se muestra cómo usar `reset-db-cluster-parameter-group`.

### AWS CLI

Ejemplo 1: Para restablecer todos los parámetros a sus valores por defecto

El siguiente `reset-db-cluster-parameter-group` ejemplo restablece todos los valores de los parámetros de un grupo de parámetros de un clúster de base de datos creado por el cliente a sus valores predeterminados.

```
aws rds reset-db-cluster-parameter-group \  
  --db-cluster-parameter-group-name mydbc1pg \  
  --parameter-name myparam
```

**--reset-all-parameters**

Salida:

```
{
  "DBClusterParameterGroupName": "mydbclpg"
}
```

Para obtener más información, consulte [Trabajar con grupos de parámetros de bases de datos y grupos de parámetros de clústeres de bases de datos](#) en la Guía del usuario de Amazon Aurora.

Ejemplo 2: Para restablecer parámetros específicos a sus valores predeterminados

El siguiente `reset-db-cluster-parameter-group` ejemplo restablece los valores de los parámetros específicos a sus valores predeterminados en un grupo de parámetros de un clúster de base de datos creado por el cliente.

```
aws rds reset-db-cluster-parameter-group \
  --db-cluster-parameter-group-name mydbclpgy \
  --parameters "ParameterName=max_connections,ApplyMethod=immediate" \
  "ParameterName=max_allowed_packet,ApplyMethod=immediate"
```

Salida:

```
{
  "DBClusterParameterGroupName": "mydbclpg"
}
```

Para obtener más información, consulte [Trabajar con grupos de parámetros de bases de datos y grupos de parámetros de clústeres de bases de datos](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ResetDbClusterParameterGroup](#) la Referencia de AWS CLI comandos.

**reset-db-parameter-group**

En el siguiente ejemplo de código, se muestra cómo usar `reset-db-parameter-group`.

AWS CLI

Ejemplo 1: Para restablecer todos los parámetros a sus valores por defecto

El siguiente `reset-db-parameter-group` ejemplo restablece todos los valores de los parámetros de un grupo de parámetros de base de datos creado por el cliente a sus valores predeterminados.

```
aws rds reset-db-parameter-group \  
  --db-parameter-group-name mypg \  
  --reset-all-parameters
```

Salida:

```
{  
  "DBParameterGroupName": "mypg"  
}
```

Para obtener más información, consulte [Trabajar con grupos de parámetros de base de datos](#) en la Guía del RDS usuario de Amazon y [Trabajar con grupos de parámetros de base de datos y grupos de parámetros de clústeres de base de datos](#) en la Guía del usuario de Amazon Aurora.

Ejemplo 2: Para restablecer parámetros específicos a sus valores predeterminados

El siguiente `reset-db-parameter-group` ejemplo restablece los valores de los parámetros específicos a sus valores predeterminados en un grupo de parámetros de base de datos creado por el cliente.

```
aws rds reset-db-parameter-group \  
  --db-parameter-group-name mypg \  
  --parameters "ParameterName=max_connections,ApplyMethod=immediate" \  
               "ParameterName=max_allowed_packet,ApplyMethod=immediate"
```

Salida:

```
{  
  "DBParameterGroupName": "mypg"  
}
```

Para obtener más información, consulte [Trabajar con grupos de parámetros de base de datos](#) en la Guía del RDS usuario de Amazon y [Trabajar con grupos de parámetros de base de datos y grupos de parámetros de clústeres de base de datos](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [ResetDbParameterGroup](#) la Referencia de AWS CLI comandos.

## restore-db-cluster-from-s3

En el siguiente ejemplo de código, se muestra cómo usar `restore-db-cluster-from-s3`.

### AWS CLI

Para restaurar un clúster de base de datos Amazon Aurora desde Amazon S3

El siguiente `restore-db-cluster-from-s3` ejemplo restaura un clúster de base de datos compatible con Amazon Aurora My SQL versión 5.7 a partir de un archivo de respaldo de base de datos My SQL 5.7 en Amazon S3.

```
aws rds restore-db-cluster-from-s3 \
  --db-cluster-identifier cluster-s3-restore \
  --engine aurora-mysql \
  --master-username admin \
  --master-user-password mypassword \
  --s3-bucket-name mybucket \
  --s3-prefix test-backup \
  --s3-ingestion-role-arn arn:aws:iam::123456789012:role/service-role/TestBackup \
  --source-engine mysql \
  --source-engine-version 5.7.28
```

Salida:

```
{
  "DBCluster": {
    "AllocatedStorage": 1,
    "AvailabilityZones": [
      "us-west-2c",
      "us-west-2a",
      "us-west-2b"
    ],
    "BackupRetentionPeriod": 1,
    "DBClusterIdentifier": "cluster-s3-restore",
    "DBClusterParameterGroup": "default.aurora-mysql5.7",
    "DBSubnetGroup": "default",
    "Status": "creating",
    "Endpoint": "cluster-s3-restore.cluster-co3xyzabc123.us-
west-2.rds.amazonaws.com",
    "ReaderEndpoint": "cluster-s3-restore.cluster-ro-co3xyzabc123.us-
west-2.rds.amazonaws.com",
```

```

    "MultiAZ": false,
    "Engine": "aurora-mysql",
    "EngineVersion": "5.7.12",
    "Port": 3306,
    "MasterUsername": "admin",
    "PreferredBackupWindow": "11:15-11:45",
    "PreferredMaintenanceWindow": "thu:12:19-thu:12:49",
    "ReadReplicaIdentifiers": [],
    "DBClusterMembers": [],
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-#####",
        "Status": "active"
      }
    ],
    "HostedZoneId": "Z1PVIIF0EXAMPLE",
    "StorageEncrypted": false,
    "DbClusterResourceId": "cluster-SU5THYQQH0WCXZZDGXREXAMPLE",
    "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:cluster-s3-
restore",
    "AssociatedRoles": [],
    "IAMDatabaseAuthenticationEnabled": false,
    "ClusterCreateTime": "2020-07-27T14:22:08.095Z",
    "EngineMode": "provisioned",
    "DeletionProtection": false,
    "HttpEndpointEnabled": false,
    "CopyTagsToSnapshot": false,
    "CrossAccountClone": false,
    "DomainMemberships": []
  }
}

```

Para obtener más información, consulte [Migración de datos desde My SQL mediante un bucket de Amazon S3](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [RestoreDbClusterFromS3](#) en la referencia de AWS CLI comandos.

## **restore-db-cluster-from-snapshot**

En el siguiente ejemplo de código, se muestra cómo usar `restore-db-cluster-from-snapshot`.

## AWS CLI

Para restaurar un clúster de base de datos a partir de una instantánea

`restore-db-cluster-from-snapshot`A continuación, se restaura un SQL clúster de base de datos Aurora Postgre compatible con la SQL versión 10.7 de Postgre a partir de una instantánea de clúster de base de datos denominada `test-instance-snapshot`

```
aws rds restore-db-cluster-from-snapshot \  
  --db-cluster-identifier newdbcluster \  
  --snapshot-identifier test-instance-snapshot \  
  --engine aurora-postgresql \  
  --engine-version 10.7
```

Salida:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 1,  
    "AvailabilityZones": [  
      "us-west-2c",  
      "us-west-2a",  
      "us-west-2b"  
    ],  
    "BackupRetentionPeriod": 7,  
    "DatabaseName": "",  
    "DBClusterIdentifier": "newdbcluster",  
    "DBClusterParameterGroup": "default.aurora-postgresql10",  
    "DBSubnetGroup": "default",  
    "Status": "creating",  
    "Endpoint": "newdbcluster.cluster-#####.us-west-2.rds.amazonaws.com",  
    "ReaderEndpoint": "newdbcluster.cluster-ro-#####.us-  
west-2.rds.amazonaws.com",  
    "MultiAZ": false,  
    "Engine": "aurora-postgresql",  
    "EngineVersion": "10.7",  
    "Port": 5432,  
    "MasterUsername": "postgres",  
    "PreferredBackupWindow": "09:33-10:03",  
    "PreferredMaintenanceWindow": "sun:12:22-sun:12:52",  
    "ReadReplicaIdentifiers": [],  
    "DBClusterMembers": [],  
    "VpcSecurityGroups": [  

```

```
    {
      "VpcSecurityGroupId": "sg-#####",
      "Status": "active"
    }
  ],
  "HostedZoneId": "Z1PVIIF0EXAMPLE",
  "StorageEncrypted": true,
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/287364e4-33e3-4755-a3b0-
a1b2c3d4e5f6",
  "DbClusterResourceId": "cluster-5DSB5IFQDDUVAWOUWM1EXAMPLE",
  "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:newdbcluster",
  "AssociatedRoles": [],
  "IAMDatabaseAuthenticationEnabled": false,
  "ClusterCreateTime": "2020-06-05T15:06:58.634Z",
  "EngineMode": "provisioned",
  "DeletionProtection": false,
  "HttpEndpointEnabled": false,
  "CopyTagsToSnapshot": false,
  "CrossAccountClone": false,
  "DomainMemberships": []
}
}
```

Para obtener más información, consulte [Restauración desde una instantánea de un clúster](#) de base de datos en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [RestoreDbClusterFromSnapshot](#) la Referencia de AWS CLI comandos.

## **restore-db-cluster-to-point-in-time**

En el siguiente ejemplo de código, se muestra cómo usar `restore-db-cluster-to-point-in-time`.

### AWS CLI

Para restaurar un clúster de base de datos a una hora específica

El siguiente `restore-db-cluster-to-point-in-time` ejemplo restaura el clúster de base de datos denominado `database-4` a la última hora posible. Si `copy-on-write` se utiliza como tipo de restauración, se restaura el nuevo clúster de base de datos como un clon del clúster de base de datos de origen.



```
aws rds restore-db-cluster-to-point-in-time \
  --source-db-cluster-identifier database-4 \
  --db-cluster-identifier sample-cluster-clone \
  --restore-type copy-on-write \
  --use-latest-restorable-time
```

Salida:

```
{
  "DBCluster": {
    "AllocatedStorage": 1,
    "AvailabilityZones": [
      "us-west-2c",
      "us-west-2a",
      "us-west-2b"
    ],
    "BackupRetentionPeriod": 7,
    "DatabaseName": "",
    "DBClusterIdentifier": "sample-cluster-clone",
    "DBClusterParameterGroup": "default.aurora-postgresql10",
    "DBSubnetGroup": "default",
    "Status": "creating",
    "Endpoint": "sample-cluster-clone.cluster-#####.us-
west-2.rds.amazonaws.com",
    "ReaderEndpoint": "sample-cluster-clone.cluster-ro-#####.us-
west-2.rds.amazonaws.com",
    "MultiAZ": false,
    "Engine": "aurora-postgresql",
    "EngineVersion": "10.7",
    "Port": 5432,
    "MasterUsername": "postgres",
    "PreferredBackupWindow": "09:33-10:03",
    "PreferredMaintenanceWindow": "sun:12:22-sun:12:52",
    "ReadReplicaIdentifiers": [],
    "DBClusterMembers": [],
    "VpcSecurityGroups": [
      {
        "VpcSecurityGroupId": "sg-#####",
        "Status": "active"
      }
    ],
    "HostedZoneId": "Z1PVIF0EXAMPLE",
    "StorageEncrypted": true,
```

```

    "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/287364e4-33e3-4755-a3b0-
a1b2c3d4e5f6",
    "DbClusterResourceId": "cluster-BIZ77GDSA2XBSTNPFW1EXAMPLE",
    "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster-
clone",
    "AssociatedRoles": [],
    "IAMDatabaseAuthenticationEnabled": false,
    "CloneGroupId": "8d19331a-099a-45a4-b4aa-11aa22bb33cc44dd",
    "ClusterCreateTime": "2020-03-10T19:57:38.967Z",
    "EngineMode": "provisioned",
    "DeletionProtection": false,
    "HttpEndpointEnabled": false,
    "CopyTagsToSnapshot": false,
    "CrossAccountClone": false
  }
}

```

Para obtener más información, consulte [Restauración de un clúster de base de datos a una hora específica](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [RestoreDbClusterToPointInTime](#) la Referencia de AWS CLI comandos.

## restore-db-instance-from-db-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `restore-db-instance-from-db-snapshot`.

### AWS CLI

Para restaurar una instancia de base de datos a partir de una instantánea de base de datos

En el siguiente `restore-db-instance-from-db-snapshot` ejemplo, se crea una nueva instancia de base de datos denominada `db7-new-instance` con la clase de `db.t3.small` instancia de base de datos de la instantánea de base de datos especificada. La instancia de base de datos de origen de la que se tomó la instantánea utiliza una clase de instancia de base de datos obsoleta, por lo que no se puede actualizar.

```

aws rds restore-db-instance-from-db-snapshot \
  --db-instance-identifier db7-new-instance \
  --db-snapshot-identifier db7-test-snapshot \
  --db-instance-class db.t3.small

```

Salida:

```
{
  "DBInstance": {
    "DBInstanceIdentifier": "db7-new-instance",
    "DBInstanceClass": "db.t3.small",
    "Engine": "mysql",
    "DBInstanceStatus": "creating",

    ...output omitted...

    "PreferredMaintenanceWindow": "mon:07:37-mon:08:07",
    "PendingModifiedValues": {},
    "MultiAZ": false,
    "EngineVersion": "5.7.22",
    "AutoMinorVersionUpgrade": true,
    "ReadReplicaDBInstanceIdentifiers": [],
    "LicenseModel": "general-public-license",

    ...output omitted...

    "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:db7-new-instance",
    "IAMDatabaseAuthenticationEnabled": false,
    "PerformanceInsightsEnabled": false,
    "DeletionProtection": false,
    "AssociatedRoles": []
  }
}
```

Para obtener más información, consulte [Restauración a partir de una instantánea de base de datos](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [RestoreDbInstanceFromDbSnapshot](#) la Referencia de AWS CLI comandos.

### **restore-db-instance-from-s3**

En el siguiente ejemplo de código, se muestra cómo usar `restore-db-instance-from-s3`.

#### AWS CLI

Para restaurar una instancia de base de datos a partir de una copia de seguridad en Amazon S3

En el siguiente `restore-db-instance-from-s3` ejemplo, se crea una nueva instancia de base de datos con el nombre `restored-test-instance` de una copia de seguridad existente en el bucket de `my-backups` S3.

```
aws rds restore-db-instance-from-s3 \
  --db-instance-identifier restored-test-instance \
  --allocated-storage 250 --db-instance-class db.m4.large --engine mysql \
  --master-username master --master-user-password secret99 \
  --s3-bucket-name my-backups --s3-ingestion-role-
arn arn:aws:iam::123456789012:role/my-role \
  --source-engine mysql --source-engine-version 5.6.27
```

- Para API obtener más información, consulte [RestoreDbInstanceFromS3](#) en la Referencia de AWS CLI comandos.

## **restore-db-instance-to-point-in-time**

En el siguiente ejemplo de código, se muestra cómo usar `restore-db-instance-to-point-in-time`.

### AWS CLI

Ejemplo 1: Para restaurar una instancia de base de datos a un punto en el tiempo

El siguiente `restore-db-instance-to-point-in-time` ejemplo `test-instance` restaura una nueva instancia de base de datos denominada `restored-test-instance`, a partir del momento especificado.

```
aws rds restore-db-instance-to-point-in-time \
  --source-db-instance-identifier test-instance \
  --target-db-instance restored-test-instance \
  --restore-time 2018-07-30T23:45:00.000Z
```

Salida:

```
{
  "DBInstance": {
    "AllocatedStorage": 20,
    "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:restored-test-
instance",
    "DBInstanceStatus": "creating",
```

```

    "DBInstanceIdentifier": "restored-test-instance",
    ...some output omitted...
  }
}

```

Para obtener más información, consulte [Restauración de una instancia de base de datos a una hora específica](#) en la Guía del RDS usuario de Amazon.

Ejemplo 2: Para restaurar una instancia de base de datos a una hora específica a partir de una copia de seguridad replicada

El siguiente `restore-db-instance-to-point-in-time` ejemplo restaura una instancia de base de datos de Oracle a la hora especificada a partir de una copia de seguridad automatizada replicada.

```

aws rds restore-db-instance-to-point-in-time \
  --source-db-instance-automated-backups-arn "arn:aws:rds:us-
west-2:123456789012:auto-backup:ab-jkib2gfg5rv7replzadausbrktni2bn4example" \
  --target-db-instance-identifier myorclinstance-from-replicated-backup \
  --restore-time 2020-12-08T18:45:00.000Z

```

Salida:

```

{
  "DBInstance": {
    "DBInstanceIdentifier": "myorclinstance-from-replicated-backup",
    "DBInstanceClass": "db.t3.micro",
    "Engine": "oracle-se2",
    "DBInstanceStatus": "creating",
    "MasterUsername": "admin",
    "DBName": "ORCL",
    "AllocatedStorage": 20,
    "PreferredBackupWindow": "07:45-08:15",
    "BackupRetentionPeriod": 14,
    ... some output omitted ...
    "DbiResourceId": "db-KGLXG75BGVIWKQT7NQ4EXAMPLE",
    "CACertificateIdentifier": "rds-ca-2019",
    "DomainMemberships": [],
    "CopyTagsToSnapshot": false,
    "MonitoringInterval": 0,
    "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:myorclinstance-from-
replicated-backup",

```

```

    "IAMDatabaseAuthenticationEnabled": false,
    "PerformanceInsightsEnabled": false,
    "DeletionProtection": false,
    "AssociatedRoles": [],
    "TagList": []
  }
}

```

Para obtener más información, consulte [Restauración a un tiempo específico a partir de una copia de seguridad replicada](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [RestoreDbInstanceToPointInTime](#) la Referencia de AWS CLI comandos.

## start-activity-stream

En el siguiente ejemplo de código, se muestra cómo usar `start-activity-stream`.

### AWS CLI

Para iniciar un flujo de actividad de una base de datos

El siguiente `start-activity-stream` ejemplo inicia un flujo de actividad asíncrona para supervisar un clúster de Aurora denominado. `my-pg-cluster`

```

aws rds start-activity-stream \
  --region us-east-1 \
  --mode async \
  --kms-key-id arn:aws:kms:us-east-1:1234567890123:key/a12c345d-6ef7-890g-h123-456i789jk011 \
  --resource-arn arn:aws:rds:us-east-1:1234567890123:cluster:my-pg-cluster \
  --apply-immediately

```

Salida:

```

{
  "KmsKeyId": "arn:aws:kms:us-east-1:1234567890123:key/a12c345d-6ef7-890g-h123-456i789jk011",
  "KinesisStreamName": "aws-rds-das-cluster-0ABCDEFGH11JKLM2NOPQ3R4S",
  "Status": "starting",
  "Mode": "async",
  "ApplyImmediately": true
}

```

```
}
```

Para obtener más información, consulte [Iniciar un flujo de actividad de base](#) de datos en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [StartActivityStream](#) la Referencia de AWS CLI comandos.

## start-db-cluster

En el siguiente ejemplo de código, se muestra cómo usar `start-db-cluster`.

### AWS CLI

Para iniciar un clúster de base de datos

El siguiente `start-db-cluster` ejemplo inicia un clúster de base de datos y sus instancias de base de datos.

```
aws rds start-db-cluster \  
  --db-cluster-identifier mydbcluster
```

Salida:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 1,  
    "AvailabilityZones": [  
      "us-east-1a",  
      "us-east-1e",  
      "us-east-1b"  
    ],  
    "BackupRetentionPeriod": 1,  
    "DatabaseName": "mydb",  
    "DBClusterIdentifier": "mydbcluster",  
    "...some output truncated..."  
  }  
}
```

Para obtener más información, consulte [Detener e iniciar un clúster de base de datos de Amazon Aurora](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [StartDbCluster](#) la Referencia de AWS CLI comandos.

## start-db-instance-automated-backups-replication

En el siguiente ejemplo de código, se muestra cómo usar `start-db-instance-automated-backups-replication`.

### AWS CLI

Para habilitar las copias de seguridad automatizadas entre regiones

En el siguiente `start-db-instance-automated-backups-replication` ejemplo, se replican las copias de seguridad automatizadas de una instancia de base de datos de la región EE.UU. Este (Norte de Virginia) a EE.UU. Oeste (Oregón). El período de retención de las copias de seguridad es de 14 días.

```
aws rds start-db-instance-automated-backups-replication \  
  --region us-west-2 \  
  --source-db-instance-arn "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db" \  
  --backup-retention-period 14
```

Salida:

```
{  
  "DBInstanceAutomatedBackup": {  
    "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db",  
    "DbiResourceId": "db-JKIB2GFQ5RV7REPLZA4EXAMPLE",  
    "Region": "us-east-1",  
    "DBInstanceIdentifier": "new-orcl-db",  
    "RestoreWindow": {},  
    "AllocatedStorage": 20,  
    "Status": "pending",  
    "Port": 1521,  
    "InstanceCreateTime": "2020-12-04T15:28:31Z",  
    "MasterUsername": "admin",  
    "Engine": "oracle-se2",  
    "EngineVersion": "12.1.0.2.v21",  
    "LicenseModel": "bring-your-own-license",  
    "OptionGroupName": "default:oracle-se2-12-1",  
    "Encrypted": false,  
  }  
}
```



```

    "StorageType": "gp2",
    "IAMDatabaseAuthenticationEnabled": false,
    "BackupRetentionPeriod": 14,
    "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-west-2:123456789012:auto-
backup:ab-jkib2gfg5rv7replzadausbrktni2bn4example"
  }
}

```

Para obtener más información, consulte [Habilitar copias de seguridad automatizadas entre regiones](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [StartDbInstanceAutomatedBackupsReplication](#) la Referencia de AWS CLI comandos.

## start-db-instance

En el siguiente ejemplo de código, se muestra cómo usar `start-db-instance`.

### AWS CLI

Para iniciar una instancia de base de datos

En el siguiente `start-db-instance` ejemplo, se inicia la instancia de base de datos especificada.

```

aws rds start-db-instance \
  --db-instance-identifier test-instance

```

Salida:

```

{
  "DBInstance": {
    "DBInstanceStatus": "starting",
    ...some output truncated...
  }
}

```

- Para API obtener más información, consulte [StartDbInstance](#) la Referencia de AWS CLI comandos.

## start-export-task

En el siguiente ejemplo de código, se muestra cómo usar `start-export-task`.

### AWS CLI

Para exportar una instantánea a Amazon S3

En el siguiente `start-export-task` ejemplo, se exporta una instantánea de base de datos denominada `db5-snapshot-test` al bucket de Amazon S3 denominado `mybucket`.

```
aws rds start-export-task \  
  --export-task-identifier my-s3-export \  
  --source-arn arn:aws:rds:us-west-2:123456789012:snapshot:db5-snapshot-test \  
  --s3-bucket-name mybucket \  
  --iam-role-arn arn:aws:iam::123456789012:role/service-role/ExportRole \  
  --kms-key-id arn:aws:kms:us-west-2:123456789012:key/abcd0000-7fca-4128-82f2-  
aabbccddeeff
```

Salida:

```
{  
  "ExportTaskIdentifier": "my-s3-export",  
  "SourceArn": "arn:aws:rds:us-west-2:123456789012:snapshot:db5-snapshot-test",  
  "SnapshotTime": "2020-03-27T20:48:42.023Z",  
  "S3Bucket": "mybucket",  
  "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/ExportRole",  
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/abcd0000-7fca-4128-82f2-  
aabbccddeeff",  
  "Status": "STARTING",  
  "PercentProgress": 0,  
  "TotalExtractedDataInGB": 0  
}
```

Para obtener más información, consulte [Exportación de una instantánea a un bucket de Amazon S3](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [StartExportTask](#) la Referencia de AWS CLI comandos.

## stop-activity-stream

En el siguiente ejemplo de código, se muestra cómo usar `stop-activity-stream`.

### AWS CLI

Para detener un flujo de actividad de una base de datos

El siguiente `stop-activity-stream` ejemplo detiene un flujo de actividad en un clúster de Aurora denominado `my-pg-cluster`.

```
aws rds stop-activity-stream \  
  --region us-east-1 \  
  --resource-arn arn:aws:rds:us-east-1:1234567890123:cluster:my-pg-cluster \  
  --apply-immediately
```

Salida:

```
{  
  "KmsKeyId": "arn:aws:kms:us-east-1:1234567890123:key/a12c345d-6ef7-890g-  
h123-456i789jk011",  
  "KinesisStreamName": "aws-rds-das-cluster-0ABCDEFGH11JKLM2NOPQ3R4S",  
  "Status": "stopping"  
}
```

Para obtener más información, consulte [Detener una transmisión de actividad](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [StopActivityStream](#) la Referencia de AWS CLI comandos.

## stop-db-cluster

En el siguiente ejemplo de código, se muestra cómo usar `stop-db-cluster`.

### AWS CLI

Para detener un clúster de base de datos

El siguiente `stop-db-cluster` ejemplo detiene un clúster de base de datos y sus instancias de base de datos.

```
aws rds stop-db-cluster \  
--db-cluster-identifier mydbcluster
```

Salida:

```
{  
  "DBCluster": {  
    "AllocatedStorage": 1,  
    "AvailabilityZones": [  
      "us-east-1a",  
      "us-east-1e",  
      "us-east-1b"  
    ],  
    "BackupRetentionPeriod": 1,  
    "DatabaseName": "mydb",  
    "DBClusterIdentifier": "mydbcluster",  
    "...some output truncated..."  
  }  
}
```

Para obtener más información, consulte [Detener e iniciar un clúster de base de datos de Amazon Aurora](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte [StopDbCluster](#) la Referencia de AWS CLI comandos.

## stop-db-instance-automated-backups-replication

En el siguiente ejemplo de código, se muestra cómo usar `stop-db-instance-automated-backups-replication`.

AWS CLI

Para dejar de replicar las copias de seguridad automatizadas

A continuación, se `stop-db-instance-automated-backups-replication` pone fin a la replicación de las copias de seguridad automatizadas en la región EE.UU. Oeste (Oregón). Las copias de seguridad replicadas se conservan de acuerdo con el período de retención de copias de seguridad establecido.

```
aws rds stop-db-instance-automated-backups-replication \  

```

```
--region us-west-2 \  
--source-db-instance-arn "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db"
```

Salida:

```
{  
  "DBInstanceAutomatedBackup": {  
    "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db",  
    "DbiResourceId": "db-JKIB2GFQ5RV7REPLZA4EXAMPLE",  
    "Region": "us-east-1",  
    "DBInstanceIdentifier": "new-orcl-db",  
    "RestoreWindow": {  
      "EarliestTime": "2020-12-04T23:13:21.030Z",  
      "LatestTime": "2020-12-07T19:59:57Z"  
    },  
    "AllocatedStorage": 20,  
    "Status": "replicating",  
    "Port": 1521,  
    "InstanceCreateTime": "2020-12-04T15:28:31Z",  
    "MasterUsername": "admin",  
    "Engine": "oracle-se2",  
    "EngineVersion": "12.1.0.2.v21",  
    "LicenseModel": "bring-your-own-license",  
    "OptionGroupName": "default:oracle-se2-12-1",  
    "Encrypted": false,  
    "StorageType": "gp2",  
    "IAMDatabaseAuthenticationEnabled": false,  
    "BackupRetentionPeriod": 7,  
    "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-west-2:123456789012:auto-backup:ab-jkib2gfgq5rv7replzadtausbrktni2bn4example"  
  }  
}
```

Para obtener más información, consulte [Detener la replicación automática de copias](#) de seguridad en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [StopDbInstanceAutomatedBackupsReplication](#) la Referencia de AWS CLI comandos.

## stop-db-instance

En el siguiente ejemplo de código, se muestra cómo usar stop-db-instance.

## AWS CLI

Para detener una instancia de base de datos

El siguiente `stop-db-instance` ejemplo detiene la instancia de base de datos especificada.

```
aws rds stop-db-instance \  
  --db-instance-identifier test-instance
```

Salida:

```
{  
  "DBInstance": {  
    "DBInstanceStatus": "stopping",  
    ...some output truncated...  
  }  
}
```

- Para API obtener más información, consulte [StopDbInstance](#) la Referencia de AWS CLI comandos.

## switchover-blue-green-deployment

En el siguiente ejemplo de código, se muestra cómo usar `switchover-blue-green-deployment`.

## AWS CLI

Ejemplo 1: Para cambiar una implementación azul a verde para una RDS instancia de base de datos

El siguiente `switchover-blue-green-deployment` ejemplo promueve el entorno ecológico especificado como nuevo entorno de producción.

```
aws rds switchover-blue-green-deployment \  
  --blue-green-deployment-identifier bgd-wi89nwzglccsfake \  
  --switchover-timeout 300
```

Salida:

```
{
```

```
"BlueGreenDeployment": {
  "BlueGreenDeploymentIdentifier": "bgd-v53303651eexfake",
  "BlueGreenDeploymentName": "bgd-cli-test-instance",
  "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
  "Target": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-green-
blhile",
  "SwitchoverDetails": [
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-green-blhile",
      "Status": "AVAILABLE"
    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1-green-k5fv7u",
      "Status": "AVAILABLE"
    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2-green-ggsh8m",
      "Status": "AVAILABLE"
    },
    {
      "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3",
      "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3-green-o2vwm0",
      "Status": "AVAILABLE"
    }
  ],
  "Tasks": [
    {
      "Name": "CREATING_READ_REPLICA_OF_SOURCE",
      "Status": "COMPLETED"
    },
    {
      "Name": "DB_ENGINE_VERSION_UPGRADE",
      "Status": "COMPLETED"
    }
  ]
}
```

```

    },
    {
      "Name": "CONFIGURE_BACKUPS",
      "Status": "COMPLETED"
    },
    {
      "Name": "CREATING_TOPOLOGY_OF_SOURCE",
      "Status": "COMPLETED"
    }
  ],
  "Status": "SWITCHOVER_IN_PROGRESS",
  "CreateTime": "2022-02-25T22:33:22.225000+00:00"
}
}

```

Para obtener más información, consulta [Cambiar una implementación azul/verde](#) en la Guía del RDSusuario de Amazon.

Ejemplo 2: Promover una implementación azul/verde para un clúster Aurora My DB SQL

El siguiente `switchover-blue-green-deployment` ejemplo promueve el entorno ecológico especificado como nuevo entorno de producción.

```

aws rds switchover-blue-green-deployment \
  --blue-green-deployment-identifier bgd-wi89nwzglccsfake \
  --switchover-timeout 300

```

Salida:

```

{
  "BlueGreenDeployment": {
    "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
    "BlueGreenDeploymentName": "my-blue-green-deployment",
    "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
    "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3ud8z6",
    "SwitchoverDetails": [
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster-green-3ud8z6",

```



```
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-1",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-1-green-bvxc73",
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-2",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-2-green-7wc4ie",
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-3",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-3-green-p4xxkz",
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint-green-np1ikl",
        "Status": "AVAILABLE"
      },
      {
        "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint",
        "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint-green-miszlf",
        "Status": "AVAILABLE"
      }
    ],
    "Tasks": [
      {
        "Name": "CREATING_READ_REPLICA_OF_SOURCE",
        "Status": "COMPLETED"
      }
    ]
  }
```

```
        "Name": "DB_ENGINE_VERSION_UPGRADE",
        "Status": "COMPLETED"
    },
    {
        "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
        "Status": "COMPLETED"
    },
    {
        "Name": "CREATE_CUSTOM_ENDPOINTS",
        "Status": "COMPLETED"
    }
],
"Status": "SWITCHOVER_IN_PROGRESS",
"CreateTime": "2022-02-25T22:38:49.522000+00:00"
}
}
```

Para obtener más información, consulte [Cambiar una implementación azul/verde](#) en la Guía del usuario de Amazon Aurora.

- Para API obtener más información, consulte la Referencia [SwitchoverBlueGreenDeployment](#) de AWS CLI comandos.

## Ejemplos RDS de Amazon Data Service que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface mediante Amazon RDS Data Service.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### batch-execute-statement

En el siguiente ejemplo de código, se muestra cómo usar `batch-execute-statement`.

#### AWS CLI

Para ejecutar una SQL sentencia por lotes

El siguiente `batch-execute-statement` ejemplo ejecuta una SQL sentencia por lotes sobre una matriz de datos con un conjunto de parámetros.

```
aws rds-data batch-execute-statement \
  --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \
  --database "mydb" \
  --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret" \
  --sql "insert into mytable values (:id, :val)" \
  --parameter-sets "[[{"name": "id", "value": {"longValue": 1}}, {"name": "val", "value": {"stringValue": "ValueOne"}}], [{"name": "id", "value": {"longValue": 2}}, {"name": "val", "value": {"stringValue": "ValueTwo"}}], [{"name": "id", "value": {"longValue": 3}}, {"name": "val", "value": {"stringValue": "ValueThree"}}]"]"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de los datos API para Aurora Serverless](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [BatchExecuteStatement](#) la Referencia de AWS CLI comandos.

### begin-transaction

En el siguiente ejemplo de código, se muestra cómo usar `begin-transaction`.

#### AWS CLI

Para iniciar una SQL transacción

El siguiente `begin-transaction` ejemplo inicia una SQL transacción.

```
aws rds-data begin-transaction \  
  --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \  
  --database "mydb" \  
  --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret"
```

Salida:

```
{  
  "transactionId": "ABC1234567890xyz"  
}
```

Para obtener más información, consulte [Uso de los datos API para Aurora Serverless](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [BeginTransaction](#) la Referencia de AWS CLI comandos.

## commit-transaction

En el siguiente ejemplo de código, se muestra cómo usar commit-transaction.

AWS CLI

Para confirmar una SQL transacción

El siguiente commit-transaction ejemplo finaliza la SQL transacción especificada y confirma los cambios que realizó como parte de ella.

```
aws rds-data commit-transaction \  
  --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \  
  --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret" \  
  --transaction-id "ABC1234567890xyz"
```

Salida:

```
{  
  "transactionStatus": "Transaction Committed"  
}
```

Para obtener más información, consulte [Uso de los datos API para Aurora Serverless](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [CommitTransaction](#) la Referencia de AWS CLI comandos.

## execute-statement

En el siguiente ejemplo de código, se muestra cómo usar `execute-statement`.

### AWS CLI

Ejemplo 1: Para ejecutar una SQL sentencia que forma parte de una transacción

El siguiente `execute-statement` ejemplo ejecuta una SQL sentencia que forma parte de una transacción.

```
aws rds-data execute-statement \
  --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \
  --database "mydb" \
  --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret" \
  --sql "update mytable set quantity=5 where id=201" \
  --transaction-id "ABC1234567890xyz"
```

Salida:

```
{
  "numberOfRecordsUpdated": 1
}
```

Ejemplo 2: Para ejecutar una SQL sentencia con parámetros

El siguiente `execute-statement` ejemplo ejecuta una SQL sentencia con parámetros.

```
aws rds-data execute-statement \
  --resource-arn "arn:aws:rds:us-east-1:123456789012:cluster:mydbcluster" \
  --database "mydb" \
  --secret-arn "arn:aws:secretsmanager:us-east-1:123456789012:secret:mysecret" \
  --sql "insert into mytable values (:id, :val)" \
  --parameters "[{\"name\": \"id\", \"value\": {\"longValue\": 1}}, {\"name\": \"val\", \"value\": {\"stringValue\": \"value1\"}}]"
```

Salida:

```
{
  "numberOfRecordsUpdated": 1
}
```

Para obtener más información, consulte [Uso de los datos API para Aurora Serverless](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [ExecuteStatement](#) la Referencia de AWS CLI comandos.

## rollback-transaction

En el siguiente ejemplo de código, se muestra cómo usar `rollback-transaction`.

### AWS CLI

Para anular una SQL transacción

El siguiente `rollback-transaction` ejemplo anula la SQL transacción especificada.

```
aws rds-data rollback-transaction \
  --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \
  --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret" \
  --transaction-id "ABC1234567890xyz"
```

Salida:

```
{
  "transactionStatus": "Rollback Complete"
}
```

Para obtener más información, consulte [Uso de los datos API para Aurora Serverless](#) en la Guía del RDS usuario de Amazon.

- Para API obtener más información, consulte [RollbackTransaction](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon RDS Performance Insights utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante Amazon RDS Performance Insights. AWS Command Line Interface

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **describe-dimension-keys**

En el siguiente ejemplo de código, se muestra cómo usar `describe-dimension-keys`.

AWS CLI

Para describir las claves de dimensión

En este ejemplo se solicitan los nombres de todos los eventos de espera. Los datos se resumen por nombre del evento y los valores agregados de esos eventos durante el período de tiempo especificado.

Comando:

```
aws pi describe-dimension-keys --service-type RDS --identifier db-LKCG0BK26374TPTDFX0IWVCP --start-time 1527026400 --end-time 1527080400 --metric db.load.avg --group-by '{"Group": "db.wait_event"}
```

Salida:

```
{
  "AlignedEndTime": 1.5270804E9,
  "AlignedStartTime": 1.5270264E9,
  "Keys": [
    {
      "Dimensions": {"db.wait_event.name": "wait/synch/mutex/innodb/aurora_lock_thread_slot_futex"},
      "Total": 0.05906906851195666
    },
    {
```

```

        "Dimensions": {"db.wait_event.name": "wait/io/aurora_redo_log_flush"},
        "Total": 0.015824722186149193
    },
    {
        "Dimensions": {"db.wait_event.name": "CPU"},
        "Total": 0.008014396230265477
    },
    {
        "Dimensions": {"db.wait_event.name": "wait/io/
aurora_respond_to_client"},
        "Total": 0.0036361612526204477
    },
    {
        "Dimensions": {"db.wait_event.name": "wait/io/table/sql/handler"},
        "Total": 0.0019108398419382965
    },
    {
        "Dimensions": {"db.wait_event.name": "wait/synch/cond/mysys/
my_thread_var::suspend"},
        "Total": 8.533847837782684E-4
    },
    {
        "Dimensions": {"db.wait_event.name": "wait/io/file/csv/data"},
        "Total": 6.864181956477376E-4
    },
    {
        "Dimensions": {"db.wait_event.name": "Unknown"},
        "Total": 3.895887056379051E-4
    },
    {
        "Dimensions": {"db.wait_event.name": "wait/synch/mutex/sql/
FILE_AS_TABLE::LOCK_shim_lists"},
        "Total": 3.710368625122906E-5
    },
    {
        "Dimensions": {"db.wait_event.name": "wait/lock/table/sql/handler"},
        "Total": 0
    }
]
}

```

- Para API obtener más información, consulte [DescribeDimensionKeys](#) la Referencia de AWS CLI comandos.



## get-resource-metrics

En el siguiente ejemplo de código, se muestra cómo usar `get-resource-metrics`.

### AWS CLI

Para obtener métricas de recursos

En este ejemplo, se solicitan puntos de datos para el grupo de dimensiones `db.wait_event` y para la dimensión `db.wait_event.name` dentro de ese grupo. En la respuesta, los puntos de datos relevantes se agrupan por la dimensión solicitada (`db.wait_event.name`).

Comando:

```
aws pi get-resource-metrics --service-type RDS --identifier db-LKCG0BK26374TPTDFX0IWVCP  
PM --start-time 1527026400 --end-time 1527080400 --period-  
in-seconds 300 --metric db.load.avg --metric-queries file://metric-queries.json
```

Los argumentos de se almacenan en un archivo `--metric-queries.` JSON `metric-queries.json` A continuación, se muestra el contenido de ese archivo:

```
[  
  {  
    "Metric": "db.load.avg",  
    "GroupBy": {  
      "Group": "db.wait_event"  
    }  
  }  
]
```

Salida:

```
{  
  "AlignedEndTime": 1.5270804E9,  
  "AlignedStartTime": 1.5270264E9,  
  "Identifier": "db-LKCG0BK26374TPTDFX0IWVCP",  
  "MetricList": [  
    {  
      "Key": {  
        "Metric": "db.load.avg"  
      },  
      "DataPoints": [  

```

```

        {
            "Timestamp": 1527026700.0,
            "Value": 1.3533333333333333
        },
        {
            "Timestamp": 1527027000.0,
            "Value": 0.88
        },
        <...remaining output omitted...>
    ]
},
{
    "Key": {
        "Metric": "db.load.avg",
        "Dimensions": {
            "db.wait_event.name": "wait/synch/mutex/innodb/
aurora_lock_thread_slot_futex"
        }
    },
    "DataPoints": [
        {
            "Timestamp": 1527026700.0,
            "Value": 0.8566666666666667
        },
        {
            "Timestamp": 1527027000.0,
            "Value": 0.8633333333333333
        },
        <...remaining output omitted...>
    ],
},
<...remaining output omitted...>
]
}

```

- Para API obtener más información, consulte [GetResourceMetrics](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon Redshift que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Amazon Redshift.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **accept-reserved-node-exchange**

En el siguiente ejemplo de código, se muestra cómo usar `accept-reserved-node-exchange`.

AWS CLI

Para aceptar el intercambio de nodos reservados

El siguiente `accept-reserved-node-exchange` ejemplo acepta el intercambio de un nodo DC1 reservado por un nodo DC2 reservado.

```
aws redshift accept-reserved-node-exchange /  
  --reserved-node-id 12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE /  
  --target-reserved-node-offering-id 12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE
```

Salida:

```
{  
  "ExchangedReservedNode": {  
    "ReservedNodeId": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",  
    "ReservedNodeOfferingId": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",  
    "NodeType": "dc2.large",  
    "StartTime": "2019-12-06T21:17:26Z",  
    "Duration": 31536000,  
    "FixedPrice": 0.0,  
    "UsagePrice": 0.0,  
    "CurrencyCode": "USD",  
    "NodeCount": 1,  
  }  
}
```

```

    "State": "exchanging",
    "OfferingType": "All Upfront",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": 0.0,
        "RecurringChargeFrequency": "Hourly"
      }
    ],
    "ReservedNodeOfferingType": "Regular"
  }
}

```

Para obtener más información, consulte [Actualización de nodos reservados con la Guía AWS CLI](#) de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [AcceptReservedNodeExchange](#) la Referencia de AWS CLI comandos.

## authorize-cluster-security-group-ingress

En el siguiente ejemplo de código, se muestra cómo usar `authorize-cluster-security-group-ingress`.

### AWS CLI

El GroupThis ejemplo de autorización de acceso a una EC2 seguridad autoriza el acceso a un grupo de EC2 seguridad de Amazon con nombre.Comando:

```
aws redshift authorize-cluster-security-group-ingress --cluster-security-group-name
mysecuritygroup --ec2-security-group-name myec2securitygroup --ec2-security-group-
owner-id 123445677890
```

Al autorizar el acceso a un CIDR rangeThis ejemplo, se autoriza el acceso a un rango. Comando:  
CIDR

```
aws redshift authorize-cluster-security-group-ingress --cluster-security-group-name
mysecuritygroup --cidrip 192.168.100.100/32
```

- Para obtener API más información, consulte la Referencia de comandos [AuthorizeClusterSecurityGroupIngress](#).AWS CLI

## authorize-snapshot-access

En el siguiente ejemplo de código, se muestra cómo usar `authorize-snapshot-access`.

### AWS CLI

Autorizar la restauración de una AWS cuenta: un SnapshotThis ejemplo autoriza a la AWS cuenta 444455556666 a restaurar la instantáneamy-snapshot-id. De forma predeterminada, la salida está en JSON formato.Comando:

```
aws redshift authorize-snapshot-access --snapshot-id my-snapshot-id --account-with-restore-access 444455556666
```

### Resultado:

```
{
  "Snapshot": {
    "Status": "available",
    "SnapshotCreateTime": "2013-07-17T22:04:18.947Z",
    "EstimatedSecondsToCompletion": 0,
    "AvailabilityZone": "us-east-1a",
    "ClusterVersion": "1.0",
    "MasterUsername": "adminuser",
    "Encrypted": false,
    "OwnerAccount": "111122223333",
    "BackupProgressInMegabytes": 11.0,
    "ElapsedTimeInSeconds": 0,
    "DBName": "dev",
    "CurrentBackupRateInMegabytesPerSecond": 0.1534,
    "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
    "ActualIncrementalBackupSizeInMegabytes"; 11.0,
    "SnapshotType": "manual",
    "NodeType": "dw.hs1.xlarge",
    "ClusterIdentifier": "mycluster",
    "TotalBackupSizeInMegabytes": 20.0,
    "Port": 5439,
    "NumberOfNodes": 2,
    "SnapshotIdentifier": "my-snapshot-id"
  }
}
```

- Para API obtener más información, consulte la Referencia [AuthorizeSnapshotAccess](#) de AWS CLI comandos.

## batch-delete-cluster-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `batch-delete-cluster-snapshots`.

### AWS CLI

Para eliminar un conjunto de instantáneas de un clúster

En el siguiente `batch-delete-cluster-snapshots` ejemplo, se elimina un conjunto de instantáneas de clústeres manuales.

```
aws redshift batch-delete-cluster-snapshots \
    --
    identifiers SnapshotIdentifier=mycluster-2019-11-06-14-12 SnapshotIdentifier=mycluster-2019-
```

Salida:

```
{
  "Resources": [
    "mycluster-2019-11-06-14-12",
    "mycluster-2019-11-06-14-20"
  ]
}
```

Para obtener más información, consulte [Amazon Redshift Snapshots](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia de [BatchDeleteClusterSnapshots](#) comandos AWS CLI .

## batch-modify-cluster-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `batch-modify-cluster-snapshots`.

### AWS CLI

Para modificar un conjunto de instantáneas de un clúster

El siguiente `batch-modify-cluster-snapshots` ejemplo modifica la configuración de un conjunto de instantáneas de clúster.

```
aws redshift batch-modify-cluster-snapshots \
```

```
--snapshot-identifier-list mycluster-2019-11-06-16-31 mycluster-2019-11-06-16-32  
\  
--manual-snapshot-retention-period 30
```

Salida:

```
{  
  "Resources": [  
    "mycluster-2019-11-06-16-31",  
    "mycluster-2019-11-06-16-32"  
  ],  
  "Errors": [],  
  "ResponseMetadata": {  
    "RequestId": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",  
    "HTTPStatusCode": 200,  
    "HTTPHeaders": {  
      "x-amzn-requestid": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",  
      "content-type": "text/xml",  
      "content-length": "480",  
      "date": "Sat, 07 Dec 2019 00:36:09 GMT",  
      "connection": "keep-alive"  
    },  
    "RetryAttempts": 0  
  }  
}
```

Para obtener más información, consulte [Amazon Redshift Snapshots](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia de [BatchModifyClusterSnapshots](#) comandos AWS CLI .

## cancel-resize

En el siguiente ejemplo de código, se muestra cómo usar `cancel-resize`.

### AWS CLI

Para cancelar el cambio de tamaño de un clúster

El siguiente `cancel-resize` ejemplo cancela una operación de cambio de tamaño clásica de un clúster.

```
aws redshift cancel-resize \  
  --cluster-identifier mycluster
```

Salida:

```
{  
  "TargetNodeType": "dc2.large",  
  "TargetNumberOfNodes": 2,  
  "TargetClusterType": "multi-node",  
  "Status": "CANCELLING",  
  "ResizeType": "ClassicResize",  
  "TargetEncryptionType": "NONE"  
}
```

Para obtener más información, consulte Cambiar [el tamaño de los clústeres en Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia [CancelResized](#) de AWS CLI comandos.

## copy-cluster-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `copy-cluster-snapshot`.

AWS CLI

El `VersionThis` ejemplo Obtener una descripción de todo el clúster devuelve una descripción de todas las versiones del clúster. De forma predeterminada, el resultado está en JSON formato.

Comando:

```
aws redshift copy-cluster-snapshot --source-snapshot-identifier  
  cm:examplecluster-2013-01-22-19-27-58 --target-snapshot-identifier my-saved-  
  snapshot-copy
```

Resultado:

```
{  
  "Snapshot": {  
    "Status": "available",  
    "SnapshotCreateTime": "2013-01-22T19:27:58.931Z",  
    "AvailabilityZone": "us-east-1c",
```



```

    "ClusterVersion": "1.0",
    "MasterUsername": "adminuser",
    "DBName": "dev",
    "ClusterCreateTime": "2013-01-22T19:23:59.368Z",
    "SnapshotType": "manual",
    "NodeType": "dw.hs1.xlarge",
    "ClusterIdentifier": "examplecluster",
    "Port": 5439,
    "NumberOfNodes": "2",
    "SnapshotIdentifier": "my-saved-snapshot-copy"
  },
  "ResponseMetadata": {
    "RequestId": "3b279691-64e3-11e2-bec0-17624ad140dd"
  }
}

```

- Para API obtener más información, consulte la Referencia [CopyClusterSnapshot](#) de AWS CLI comandos.

## create-cluster-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `create-cluster-parameter-group`.

### AWS CLI

El GroupThis ejemplo de creación de un parámetro de clúster crea un nuevo grupo de parámetros de clúster. Comando:

```

aws redshift create-cluster-parameter-group --parameter-group-name
myclusterparametergroup --parameter-group-family redshift-1.0 --description "My
first cluster parameter group"

```

### Resultado:

```

{
  "ClusterParameterGroup": {
    "ParameterGroupFamily": "redshift-1.0",
    "Description": "My first cluster parameter group",
    "ParameterGroupName": "myclusterparametergroup"
  },
  "ResponseMetadata": {
    "RequestId": "739448f0-64cc-11e2-8f7d-3b939af52818"
  }
}

```

```
}
}
```

- Para API obtener más información, consulte la Referencia [CreateClusterParameterGroup](#) de AWS CLI comandos.

## create-cluster-security-group

En el siguiente ejemplo de código, se muestra cómo usar `create-cluster-security-group`.

### AWS CLI

Al crear un GroupThis ejemplo de seguridad de clúster, se crea un nuevo grupo de seguridad de clúster. De forma predeterminada, el resultado está en JSON formato. Comando:

```
aws redshift create-cluster-security-group --cluster-security-group-name
mysecuritygroup --description "This is my cluster security group"
```

### Resultado:

```
{
  "create_cluster_security_group_response": {
    "create_cluster_security_group_result": {
      "cluster_security_group": {
        "description": "This is my cluster security group",
        "owner_id": "300454760768",
        "cluster_security_group_name": "mysecuritygroup",
        "ec2_security_groups": \[],
        "ip_ranges": \[]
      }
    },
    "response_metadata": {
      "request_id": "5df486a0-343a-11e2-b0d8-d15d0ef48549"
    }
  }
}
```

También puede obtener la misma información en formato de texto mediante la opción `--output text`. Comando:

`--output text option.Command:`

**option.Command:**

```
aws redshift create-cluster-security-group --cluster-security-group-name
mysecuritygroup --description "This is my cluster security group" --output text
```

**Resultado:**

```
This is my cluster security group 300454760768 mysecuritygroup
a0c0bfab-343a-11e2-95d2-c3dc9fe8ab57
```

- Para API obtener más información, consulte la Referencia [CreateClusterSecurityGroup](#) de AWS CLI comandos.

**create-cluster-snapshot**

En el siguiente ejemplo de código, se muestra cómo usar `create-cluster-snapshot`.

**AWS CLI**

El SnapshotThis ejemplo de creación de un clúster crea una nueva instantánea del clúster. De forma predeterminada, la salida está en JSON formato. Comando:

```
aws redshift create-cluster-snapshot --cluster-identifier mycluster --snapshot-
identifier my-snapshot-id
```

**Resultado:**

```
{
  "Snapshot": {
    "Status": "creating",
    "SnapshotCreateTime": "2013-01-22T22:20:33.548Z",
    "AvailabilityZone": "us-east-1a",
    "ClusterVersion": "1.0",
    "MasterUsername": "adminuser",
    "DBName": "dev",
    "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
    "SnapshotType": "manual",
    "NodeType": "dw.hs1.xlarge",
    "ClusterIdentifier": "mycluster",
    "Port": 5439,
```

```

    "NumberOfNodes": "2",
    "SnapshotIdentifier": "my-snapshot-id"
  },
  "ResponseMetadata": {
    "RequestId": "f024d1a5-64e1-11e2-88c5-53eb05787dfb"
  }
}

```

- Para API obtener más información, consulte la Referencia [CreateClusterSnapshot](#) de AWS CLI comandos.

## create-cluster-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `create-cluster-subnet-group`.

### AWS CLI

El GroupThis ejemplo de creación de una subred de clúster crea un nuevo grupo de subredes de clúster. Comando:

```
aws redshift create-cluster-subnet-group --cluster-subnet-group-name mysubnetgroup
--description "My subnet group" --subnet-ids subnet-763fdd1c
```

### Resultado:

```

{
  "ClusterSubnetGroup": {
    "Subnets": [
      {
        "SubnetStatus": "Active",
        "SubnetIdentifier": "subnet-763fdd1c",
        "SubnetAvailabilityZone": {
          "Name": "us-east-1a"
        }
      }
    ],
    "VpcId": "vpc-7e3fdd14",
    "SubnetGroupStatus": "Complete",
    "Description": "My subnet group",
    "ClusterSubnetGroupName": "mysubnetgroup"
  },
  "ResponseMetadata": {
    "RequestId": "500b8ce2-698f-11e2-9790-fd67517fb6fd"
  }
}

```

```
}
}
```

- Para obtener API más información, consulte la Referencia de comandos [CreateClusterSubnetGroup](#).AWS CLI

## create-cluster

En el siguiente ejemplo de código, se muestra cómo usar `create-cluster`.

### AWS CLI

El ParametersThis ejemplo Crear un clúster con un mínimo crea un clúster con el conjunto mínimo de parámetros. De forma predeterminada, la salida está en JSON formato.Comando:

```
aws redshift create-cluster --node-type dw.hs1.xlarge --number-of-nodes 2 --master-username adminuser --master-user-password TopSecret1 --cluster-identifier mycluster
```

### Resultado:

```
{
  "Cluster": {
    "NodeType": "dw.hs1.xlarge",
    "ClusterVersion": "1.0",
    "PubliclyAccessible": "true",
    "MasterUsername": "adminuser",
    "ClusterParameterGroups": [
      {
        "ParameterApplyStatus": "in-sync",
        "ParameterGroupName": "default.redshift-1.0"
      }
    ],
    "ClusterSecurityGroups": [
      {
        "Status": "active",
        "ClusterSecurityGroupName": "default"
      }
    ],
    "AllowVersionUpgrade": true,
    "VpcSecurityGroups": [],
    "PreferredMaintenanceWindow": "sat:03:30-sat:04:00",
    "AutomatedSnapshotRetentionPeriod": 1,
    "ClusterStatus": "creating",
    "ClusterIdentifier": "mycluster",
```

```

    "DBName": "dev",
    "NumberOfNodes": 2,
    "PendingModifiedValues": {
      "MasterUserPassword": "\*****"
    }
  },
  "ResponseMetadata": {
    "RequestId": "7cf4bcfc-64dd-11e2-bea9-49e0ce183f07"
  }
}

```

- Para API obtener más información, consulte la Referencia [CreateCluster](#) de AWS CLI comandos.

## create-event-subscription

En el siguiente ejemplo de código, se muestra cómo usar `create-event-subscription`.

### AWS CLI

Para crear una suscripción de notificaciones para un evento

En el siguiente `create-event-subscription` ejemplo, se crea una suscripción a notificaciones de eventos.

```

aws redshift create-event-subscription \
  --subscription-name mysubscription \
  --sns-topic-arn arn:aws:sns:us-west-2:123456789012:MySNStopic \
  --source-type cluster \
  --source-ids mycluster

```

Salida:

```

{
  "EventSubscription": {
    "CustomerAwsId": "123456789012",
    "CustSubscriptionId": "mysubscription",
    "SnsTopicArn": "arn:aws:sns:us-west-2:123456789012:MySNStopic",
    "Status": "active",
    "SubscriptionCreationTime": "2019-12-09T20:05:19.365Z",
    "SourceType": "cluster",
    "SourceIdsList": [

```

```

        "mycluster"
    ],
    "EventCategoriesList": [],
    "Severity": "INFO",
    "Enabled": true,
    "Tags": []
}
}

```

Para obtener más información, consulte [Suscripción a las notificaciones de eventos de Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia de [CreateEventSubscription](#) comandos AWS CLI .

## create-hsm-client-certificate

En el siguiente ejemplo de código, se muestra cómo usar `create-hsm-client-certificate`.

AWS CLI

Para crear un certificado de HSM cliente

El siguiente `create-hsm-client-certificate` ejemplo genera un certificado de HSM cliente que un clúster puede usar para conectarse a un HSM.

```

aws redshift create-hsm-client-certificate \
  --hsm-client-certificate-identifier myhsmclientcert

```

Salida:

```

{
  "HsmClientCertificate": {
    "HsmClientCertificateIdentifier": "myhsmclientcert",
    "HsmClientCertificatePublicKey": "-----BEGIN CERTIFICATE-----
MIICiEXAMPLECQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMakGA1UEBhMC
VVMxCzAJBgNVBAGTEXAMPLEEwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBA5TC01BTSBDb25EXAMPLEIwEAYDVQQDEw1UZXR0Q21sYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb2EXAMPLETEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMakGA1UEBhMCVVMxCzAJBgNVBEXAMPLEMRAwDgYD
EXAMPLETZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC01BTSBDb25z
b2x1MRIwEAEXAMPLEEw1UZXR0Q21sYWMxHzAdBgkqhkiG9w0BCQEWEG5vb251QGft

```

```

YXpvbi5jb20wgZ8wDQYJKEXAMPEAQEBBQADgY0AMIGJAOGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLyGVIk6EXAMPE3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugEXAMPEzZswY6786m86gpE
Ibb30hjZnczvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEEAMPEEAtCu4
nUhVVxYUEXAMPEh8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GEXAMPE10ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rEXAMPE=-----END CERTIFICATE-----\n",
  "Tags": []
}
}

```

Para obtener más información, consulte la [referencia de API permisos de Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia [CreateHsmClientCertificate](#) de AWS CLI comandos.

## create-hsm-configuration

En el siguiente ejemplo de código, se muestra cómo usar `create-hsm-configuration`.

### AWS CLI

Para crear una HSM configuración

En el siguiente `create-hsm-configuration` ejemplo, se crea la HSM configuración especificada que contiene la información necesaria para que un clúster almacene y utilice las claves de cifrado de la base de datos en un módulo de seguridad de hardware (HSM).

```

aws redshift create-hsm-configuration /
  --hsm-configuration-identifier myhsmconnection
  --description "My HSM connection"
  --hsm-ip-address 192.0.2.09
  --hsm-partition-name myhsmpartition /
  --hsm-partition-password A1b2c3d4 /
  --hsm-server-public-certificate myhsmclientcert

```

Salida:

```

{
  "HsmConfiguration": {

```



```

    "HsmConfigurationIdentifier": "myhsmconnection",
    "Description": "My HSM connection",
    "HsmIpAddress": "192.0.2.09",
    "HsmPartitionName": "myhsmpartition",
    "Tags": []
  }
}

```

- Para API obtener más información, consulte [CreateHsmConfiguration](#) la Referencia de AWS CLI comandos.

## create-snapshot-copy-grant

En el siguiente ejemplo de código, se muestra cómo usar `create-snapshot-copy-grant`.

### AWS CLI

Para crear una copia instantánea, conceda

En el siguiente `create-snapshot-copy-grant` ejemplo, se crea una concesión de copia instantánea y se cifran las instantáneas copiadas en una región de destino AWS .

```

aws redshift create-snapshot-copy-grant \
  --snapshot-copy-grant-name mynapshotcopygrantname

```

Salida:

```

{
  "SnapshotCopyGrant": {
    "SnapshotCopyGrantName": "mynapshotcopygrantname",
    "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/bPxRfih3yCo8nvbEXAMPLEKEY",
    "Tags": []
  }
}

```

Para obtener más información, consulte el [cifrado de bases de datos de Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia [CreateSnapshotCopyGrant](#) de AWS CLI comandos.

## create-snapshot-schedule

En el siguiente ejemplo de código, se muestra cómo usar `create-snapshot-schedule`.

### AWS CLI

Para crear un cronograma de instantáneas

En el siguiente `create-snapshot-schedule` ejemplo, se crea una programación de instantáneas con la descripción especificada y una frecuencia de cada 12 horas.

```
aws redshift create-snapshot-schedule \  
  --schedule-definitions "rate(12 hours)" \  
  --schedule-identifier mysnapshotschedule \  
  --schedule-description "My schedule description"
```

Salida:

```
{  
  "ScheduleDefinitions": [  
    "rate(12 hours)"  
  ],  
  "ScheduleIdentifier": "mysnapshotschedule",  
  "ScheduleDescription": "My schedule description",  
  "Tags": []  
}
```

Para obtener más información, consulte [Automated Snapshot Schedules](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [CreateSnapshotSchedule](#) la Referencia de AWS CLI comandos.

## create-tags

En el siguiente ejemplo de código, se muestra cómo usar `create-tags`.

### AWS CLI

Para crear etiquetas para un clúster

En el siguiente `create-tags` ejemplo, se agrega el par clave/valor de etiqueta especificado al clúster especificado.

```
aws redshift create-tags \  
  --resource-name arn:aws:redshift:us-west-2:123456789012:cluster:mycluster \  
  --tags "Key"="mytags","Value"="tag1"
```

Este comando no produce ninguna salida.

Para obtener más información, consulte [Etiquetado de recursos en Amazon Redshift en](#) la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia de [CreateTags](#) comandos AWS CLI .

## **delete-cluster-parameter-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-cluster-parameter-group`.

AWS CLI

En el GroupThis ejemplo de eliminación de un parámetro de clúster, se elimina un grupo de parámetros de clúster. Comando:

```
aws redshift delete-cluster-parameter-group --parameter-group-name  
myclusterparametergroup
```

- Para API obtener más información, consulte la Referencia de [DeleteClusterParameterGroup](#) comandos AWS CLI .

## **delete-cluster-security-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-cluster-security-group`.

AWS CLI

El GroupThis ejemplo de eliminación de un clúster de seguridad elimina un grupo de seguridad de clúster. Comando:

```
aws redshift delete-cluster-security-group --cluster-security-group-name  
mysecuritygroup
```

- Para API obtener más información, consulte la Referencia de [DeleteClusterSecurityGroup](#) comandos AWS CLI .

## delete-cluster-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `delete-cluster-snapshot`.

### AWS CLI

En el SnapshotThis ejemplo de eliminar un clúster, se elimina una instantánea de un clúster.

Comando:

```
aws redshift delete-cluster-snapshot --snapshot-identifier my-snapshot-id
```

- Para obtener API más información, consulte la Referencia de comandos [DeleteClusterSnapshot](#).AWS CLI

## delete-cluster-subnet-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-cluster-subnet-group`.

### AWS CLI

El GroupThis ejemplo de eliminar una subred de clúster elimina un grupo de subredes de clúster.Comando:

```
aws redshift delete-cluster-subnet-group --cluster-subnet-group-name mysubnetgroup
```

Resultado:

```
{
  "ResponseMetadata": {
    "RequestId": "253fbffd-6993-11e2-bc3a-47431073908a"
  }
}
```

- Para obtener API más información, consulte la Referencia de comandos. [DeleteClusterSubnetGroup](#)AWS CLI

## delete-cluster

En el siguiente ejemplo de código, se muestra cómo usar `delete-cluster`.

## AWS CLI

En el SnapshotThis ejemplo de eliminar un clúster sin un clúster final, se elimina un clúster y se fuerza la eliminación de los datos para que no se cree una instantánea final del clúster. Comando:

```
aws redshift delete-cluster --cluster-identifier mycluster --skip-final-cluster-snapshot
```

El SnapshotThis ejemplo Eliminar un clúster y permitir un clúster final elimina un clúster, pero especifica una instantánea final del clúster. Comando:

```
aws redshift delete-cluster --cluster-identifier mycluster --final-cluster-snapshot-identifier myfinalsnapshot
```

- Para obtener API más información, consulte la Referencia de comandos [DeleteCluster](#).AWS CLI

## delete-event-subscription

En el siguiente ejemplo de código, se muestra cómo usar delete-event-subscription.

### AWS CLI

Para eliminar la suscripción a un evento

En el siguiente delete-event-subscription ejemplo, se elimina la suscripción de notificaciones de eventos especificada.

```
aws redshift delete-event-subscription \  
  --subscription-name mysubscription
```

Este comando no produce ninguna salida.

Para obtener más información, consulte [Suscripción a las notificaciones de eventos de Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia de [DeleteEventSubscription](#) comandos AWS CLI .

## delete-hsm-client-certificate

En el siguiente ejemplo de código, se muestra cómo usar `delete-hsm-client-certificate`.

### AWS CLI

Para eliminar el certificado de HSM cliente

En el siguiente `delete-hsm-client-certificate` ejemplo, se elimina un certificado de HSM cliente.

```
aws redshift delete-hsm-client-certificate \  
  --hsm-client-certificate-identifier myhsmclientcert
```

Este comando no produce ninguna salida.

Para obtener más información, consulte la [referencia de API permisos de Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia [DeleteHsmClientCertificate](#) de AWS CLI comandos.

## delete-hsm-configuration

En el siguiente ejemplo de código, se muestra cómo usar `delete-hsm-configuration`.

### AWS CLI

Para eliminar una HSM configuración

En el siguiente `delete-hsm-configuration` ejemplo, se elimina la HSM configuración especificada de la AWS cuenta actual.

```
aws redshift delete-hsm-configuration \  
  --hsm-configuration-identifier myhsmconnection
```

Este comando no produce ninguna salida.

- Para API obtener más información, consulte [DeleteHsmConfiguration](#) la Referencia de AWS CLI comandos.

## delete-scheduled-action

En el siguiente ejemplo de código, se muestra cómo usar `delete-scheduled-action`.

### AWS CLI

Para eliminar una acción programada

En el siguiente `delete-scheduled-action` ejemplo, se elimina la acción programada especificada.

```
aws redshift delete-scheduled-action \  
  --scheduled-action-name myscheduledaction
```

Este comando no produce ninguna salida.

- Para API obtener más información, consulte [DeleteScheduledAction](#) la Referencia de AWS CLI comandos.

## delete-snapshot-copy-grant

En el siguiente ejemplo de código, se muestra cómo usar `delete-snapshot-copy-grant`.

### AWS CLI

Para eliminar una instantánea, copie, conceda

En el siguiente `delete-snapshot-copy-grant` ejemplo, se elimina la autorización de copia instantánea especificada.

```
aws redshift delete-snapshot-copy-grant \  
  --snapshot-copy-grant-name mysnapshotcopygrantname
```

Este comando no produce ninguna salida.

Para obtener más información, consulte el [cifrado de bases de datos de Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia [DeleteSnapshotCopyGrant](#) de AWS CLI comandos.

## delete-snapshot-schedule

En el siguiente ejemplo de código, se muestra cómo usar `delete-snapshot-schedule`.

### AWS CLI

Para eliminar la programación de instantáneas

En el siguiente `delete-snapshot-schedule` ejemplo, se elimina la programación de instantáneas especificada. Debe desasociar los clústeres antes de eliminar la programación.

```
aws redshift delete-snapshot-schedule \  
  --schedule-identifier mynapshotschedule
```

Este comando no produce ninguna salida.

Para obtener más información, consulte [Automated Snapshot Schedules](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [DeleteSnapshotSchedule](#) la Referencia de AWS CLI comandos.

## delete-tags

En el siguiente ejemplo de código, se muestra cómo usar `delete-tags`.

### AWS CLI

Para eliminar etiquetas de un clúster

En el siguiente `delete-tags` ejemplo, se eliminan las etiquetas con los nombres de clave especificados del clúster especificado.

```
aws redshift delete-tags \  
  --resource-name arn:aws:redshift:us-west-2:123456789012:cluster:mycluster \  
  --tag-keys "clustertagkey" "clustertagvalue"
```

Este comando no produce ninguna salida.

Para obtener más información, consulte [Etiquetado de recursos en Amazon Redshift en](#) la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia de [DeleteTags](#) comandos AWS CLI .



## describe-account-attributes

En el siguiente ejemplo de código, se muestra cómo usar `describe-account-attributes`.

### AWS CLI

Para describir los atributos de una AWS cuenta

En el siguiente `describe-account-attributes` ejemplo, se muestran los atributos asociados a la AWS cuenta que realiza la llamada.

```
aws redshift describe-account-attributes
```

Salida:

```
{
  "AccountAttributes": [
    {
      "AttributeName": "max-defer-maintenance-duration",
      "AttributeValues": [
        {
          "AttributeValue": "45"
        }
      ]
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeAccountAttributes](#) la Referencia de AWS CLI comandos.

## describe-cluster-db-revisions

En el siguiente ejemplo de código, se muestra cómo usar `describe-cluster-db-revisions`.

### AWS CLI

Para describir las revisiones de la base de datos de un clúster

El siguiente `describe-cluster-db-revisions` ejemplo muestra los detalles de una matriz de `ClusterDbRevision` objetos para el clúster especificado.

```
aws redshift describe-cluster-db-revisions \  
--cluster-identifier mycluster
```

Salida:

```
{  
  "ClusterDbRevisions": [  
    {  
      "ClusterIdentifier": "mycluster",  
      "CurrentDatabaseRevision": "11420",  
      "DatabaseRevisionReleaseDate": "2019-11-22T16:43:49.597Z",  
      "RevisionTargets": []  
    }  
  ]  
}
```

- Para API obtener más información, consulte [DescribeClusterDbRevisions](#) la Referencia de AWS CLI comandos.

## describe-cluster-parameter-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-cluster-parameter-groups`.

AWS CLI

El GroupsThis ejemplo Obtener una descripción de todos los parámetros del clúster devuelve una descripción de todos los grupos de parámetros del clúster de la cuenta, con los encabezados de las columnas. De forma predeterminada, la salida está en JSON formato.Comando:

```
aws redshift describe-cluster-parameter-groups
```

Resultado:

```
{  
  "ParameterGroups": [  
    {  
      "ParameterGroupFamily": "redshift-1.0",  
      "Description": "My first cluster parameter group",  
      "ParameterGroupName": "myclusterparametergroup"  
    } ],  
}
```

```
"ResponseMetadata": {
  "RequestId": "8ceb8f6f-64cc-11e2-bea9-49e0ce183f07"
}
```

También puede obtener la misma información en formato de texto mediante la opción `--output text`. Comando:

`--output text option.Command:`

`option.Command:`

```
aws redshift describe-cluster-parameter-groups --output text
```

Resultado:

```
redshift-1.0      My first cluster parameter group      myclusterparametergroup
RESPONSEMETADATA 9e665a36-64cc-11e2-8f7d-3b939af52818
```

- Para API obtener más información, consulte la Referencia [DescribeClusterParameterGroups](#) de AWS CLI comandos.

## describe-cluster-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-cluster-parameters`.

AWS CLI

Recuperar los parámetros de un GroupThis ejemplo de parámetro de clúster especificado recupera los parámetros del grupo de parámetros nombrado. De forma predeterminada, la salida está en JSON formato.Comando:

```
aws redshift describe-cluster-parameters --parameter-group-name
myclusterparametergroup
```

Resultado:

```
{
  "Parameters": [
    {
      "Description": "Sets the display format for date and time values.",
```

```

        "DataType": "string",
        "IsModifiable": true,
        "Source": "engine-default",
        "ParameterValue": "ISO, MDY",
        "ParameterName": "datestyle"
    },
    {
        "Description": "Sets the number of digits displayed for floating-point
values",
        "DataType": "integer",
        "IsModifiable": true,
        "AllowedValues": "-15-2",
        "Source": "engine-default",
        "ParameterValue": "0",
        "ParameterName": "extra_float_digits"
    },
    (...remaining output omitted...)
]
}

```

También puede obtener la misma información en formato de texto mediante la opción `--output text`. Comando:

`--output text option.Command:`

`option.Command:`

```
aws redshift describe-cluster-parameters --parameter-group-name
myclusterparametergroup --output text
```

Resultado:

```

RESPONSEMETADATA    cdac40aa-64cc-11e2-9e70-918437dd236d
Sets the display format for date and time values.  string True    engine-default
ISO, MDY        datestyle
Sets the number of digits displayed for floating-point values      integer True
-15-2  engine-default  0        extra_float_digits
This parameter applies a user-defined label to a group of queries that are run
during the same session..  string True    engine-default  default query_group
require ssl for all databaseconnections  boolean True    true,false    engine-
default  false  require_ssl
Sets the schema search order for names that are not schema-qualified.  string
True    engine-default  $user, public  search_path

```

```
Aborts any statement that takes over the specified number of milliseconds. integer
True engine-default 0 statement_timeout
wlm json configuration string True engine-default
\["query_concurrency":5]] wlm_json_configuration
```

- Para API obtener más información, consulte la Referencia [DescribeClusterParameters](#) de AWS CLI comandos.

## describe-cluster-security-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-cluster-security-groups`.

### AWS CLI

El `GroupsThis` ejemplo Obtener una descripción de toda la seguridad del clúster devuelve una descripción de todos los grupos de seguridad del clúster de la cuenta. De forma predeterminada, el resultado está en JSON formato. Comando:

```
aws redshift describe-cluster-security-groups
```

### Resultado:

```
{
  "ClusterSecurityGroups": [
    {
      "OwnerId": "100447751468",
      "Description": "default",
      "ClusterSecurityGroupName": "default",
      "EC2SecurityGroups": [],
      "IPRanges": [
        {
          "Status": "authorized",
          "CIDRIP": "0.0.0.0/0"
        }
      ]
    },
    {
      "OwnerId": "100447751468",
      "Description": "This is my cluster security group",
      "ClusterSecurityGroupName": "mysecuritygroup",
```

```

    "EC2SecurityGroups": \[],
    "IPRanges": \[]
  },
  (...remaining output omitted...)
]
}

```

- Para API obtener más información, consulte la Referencia [DescribeClusterSecurityGroups](#) de AWS CLI comandos.

## describe-cluster-snapshots

En el siguiente ejemplo de código, se muestra cómo usar `describe-cluster-snapshots`.

### AWS CLI

El SnapshotsThis ejemplo Obtener una descripción de todo el clúster devuelve una descripción de todas las instantáneas del clúster de la cuenta. De forma predeterminada, la salida está en JSON formato. Comando:

```
aws redshift describe-cluster-snapshots
```

### Resultado:

```

{
  "Snapshots": [
    {
      "Status": "available",
      "SnapshotCreateTime": "2013-07-17T22:02:22.852Z",
      "EstimatedSecondsToCompletion": -1,
      "AvailabilityZone": "us-east-1a",
      "ClusterVersion": "1.0",
      "MasterUsername": "adminuser",
      "Encrypted": false,
      "OwnerAccount": "111122223333",
      "BackupProgressInMegabytes": 20.0,
      "ElapsedTimeInSeconds": 0,
      "DBName": "dev",
      "CurrentBackupRateInMegabytesPerSecond": 0.0,
      "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
      "ActualIncrementalBackupSizeInMegabytes"; 20.0
      "SnapshotType": "automated",

```

```

    "NodeType": "dw.hs1.xlarge",
    "ClusterIdentifier": "mycluster",
    "Port": 5439,
    "TotalBackupSizeInMegabytes": 20.0,
    "NumberOfNodes": "2",
    "SnapshotIdentifier": "cm:mycluster-2013-01-22-22-04-18"
  },
  {
    "EstimatedSecondsToCompletion": 0,
    "OwnerAccount": "111122223333",
    "CurrentBackupRateInMegabytesPerSecond": 0.1534,
    "ActualIncrementalBackupSizeInMegabytes": 11.0,
    "NumberOfNodes": "2",
    "Status": "available",
    "ClusterVersion": "1.0",
    "MasterUsername": "adminuser",
    "AccountsWithRestoreAccess": [
      {
        "AccountID": "444455556666"
      }
    ],
    "TotalBackupSizeInMegabytes": 20.0,
    "DBName": "dev",
    "BackupProgressInMegabytes": 11.0,
    "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
    "ElapsedTimeInSeconds": 0,
    "ClusterIdentifier": "mycluster",
    "SnapshotCreateTime": "2013-07-17T22:04:18.947Z",
    "AvailabilityZone": "us-east-1a",
    "NodeType": "dw.hs1.xlarge",
    "Encrypted": false,
    "SnapshotType": "manual",
    "Port": 5439,
    "SnapshotIdentifier": "my-snapshot-id"
  }
]
}
(...remaining output omitted...)

```

- Para API obtener más información, consulte la Referencia [DescribeClusterSnapshots](#) de AWS CLI comandos.

## describe-cluster-subnet-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-cluster-subnet-groups`.

## AWS CLI

El GroupsThis ejemplo Obtener una descripción de toda la subred del clúster devuelve una descripción de todos los grupos de subredes del clúster. De forma predeterminada, el resultado está en JSON formato. Comando:

```
aws redshift describe-cluster-subnet-groups
```

Resultado:

```
{
  "ClusterSubnetGroups": [
    {
      "Subnets": [
        {
          "SubnetStatus": "Active",
          "SubnetIdentifier": "subnet-763fdd1c",
          "SubnetAvailabilityZone": {
            "Name": "us-east-1a"
          }
        }
      ],
      "VpcId": "vpc-7e3fdd14",
      "SubnetGroupStatus": "Complete",
      "Description": "My subnet group",
      "ClusterSubnetGroupName": "mysubnetgroup"
    }
  ],
  "ResponseMetadata": {
    "RequestId": "37fa8c89-6990-11e2-8f75-ab4018764c77"
  }
}
```

- Para API obtener más información, consulte la Referencia [DescribeClusterSubnetGroups](#) de AWS CLI comandos.

### **describe-cluster-tracks**

En el siguiente ejemplo de código, se muestra cómo usar `describe-cluster-tracks`.



## AWS CLI

Para describir las pistas del clúster

En el siguiente `describe-cluster-tracks` ejemplo, se muestran los detalles de las pistas de mantenimiento disponibles.

```
aws redshift describe-cluster-tracks \  
  --maintenance-track-name current
```

Salida:

```
{  
  "MaintenanceTracks": [  
    {  
      "MaintenanceTrackName": "current",  
      "DatabaseVersion": "1.0.11420",  
      "UpdateTargets": [  
        {  
          "MaintenanceTrackName": "preview_features",  
          "DatabaseVersion": "1.0.11746",  
          "SupportedOperations": [  
            {  
              "OperationName": "restore-from-cluster-snapshot"  
            }  
          ]  
        },  
        {  
          "MaintenanceTrackName": "trailing",  
          "DatabaseVersion": "1.0.11116",  
          "SupportedOperations": [  
            {  
              "OperationName": "restore-from-cluster-snapshot"  
            },  
            {  
              "OperationName": "modify-cluster"  
            }  
          ]  
        }  
      ]  
    }  
  ]  
}
```

Para obtener más información, consulte [Elegir las vías de mantenimiento de clústeres](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [DescribeClusterTracks](#) la Referencia de AWS CLI comandos.

## **describe-cluster-versions**

En el siguiente ejemplo de código, se muestra cómo usar `describe-cluster-versions`.

### AWS CLI

El `describe-cluster-versions` devuelve una descripción de todo el clúster. De forma predeterminada, el resultado está en JSON formato.

Comando:

```
aws redshift describe-cluster-versions
```

Resultado:

```
{
  "ClusterVersions": [
    {
      "ClusterVersion": "1.0",
      "Description": "Initial release",
      "ClusterParameterGroupFamily": "redshift-1.0"
    }
  ],
  "ResponseMetadata": {
    "RequestId": "16a53de3-64cc-11e2-bec0-17624ad140dd"
  }
}
```

- Para API obtener más información, consulte la Referencia [DescribeClusterVersions](#) de AWS CLI comandos.

## **describe-clusters**

En el siguiente ejemplo de código, se muestra cómo usar `describe-clusters`.

## AWS CLI

El ClustersThis ejemplo Obtener una descripción de todo devuelve una descripción de todos los clústeres de la cuenta. De forma predeterminada, la salida está en JSON formato.Comando:

```
aws redshift describe-clusters
```

Resultado:

```
{
  "Clusters": [
    {
      "NodeType": "dw.hs1.xlarge",
      "Endpoint": {
        "Port": 5439,
        "Address": "mycluster.coqoarplqhsn.us-east-1.redshift.amazonaws.com"
      },
      "ClusterVersion": "1.0",
      "PubliclyAccessible": "true",
      "MasterUsername": "adminuser",
      "ClusterParameterGroups": [
        {
          "ParameterApplyStatus": "in-sync",
          "ParameterGroupName": "default.redshift-1.0"
        }
      ],
      "ClusterSecurityGroups": [
        {
          "Status": "active",
          "ClusterSecurityGroupName": "default"
        }
      ],
      "AllowVersionUpgrade": true,
      "VpcSecurityGroups": [],
      "AvailabilityZone": "us-east-1a",
      "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
      "PreferredMaintenanceWindow": "sat:03:30-sat:04:00",
      "AutomatedSnapshotRetentionPeriod": 1,
      "ClusterStatus": "available",
      "ClusterIdentifier": "mycluster",
      "DBName": "dev",
      "NumberOfNodes": 2,
      "PendingModifiedValues": {}
    }
  ],
  "ResponseMetadata": {
```

```

    "RequestId": "65b71cac-64df-11e2-8f5b-e90bd6c77476"
  }
}

```

También puede obtener la misma información en formato de texto mediante la opción `--output text`. Comando:

`--output text` option.Command:

option.Command:

```
aws redshift describe-clusters --output text
```

Resultado:

```

dw.hs1.xlarge      1.0      true      adminuser      True      us-east-1a
2013-01-22T21:59:29.559Z      sat:03:30-sat:04:00      1      available
mycluster      dev      2
ENDPOINT      5439      mycluster.coqoarplqhsn.us-east-1.redshift.amazonaws.com
in-sync      default.redshift-1.0
active      default
PENDINGMODIFIEDVALUES
RESPONSEMETADATA      934281a8-64df-11e2-b07c-f7fbdd006c67

```

- Para API obtener más información, consulte la Referencia [DescribeClusters](#) de AWS CLI comandos.

## describe-default-cluster-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-default-cluster-parameters`.

### AWS CLI

El `Parameters` This ejemplo `Get a Description of Default Cluster` devuelve una descripción de los parámetros de clúster predeterminados de la `redshift-1.0` familia. De forma predeterminada, la salida está en JSON formato. Comando:

```
aws redshift describe-default-cluster-parameters --parameter-group-family
redshift-1.0
```

**Resultado:**

```
{
  "DefaultClusterParameters": {
    "ParameterGroupFamily": "redshift-1.0",
    "Parameters": [
      {
        "Description": "Sets the display format for date and time values.",
        "DataType": "string",
        "IsModifiable": true,
        "Source": "engine-default",
        "ParameterValue": "ISO, MDY",
        "ParameterName": "datestyle"
      },
      {
        "Description": "Sets the number of digits displayed for floating-point
values",
        "DataType": "integer",
        "IsModifiable": true,
        "AllowedValues": "-15-2",
        "Source": "engine-default",
        "ParameterValue": "0",
        "ParameterName": "extra_float_digits"
      },
      (...remaining output omitted...)
    ]
  }
}
```

Para ver una lista de familias de grupos de parámetros válidas, utilice el `describe-cluster-parameter-groups` comando.

`describe-cluster-parameter-groups` comando.

comando.

- Para API obtener más información, consulte [DescribeDefaultClusterParameters](#) la Referencia de AWS CLI comandos.

**describe-event-categories**

En el siguiente ejemplo de código, se muestra cómo usar `describe-event-categories`.

## AWS CLI

Para describir las categorías de eventos de un clúster

En el siguiente `describe-event-categories` ejemplo, se muestran los detalles de las categorías de eventos de un clúster.

```
aws redshift describe-event-categories \  
  --source-type cluster
```

Salida:

```
{  
  "EventCategoriesMapList": [  
    {  
      "SourceType": "cluster",  
      "Events": [  
        {  
          "EventId": "REDSHIFT-EVENT-2000",  
          "EventCategories": [  
            "management"  
          ],  
          "EventDescription": "Cluster <cluster name> created at <time in  
UTC>.",  
          "Severity": "INFO"  
        },  
        {  
          "EventId": "REDSHIFT-EVENT-2001",  
          "EventCategories": [  
            "management"  
          ],  
          "EventDescription": "Cluster <cluster name> deleted at <time in  
UTC>.",  
          "Severity": "INFO"  
        },  
        {  
          "EventId": "REDSHIFT-EVENT-3625",  
          "EventCategories": [  
            "monitoring"  
          ],  
          "EventDescription": "The cluster <cluster name> can't be resumed  
with its previous elastic network interface <ENI id>. We will allocate a new  
elastic network interface and associate it with the cluster node.",
```

```

    "Severity": "INFO"
  }
]
}

```

- Para API obtener más información, consulte [DescribeEventCategories](#) la Referencia de AWS CLI comandos.

## describe-event-subscriptions

En el siguiente ejemplo de código, se muestra cómo usar `describe-event-subscriptions`.

### AWS CLI

Para describir las suscripciones a eventos

En el siguiente `describe-event-subscriptions` ejemplo, se muestran las suscripciones de notificaciones de eventos de la suscripción especificada.

```

aws redshift describe-event-subscriptions \
  --subscription-name mysubscription

```

Salida:

```

{
  "EventSubscriptionsList": [
    {
      "CustomerAwsId": "123456789012",
      "CustSubscriptionId": "mysubscription",
      "SnsTopicArn": "arn:aws:sns:us-west-2:123456789012:MySNSStopic",
      "Status": "active",
      "SubscriptionCreationTime": "2019-12-09T21:50:21.332Z",
      "SourceIdsList": [],
      "EventCategoriesList": [
        "management"
      ],
      "Severity": "ERROR",
      "Enabled": true,
      "Tags": []
    }
  ]
}

```

```
    }  
  ]  
}
```

Para obtener más información, consulte [Suscripción a las notificaciones de eventos de Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia de [DescribeEventSubscriptions](#) comandos AWS CLI .

## describe-events

En el siguiente ejemplo de código, se muestra cómo usar `describe-events`.

### AWS CLI

Describe todos los eventos: este ejemplo devuelve todos los eventos. De forma predeterminada, la salida está en JSON formato. Comando:

```
aws redshift describe-events
```

### Resultado:

```
{  
  "Events": [  
    {  
      "Date": "2013-01-22T19:17:03.640Z",  
      "SourceIdentifier": "myclusterparametergroup",  
      "Message": "Cluster parameter group myclusterparametergroup has been  
created.",  
      "SourceType": "cluster-parameter-group"  
    } ],  
  "ResponseMetadata": {  
    "RequestId": "9f056111-64c9-11e2-9390-ff04f2c1e638"  
  }  
}
```

También puede obtener la misma información en formato de texto mediante la opción `--output text`. Comando:

`--output text option.Command:`



option.Command:

```
aws redshift describe-events --output text
```

Resultado:

```
2013-01-22T19:17:03.640Z    myclusterparametergroup Cluster parameter group
myclusterparametergroup has been created.    cluster-parameter-group
RESPONSEMETADATA    8e5fe765-64c9-11e2-bce3-e56f52c50e17
```

- Para API obtener más información, consulte la Referencia [DescribeEvents](#) de AWS CLI comandos.

## describe-hsm-client-certificates

En el siguiente ejemplo de código, se muestra cómo usar `describe-hsm-client-certificates`.

AWS CLI

Para describir los certificados de HSM cliente

En el siguiente `describe-hsm-client-certificates` ejemplo, se muestran los detalles del certificado de HSM cliente especificado.

```
aws redshift describe-hsm-client-certificates \
  --hsm-client-certificate-identifier myhsmclientcert
```

Salida:

```
{
  "HsmClientCertificates": [
    {
      "HsmClientCertificateIdentifier": "myhsmclientcert",
      "HsmClientCertificatePublicKey": "-----BEGIN CERTIFICATE-----\
EXAMPLECAfICCQD6m7oRw0uX0jANBqkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAEXAMPLERAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBA5TC01BTSBDb25zEXAMPLEwEAYDVQQDEw1UZXR0eWw1YWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFTYXpvi5jb20wHhEXAMPLEDI1MjA0EXAMPLEN
EXAMPLE0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
```

```

VQQHEwdTZWF0dGEXAMPLEQYDVQKewZBbWF6b24xFDASBgNVBA5TC01BTSBDb25z
b2x1MRIwEAYDVQQDEw1UZsXN0Q21sEXAMPLEdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIEXAMPLEMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLygVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY67EXAMPLEE
EXAMPLEZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9EXAMPLE6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDEXAMPLEBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rEXAMPLE=-----END CERTIFICATE-----\n",
  "Tags": []
}
]
}

```

Para obtener más información, consulte la [referencia de API permisos de Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia [DescribeHsmClientCertificates](#) de AWS CLI comandos.

## describe-hsm-configurations

En el siguiente ejemplo de código, se muestra cómo usar `describe-hsm-configurations`.

### AWS CLI

Para describir HSM las configuraciones

En el siguiente `describe-hsm-configurations` ejemplo, se muestran los detalles de las HSM configuraciones disponibles para la AWS cuenta que realiza la llamada.

```

aws redshift describe-hsm-configurations /
--hsm-configuration-identifier myhsmconnection

```

Salida:

```

{
  "HsmConfigurations": [
    {
      "HsmConfigurationIdentifier": "myhsmconnection",
      "Description": "My HSM connection",

```

```
        "HsmIpAddress": "192.0.2.09",
        "HsmPartitionName": "myhsmpartition",
        "Tags": []
    }
]
}
```

- Para API obtener más información, consulte [DescribeHsmConfigurations](#) la Referencia de AWS CLI comandos.

## describe-logging-status

En el siguiente ejemplo de código, se muestra cómo usar `describe-logging-status`.

### AWS CLI

Para describir el estado de registro de un clúster

En el siguiente `describe-logging-status` ejemplo, se muestra si la información, como las consultas y los intentos de conexión, se está registrando en un clúster.

```
aws redshift describe-logging-status \
  --cluster-identifier mycluster
```

Salida:

```
{
  "LoggingEnabled": false
}
```

Para obtener más información, consulte el [registro de auditorías de bases](#) de datos en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [DescribeLoggingStatus](#) la Referencia de AWS CLI comandos.

## describe-node-configuration-options

En el siguiente ejemplo de código, se muestra cómo usar `describe-node-configuration-options`.

## AWS CLI

Para describir las opciones de configuración de los nodos

En el siguiente `describe-node-configuration-options` ejemplo, se muestran las propiedades de las posibles configuraciones de nodos, como el tipo de nodo, el número de nodos y el uso del disco, para la instantánea del clúster especificada.

```
aws redshift describe-node-configuration-options \  
  --action-type restore-cluster \  
  --snapshot-identifier rs:mycluster-2019-12-09-16-42-43
```

Salida:

```
{  
  "NodeConfigurationOptionList": [  
    {  
      "NodeType": "dc2.large",  
      "NumberOfNodes": 2,  
      "EstimatedDiskUtilizationPercent": 19.61  
    },  
    {  
      "NodeType": "dc2.large",  
      "NumberOfNodes": 4,  
      "EstimatedDiskUtilizationPercent": 9.96  
    },  
    {  
      "NodeType": "ds2.xlarge",  
      "NumberOfNodes": 2,  
      "EstimatedDiskUtilizationPercent": 1.53  
    },  
    {  
      "NodeType": "ds2.xlarge",  
      "NumberOfNodes": 4,  
      "EstimatedDiskUtilizationPercent": 0.78  
    }  
  ]  
}
```

Para obtener más información, consulte [Adquisición de nodos reservados de Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia [DescribeNodeConfigurationOptions](#) de AWS CLI comandos.

## describe-orderable-cluster-options

En el siguiente ejemplo de código, se muestra cómo usar `describe-orderable-cluster-options`.

### AWS CLI

El `OptionsThis` ejemplo Describir todos los clústeres ordenables devuelve descripciones de todas las opciones de clústeres ordenables. De forma predeterminada, la salida está en JSON formato. Comando:

```
aws redshift describe-orderable-cluster-options
```

### Resultado:

```
{
  "OrderableClusterOptions": [
    {
      "NodeType": "dw.hs1.8xlarge",
      "AvailabilityZones": [
        { "Name": "us-east-1a" },
        { "Name": "us-east-1b" },
        { "Name": "us-east-1c" } ],
      "ClusterVersion": "1.0",
      "ClusterType": "multi-node"
    },
    {
      "NodeType": "dw.hs1.xlarge",
      "AvailabilityZones": [
        { "Name": "us-east-1a" },
        { "Name": "us-east-1b" },
        { "Name": "us-east-1c" } ],
      "ClusterVersion": "1.0",
      "ClusterType": "multi-node"
    },
    {
      "NodeType": "dw.hs1.xlarge",
      "AvailabilityZones": [
        { "Name": "us-east-1a" },
```

```

    { "Name": "us-east-1b" },
    { "Name": "us-east-1c" } ],
  "ClusterVersion": "1.0",
  "ClusterType": "single-node"
} ],
"ResponseMetadata": {
  "RequestId": "f6000035-64cb-11e2-9135-ff82df53a51a"
}
}

```

También puede obtener la misma información en formato de texto mediante la opción `--output text`. Comando:

`--output text option.Command:`

`option.Command:`

```
aws redshift describe-orderable-cluster-options --output text
```

Resultado:

```

dw.hs1.8xlarge      1.0      multi-node
us-east-1a
us-east-1b
us-east-1c
dw.hs1.xlarge      1.0      multi-node
us-east-1a
us-east-1b
us-east-1c
dw.hs1.xlarge      1.0      single-node
us-east-1a
us-east-1b
us-east-1c
RESPONSEMETADATA  e648696b-64cb-11e2-bec0-17624ad140dd

```

- Para API obtener más información, consulte la Referencia [DescribeOrderableClusterOptions](#) de AWS CLI comandos.

## describe-reserved-node-offerings

En el siguiente ejemplo de código, se muestra cómo usar `describe-reserved-node-offerings`.

## AWS CLI

El `OfferingsThis` ejemplo Describe un nodo reservado muestra todas las ofertas de nodos reservados que están disponibles para su compra. Comando:

```
aws redshift describe-reserved-node-offerings
```

Resultado:

```
{
  "ReservedNodeOfferings": [
    {
      "OfferingType": "Heavy Utilization",
      "FixedPrice": "",
      "NodeType": "dw.hs1.xlarge",
      "UsagePrice": "",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": "",
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "Duration": 31536000,
      "ReservedNodeOfferingId": "ceb6a579-cf4c-4343-be8b-d832c45ab51c"
    },
    {
      "OfferingType": "Heavy Utilization",
      "FixedPrice": "",
      "NodeType": "dw.hs1.8xlarge",
      "UsagePrice": "",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": "",
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "Duration": 31536000,
      "ReservedNodeOfferingId": "e5a2ff3b-352d-4a9c-ad7d-373c4cab5dd2"
    },
    ...remaining output omitted...
  ],
  "ResponseMetadata": {
    "RequestId": "8b1a1a43-75ff-11e2-9666-e142fe91ddd1"
  }
}
```

Si desea comprar una oferta de nodo reservado, puede llamar `purchase-reserved-node-offering` utilizando una oferta válida. `ReservedNodeOfferingId`

`purchase-reserved-node-offering` utilizando una válida `ReservedNodeOfferingId`.

utilizando un válido `ReservedNodeOfferingId`.

`ReservedNodeOfferingId`.

- Para API obtener más información, consulte [DescribeReservedNodeOfferings](#) la Referencia de AWS CLI comandos.

## **describe-reserved-nodes**

En el siguiente ejemplo de código, se muestra cómo usar `describe-reserved-nodes`.

### AWS CLI

El `NodesThis` ejemplo `Describe Reserved` muestra una oferta de nodo reservado que se ha comprado. Comando:

```
aws redshift describe-reserved-nodes
```

### Resultado:

```
{
  "ResponseMetadata": {
    "RequestId": "bc29ce2e-7600-11e2-9949-4b361e7420b7"
  },
  "ReservedNodes": [
    {
      "OfferingType": "Heavy Utilization",
      "FixedPrice": "",
      "NodeType": "dw.hs1.xlarge",
      "ReservedNodeId": "1ba8e2e3-bc01-4d65-b35d-a4a3e931547e",
      "UsagePrice": "",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": "",
          "RecurringChargeFrequency": "Hourly"
        }
      ]
    }
  ],
}
```



```
        "NodeCount": 1,
        "State": "payment-pending",
        "StartTime": "2013-02-13T17:08:39.051Z",
        "Duration": 31536000,
        "ReservedNodeOfferingId": "ceb6a579-cf4c-4343-be8b-d832c45ab51c"
    }
]
}
```

- Para obtener API más información, consulte la Referencia de comandos [DescribeReservedNodes](#).AWS CLI

## describe-resize

En el siguiente ejemplo de código, se muestra cómo usar `describe-resize`.

### AWS CLI

El `ResizeThis` ejemplo `Describe` describe el último cambio de tamaño de un clúster. La solicitud era para 3 nodos del tipo `dw.hs1.8xlarge`. `Command`:

```
aws redshift describe-resize --cluster-identifier mycluster
```

### Resultado:

```
{
  "Status": "NONE",
  "TargetClusterType": "multi-node",
  "TargetNodeType": "dw.hs1.8xlarge",
  "ResponseMetadata": {
    "RequestId": "9f52b0b4-7733-11e2-aa9b-318b2909bd27"
  },
  "TargetNumberOfNodes": "3"
}
```

- Para API obtener más información, consulte [DescribeResizela](#) Referencia de AWS CLI comandos.

## describe-scheduled-actions

En el siguiente ejemplo de código, se muestra cómo usar `describe-scheduled-actions`.

## AWS CLI

Para describir las acciones programadas

En el siguiente `describe-scheduled-actions` ejemplo, se muestran los detalles de las acciones programadas actualmente.

```
aws redshift describe-scheduled-actions
```

Salida:

```
{
  "ScheduledActions": [
    {
      "ScheduledActionName": "resizecluster",
      "TargetAction": {
        "ResizeCluster": {
          "ClusterIdentifier": "mycluster",
          "NumberOfNodes": 4,
          "Classic": false
        }
      },
      "Schedule": "at(2019-12-10T00:07:00)",
      "IamRole": "arn:aws:iam::123456789012:role/myRedshiftRole",
      "State": "ACTIVE",
      "NextInvocations": [
        "2019-12-10T00:07:00Z"
      ]
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeScheduledActions](#) la Referencia de AWS CLI comandos.

## `describe-snapshot-copy-grants`

En el siguiente ejemplo de código, se muestra cómo usar `describe-snapshot-copy-grants`.

## AWS CLI

Para describir las concesiones de copias instantáneas

En el siguiente `describe-snapshot-copy-grants` ejemplo, se muestran los detalles de la concesión de copia de instantáneas del clúster especificada.

```
aws redshift describe-snapshot-copy-grants \  
  --snapshot-copy-grant-name mysnapshotcopygrantname
```

Salida:

```
{  
  "SnapshotCopyGrants": [  
    {  
      "SnapshotCopyGrantName": "mysnapshotcopygrantname",  
      "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/  
bPxRfih3yCo8nvbEXAMPLEKEY",  
      "Tags": []  
    }  
  ]  
}
```

Para obtener más información, consulte el [cifrado de bases de datos de Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia [DescribeSnapshotCopyGrants](#) de AWS CLI comandos.

## **describe-snapshot-schedules**

En el siguiente ejemplo de código, se muestra cómo usar `describe-snapshot-schedules`.

AWS CLI

Para describir los cronogramas de las instantáneas

En el siguiente `describe-snapshot-schedules` ejemplo, se muestran los detalles del programa de instantáneas del clúster especificado.

```
aws redshift describe-snapshot-schedules \  
  --cluster-identifier mycluster \  
  --schedule-identifier mysnapshotschedule
```

Salida:

```
{
  "SnapshotSchedules": [
    {
      "ScheduleDefinitions": [
        "rate(12 hours)"
      ],
      "ScheduleIdentifier": "mysnapshotschedule",
      "ScheduleDescription": "My schedule description",
      "Tags": [],
      "AssociatedClusterCount": 1,
      "AssociatedClusters": [
        {
          "ClusterIdentifier": "mycluster",
          "ScheduleAssociationState": "ACTIVE"
        }
      ]
    }
  ]
}
```

Para obtener más información, consulte [Automated Snapshot Schedules](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [DescribeSnapshotSchedules](#) la Referencia de AWS CLI comandos.

## describe-storage

En el siguiente ejemplo de código, se muestra cómo usar `describe-storage`.

### AWS CLI

Para describir el almacenamiento

En el siguiente `describe-storage` ejemplo, se muestran detalles sobre el almacenamiento de copias de seguridad y los tamaños de almacenamiento provisional de la cuenta.

```
aws redshift describe-storage
```

Salida:

```
{
```

```
"TotalBackupSizeInMegaBytes": 193149.0,  
"TotalProvisionedStorageInMegaBytes": 655360.0  
}
```

Para obtener más información, consulte [Administrar el almacenamiento de instantáneas](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [DescribeStorage](#) la Referencia de AWS CLI comandos.

## describe-table-restore-status

En el siguiente ejemplo de código, se muestra cómo usar `describe-table-restore-status`.

### AWS CLI

Para describir el estado de las solicitudes de restauración de tablas desde una instantánea de clúster

En el siguiente `describe-table-restore-status` ejemplo, se muestran los detalles de las solicitudes de restauración de tablas realizadas para el clúster especificado.

```
aws redshift describe-table-restore-status /  
--cluster-identifier mycluster
```

Salida:

```
{  
  "TableRestoreStatusDetails": [  
    {  
      "TableRestoreRequestId": "z1116630-0e80-46f4-ba86-bd9670411ebd",  
      "Status": "IN_PROGRESS",  
      "RequestTime": "2019-12-27T18:22:12.257Z",  
      "ClusterIdentifier": "mycluster",  
      "SnapshotIdentifier": "mysnapshotid",  
      "SourceDatabaseName": "dev",  
      "SourceSchemaName": "public",  
      "SourceTableName": "mytable",  
      "TargetDatabaseName": "dev",  
      "TargetSchemaName": "public",  
      "NewTableName": "mytable-clone"  
    }  
  ]  
}
```

```
    }
  ]
}
```

Para obtener más información, consulte [Restauración de una tabla a partir de una instantánea](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [DescribeTableRestoreStatus](#) la Referencia de AWS CLI comandos.

## describe-tags

En el siguiente ejemplo de código, se muestra cómo usar `describe-tags`.

### AWS CLI

Para describir las etiquetas

En el siguiente `describe-tags` ejemplo, se muestran los recursos que el clúster especificado asoció a los nombres y valores de las etiquetas especificadas.

```
aws redshift describe-tags \
  --resource-name arn:aws:redshift:us-west-2:123456789012:cluster:mycluster \
  --tag-keys clustertagkey \
  --tag-values clustertagvalue
```

Salida:

```
{
  "TaggedResources": [
    {
      "Tag": {
        "Key": "clustertagkey",
        "Value": "clustertagvalue"
      },
      "ResourceName": "arn:aws:redshift:us-
west-2:123456789012:cluster:mycluster",
      "ResourceType": "cluster"
    }
  ]
}
```

Para obtener más información, consulte [Etiquetado de recursos en Amazon Redshift en](#) la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia de [DescribeTags](#) comandos AWS CLI .

## disable-snapshot-copy

En el siguiente ejemplo de código, se muestra cómo usar `disable-snapshot-copy`.

### AWS CLI

Para deshabilitar la copia instantánea de un clúster

El siguiente `disable-snapshot-copy` ejemplo deshabilita la copia automática de una instantánea para el clúster especificado.

```
aws redshift disable-snapshot-copy \  
  --cluster-identifier mycluster
```

Salida:

```
{  
  "Cluster": {  
    "ClusterIdentifier": "mycluster",  
    "NodeType": "dc2.large",  
    "ClusterStatus": "available",  
    "ClusterAvailabilityStatus": "Available",  
    "MasterUsername": "adminuser",  
    "DBName": "dev",  
    "Endpoint": {  
      "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",  
      "Port": 5439  
    },  
    "ClusterCreateTime": "2019-12-05T18:44:36.991Z",  
    "AutomatedSnapshotRetentionPeriod": 3,  
    "ManualSnapshotRetentionPeriod": -1,  
    "ClusterSecurityGroups": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sh-i9b431cd",  
        "Status": "active"  
      }  
    ]  
  }  
}
```

```
    ],
    "ClusterParameterGroups": [
      {
        "ParameterGroupName": "default.redshift-1.0",
        "ParameterApplyStatus": "in-sync"
      }
    ],
    "ClusterSubnetGroupName": "default",
    "VpcId": "vpc-b1fel7t9",
    "AvailabilityZone": "us-west-2f",
    "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
    "PendingModifiedValues": {
      "NodeType": "dc2.large",
      "NumberOfNodes": 2,
      "ClusterType": "multi-node"
    },
    "ClusterVersion": "1.0",
    "AllowVersionUpgrade": true,
    "NumberOfNodes": 4,
    "PubliclyAccessible": false,
    "Encrypted": false,
    "Tags": [
      {
        "Key": "mytags",
        "Value": "tag1"
      }
    ],
    "EnhancedVpcRouting": false,
    "IamRoles": [
      {
        "IamRoleArn": "arn:aws:iam::123456789012:role/myRedshiftRole",
        "ApplyStatus": "in-sync"
      }
    ],
    "MaintenanceTrackName": "current",
    "DeferredMaintenanceWindows": [],
    "ExpectedNextSnapshotScheduleTime": "2019-12-10T04:42:43.390Z",
    "ExpectedNextSnapshotScheduleTimeStatus": "OnTrack",
    "NextMaintenanceWindowStartTime": "2019-12-14T16:00:00Z"
  }
}
```



Para obtener más información, consulte [Copiar instantáneas a otra AWS región en la Guía de administración de clústeres de Amazon Redshift](#).

- Para API obtener más información, consulte la Referencia [DisableSnapshotCopy](#) de AWS CLI comandos.

## enable-snapshot-copy

En el siguiente ejemplo de código, se muestra cómo usar `enable-snapshot-copy`.

### AWS CLI

Para habilitar la copia instantánea de un clúster

El siguiente `enable-snapshot-copy` ejemplo habilita la copia automática de una instantánea para el clúster especificado.

```
aws redshift enable-snapshot-copy \  
  --cluster-identifier mycluster \  
  --destination-region us-west-1
```

Salida:

```
{  
  "Cluster": {  
    "ClusterIdentifier": "mycluster",  
    "NodeType": "dc2.large",  
    "ClusterStatus": "available",  
    "ClusterAvailabilityStatus": "Available",  
    "MasterUsername": "adminuser",  
    "DBName": "dev",  
    "Endpoint": {  
      "Address": "mycluster.cmeaswquae.us-west-2.redshift.amazonaws.com",  
      "Port": 5439  
    },  
    "ClusterCreateTime": "2019-12-05T18:44:36.991Z",  
    "AutomatedSnapshotRetentionPeriod": 3,  
    "ManualSnapshotRetentionPeriod": -1,  
    "ClusterSecurityGroups": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sh-f4c731cd",  
        "Status": "active"  
      }  
    ]  
  }  
}
```

```
    }
  ],
  "ClusterParameterGroups": [
    {
      "ParameterGroupName": "default.redshift-1.0",
      "ParameterApplyStatus": "in-sync"
    }
  ],
  "ClusterSubnetGroupName": "default",
  "VpcId": "vpc-b1ael7t9",
  "AvailabilityZone": "us-west-2f",
  "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
  "PendingModifiedValues": {
    "NodeType": "dc2.large",
    "NumberOfNodes": 2,
    "ClusterType": "multi-node"
  },
  "ClusterVersion": "1.0",
  "AllowVersionUpgrade": true,
  "NumberOfNodes": 4,
  "PubliclyAccessible": false,
  "Encrypted": false,
  "ClusterSnapshotCopyStatus": {
    "DestinationRegion": "us-west-1",
    "RetentionPeriod": 7,
    "ManualSnapshotRetentionPeriod": -1
  },
  "Tags": [
    {
      "Key": "mytags",
      "Value": "tag1"
    }
  ],
  "EnhancedVpcRouting": false,
  "IamRoles": [
    {
      "IamRoleArn": "arn:aws:iam::123456789012:role/myRedshiftRole",
      "ApplyStatus": "in-sync"
    }
  ],
  "MaintenanceTrackName": "current",
  "DeferredMaintenanceWindows": [],
  "ExpectedNextSnapshotScheduleTime": "2019-12-10T04:42:43.390Z",
  "ExpectedNextSnapshotScheduleTimeStatus": "OnTrack",
```

```

    "NextMaintenanceWindowStartTime": "2019-12-14T16:00:00Z"
  }
}

```

Para obtener más información, consulte [Copiar instantáneas a otra AWS región en la Guía](#) de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia [EnableSnapshotCopy](#) de AWS CLI comandos.

## get-cluster-credentials

En el siguiente ejemplo de código, se muestra cómo usar `get-cluster-credentials`.

### AWS CLI

Para obtener las credenciales de clúster de una AWS cuenta

El siguiente `get-cluster-credentials` ejemplo recupera las credenciales temporales que permiten el acceso a una base de datos de Amazon Redshift.

```

aws redshift get-cluster-credentials \
  --db-user adminuser --db-name dev \
  --cluster-identifier mycluster

```

Salida:

```

{
  "DbUser": "IAM:adminuser",
  "DbPassword": "AMAFUyyuros/QjxPTtgzcsuQsqzIasdzJEN04aCtWDzXx109d6UmpkBtvEqFly/
EXAMPLE==",
  "Expiration": "2019-12-10T17:25:05.770Z"
}

```

Para obtener más información, consulte [Generación de credenciales IAM de bases de datos con Amazon Redshift CLI o API](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia [GetClusterCredentials](#) de AWS CLI comandos.

## get-reserved-node-exchange-offerings

En el siguiente ejemplo de código, se muestra cómo usar `get-reserved-node-exchange-offerings`.

### AWS CLI

Para obtener ofertas de intercambio de nodos reservados

El siguiente `get-reserved-node-exchange-offerings` ejemplo recupera una matriz DC2 `ReservedNodeOfferings` que coincide con el nodo DC1 reservado especificado.

```
aws redshift get-reserved-node-exchange-offerings \
  --reserved-node-id 12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE
```

Salida:

```
{
  "ReservedNodeOfferings": [
    {
      "ReservedNodeOfferingId": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",
      "NodeType": "dc2.large",
      "Duration": 31536000,
      "FixedPrice": 0.0,
      "UsagePrice": 0.0,
      "CurrencyCode": "USD",
      "OfferingType": "All Upfront",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": 0.0,
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "ReservedNodeOfferingType": "Regular"
    }
  ]
}
```

Para obtener más información, consulte [Actualización de nodos reservados con la Guía AWS CLI de administración de clústeres de Amazon Redshift](#).

- Para API obtener más información, consulte [GetReservedNodeExchangeOfferings](#) la Referencia de AWS CLI comandos.

## modify-cluster-iam-roles

En el siguiente ejemplo de código, se muestra cómo usar `modify-cluster-iam-roles`.

### AWS CLI

Para modificar la IAM función de un clúster

En el siguiente `modify-cluster-iam-roles` ejemplo, se quita el AWS IAM rol especificado del clúster especificado.

```
aws redshift modify-cluster-iam-roles \  
  --cluster-identifier mycluster \  
  --remove-iam-roles arn:aws:iam::123456789012:role/myRedshiftRole
```

Salida:

```
{  
  "Cluster": {  
    "ClusterIdentifier": "mycluster",  
    "NodeType": "dc2.large",  
    "ClusterStatus": "available",  
    "ClusterAvailabilityStatus": "Available",  
    "MasterUsername": "adminuser",  
    "DBName": "dev",  
    "Endpoint": {  
      "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",  
      "Port": 5439  
    },  
    "ClusterCreateTime": "2019-12-05T18:44:36.991Z",  
    "AutomatedSnapshotRetentionPeriod": 3,  
    "ManualSnapshotRetentionPeriod": -1,  
    "ClusterSecurityGroups": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sh-f9b731sd",  
        "Status": "active"  
      }  
    ],  
    "ClusterParameterGroups": [  
      {  
        "ParameterGroupName": "default.redshift-1.0",  
        "ParameterApplyStatus": "in-sync"  
      }  
    ]  
  }  
}
```

```
    }
  ],
  "ClusterSubnetGroupName": "default",
  "VpcId": "vpc-b2fal7t9",
  "AvailabilityZone": "us-west-2f",
  "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
  "PendingModifiedValues": {
    "NodeType": "dc2.large",
    "NumberOfNodes": 2,
    "ClusterType": "multi-node"
  },
  "ClusterVersion": "1.0",
  "AllowVersionUpgrade": true,
  "NumberOfNodes": 4,
  "PubliclyAccessible": false,
  "Encrypted": false,
  "ClusterSnapshotCopyStatus": {
    "DestinationRegion": "us-west-1",
    "RetentionPeriod": 7,
    "ManualSnapshotRetentionPeriod": -1
  },
  "Tags": [
    {
      "Key": "mytags",
      "Value": "tag1"
    }
  ],
  "EnhancedVpcRouting": false,
  "IamRoles": [],
  "MaintenanceTrackName": "current",
  "DeferredMaintenanceWindows": [],
  "ExpectedNextSnapshotScheduleTime": "2019-12-11T04:42:55.631Z",
  "ExpectedNextSnapshotScheduleTimeStatus": "OnTrack",
  "NextMaintenanceWindowStartTime": "2019-12-14T16:00:00Z"
}
}
```

Para obtener más información, consulte [Uso de políticas basadas en la identidad \(IAMpolíticas\) para Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para obtener API más información, consulte la Referencia de comandos [ModifyClusterIamRoles](#).AWS CLI

## modify-cluster-maintenance

En el siguiente ejemplo de código, se muestra cómo usar `modify-cluster-maintenance`.

### AWS CLI

Para modificar el mantenimiento del clúster

El siguiente `modify-cluster-maintenance` ejemplo aplaza 30 días el mantenimiento del clúster especificado.

```
aws redshift modify-cluster-maintenance \  
  --cluster-identifier mycluster \  
  --defer-maintenance \  
  --defer-maintenance-duration 30
```

Salida:

```
{  
  "Cluster": {  
    "ClusterIdentifier": "mycluster",  
    "NodeType": "dc2.large",  
    "ClusterStatus": "available",  
    "ClusterAvailabilityStatus": "Available",  
    "MasterUsername": "adminuser",  
    "DBName": "dev",  
    "Endpoint": {  
      "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",  
      "Port": 5439  
    },  
    "ClusterCreateTime": "2019-12-05T18:44:36.991Z",  
    "AutomatedSnapshotRetentionPeriod": 3,  
    "ManualSnapshotRetentionPeriod": -1,  
    "ClusterSecurityGroups": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sh-a1a123ab",  
        "Status": "active"  
      }  
    ],  
    "ClusterParameterGroups": [  
      {  
        "ParameterGroupName": "default.redshift-1.0",
```

```
        "ParameterApplyStatus": "in-sync"
    }
],
"ClusterSubnetGroupName": "default",
"VpcId": "vpc-b1ael7t9",
"AvailabilityZone": "us-west-2f",
"PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
"PendingModifiedValues": {
    "NodeType": "dc2.large",
    "NumberOfNodes": 2,
    "ClusterType": "multi-node"
},
"ClusterVersion": "1.0",
"AllowVersionUpgrade": true,
"NumberOfNodes": 4,
"PubliclyAccessible": false,
"Encrypted": false,
"ClusterSnapshotCopyStatus": {
    "DestinationRegion": "us-west-1",
    "RetentionPeriod": 7,
    "ManualSnapshotRetentionPeriod": -1
},
"Tags": [
    {
        "Key": "mytags",
        "Value": "tag1"
    }
],
"EnhancedVpcRouting": false,
"IamRoles": [],
"MaintenanceTrackName": "current",
"DeferredMaintenanceWindows": [
    {
        "DeferMaintenanceIdentifier": "dfm-mUdVIffFcT1B4SGhw6fyF",
        "DeferMaintenanceStartTime": "2019-12-10T18:18:39.354Z",
        "DeferMaintenanceEndTime": "2020-01-09T18:18:39.354Z"
    }
],
"ExpectedNextSnapshotScheduleTime": "2019-12-11T04:42:55.631Z",
"ExpectedNextSnapshotScheduleTimeStatus": "OnTrack",
"NextMaintenanceWindowStartTime": "2020-01-11T16:00:00Z"
}
}
```



Para obtener más información, consulte [Mantenimiento de clústeres](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [ModifyClusterMaintenance](#) la Referencia de AWS CLI comandos.

## modify-cluster-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `modify-cluster-parameter-group`.

### AWS CLI

Modifique un parámetro de un grupo de parámetros

El siguiente `modify-cluster-parameter-group` ejemplo modifica el parámetro `wlm_json_configuration` para la administración de la carga de trabajo. Acepta los parámetros de un archivo que contiene el contenido que se muestra a continuación. JSON

```
aws redshift modify-cluster-parameter-group \
  --parameter-group-name myclusterparametergroup \
  --parameters file://modify_pg.json
```

Contenidos de `modify_pg.json`:

```
[
  {
    "ParameterName": "wlm_json_configuration",
    "ParameterValue": "[{\"user_group\":\"example_user_group1\",\"query_group\":\
\"example_query_group1\", \"query_concurrency\":7},{\"query_concurrency\":5}]"
  }
]
```

Salida:

```
{
  "ParameterGroupStatus": "Your parameter group has been updated but changes won't
get applied until you reboot the associated Clusters.",
  "ParameterGroupName": "myclusterparametergroup",
  "ResponseMetadata": {
    "RequestId": "09974cc0-64cd-11e2-bea9-49e0ce183f07"
  }
}
```

```
}
```

- Para API obtener más información, consulte [ModifyClusterParameterGroup](#) la Referencia de AWS CLI comandos.

## **modify-cluster-snapshot-schedule**

En el siguiente ejemplo de código, se muestra cómo usar `modify-cluster-snapshot-schedule`.

### AWS CLI

Para modificar la programación de instantáneas del clúster

En el siguiente `modify-cluster-snapshot-schedule` ejemplo, se elimina la programación de instantáneas especificada del clúster especificado.

```
aws redshift modify-cluster-snapshot-schedule \  
  --cluster-identifier mycluster \  
  --schedule-identifier mysnapshotschedule \  
  --disassociate-schedule
```

Este comando no produce ninguna salida.

Para obtener más información, consulte [Automated Snapshot Schedules](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [ModifyClusterSnapshotSchedule](#) la Referencia de AWS CLI comandos.

## **modify-cluster-snapshot**

En el siguiente ejemplo de código, se muestra cómo usar `modify-cluster-snapshot`.

### AWS CLI

Para modificar la instantánea del clúster

El siguiente `modify-cluster-snapshot` ejemplo establece la configuración del período de retención manual para la instantánea del clúster especificada en un valor de 10 días.

```
aws redshift modify-cluster-snapshot \  
  --cluster-identifier mycluster \  
  --snapshot-identifier mysnapshot \  
  --manual-retention-period-days 10
```

```
--snapshot-identifier mycluster-2019-11-06-16-32 \  
--manual-snapshot-retention-period 10
```

Salida:

```
{  
  "Snapshot": {  
    "SnapshotIdentifier": "mycluster-2019-11-06-16-32",  
    "ClusterIdentifier": "mycluster",  
    "SnapshotCreateTime": "2019-12-07T00:34:05.633Z",  
    "Status": "available",  
    "Port": 5439,  
    "AvailabilityZone": "us-west-2f",  
    "ClusterCreateTime": "2019-12-05T18:44:36.991Z",  
    "MasterUsername": "adminuser",  
    "ClusterVersion": "1.0",  
    "SnapshotType": "manual",  
    "NodeType": "dc2.large",  
    "NumberOfNodes": 2,  
    "DBName": "dev",  
    "VpcId": "vpc-b1cel7t9",  
    "Encrypted": false,  
    "EncryptedWithHSM": false,  
    "OwnerAccount": "123456789012",  
    "TotalBackupSizeInMegaBytes": 64384.0,  
    "ActualIncrementalBackupSizeInMegaBytes": 24.0,  
    "BackupProgressInMegaBytes": 24.0,  
    "CurrentBackupRateInMegaBytesPerSecond": 13.0011,  
    "EstimatedSecondsToCompletion": 0,  
    "ElapsedTimeInSeconds": 1,  
    "Tags": [  
      {  
        "Key": "mytagkey",  
        "Value": "mytagvalue"  
      }  
    ],  
    "EnhancedVpcRouting": false,  
    "MaintenanceTrackName": "current",  
    "ManualSnapshotRetentionPeriod": 10,  
    "ManualSnapshotRemainingDays": 6,  
    "SnapshotRetentionStartTime": "2019-12-07T00:34:07.479Z"  
  }  
}
```

Para obtener más información, consulte [Amazon Redshift Snapshots](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia de [ModifyClusterSnapshot](#) comandos AWS CLI .

## **modify-cluster-subnet-group**

En el siguiente ejemplo de código, se muestra cómo usar `modify-cluster-subnet-group`.

### AWS CLI

El GroupThis ejemplo de modificación de las subredes en una subred de clúster muestra cómo modificar la lista de subredes de un grupo de subredes de caché. De forma predeterminada, la salida está en formato.Comando: JSON

```
aws redshift modify-cluster-subnet-group --cluster-subnet-group-name mysubnetgroup
--subnet-ids subnet-763fdd1 subnet-ac830e9
```

### Resultado:

```
{
  "ClusterSubnetGroup":
  {
    "Subnets": [
      {
        "SubnetStatus": "Active",
        "SubnetIdentifier": "subnet-763fdd1c",
        "SubnetAvailabilityZone":
          { "Name": "us-east-1a" }
      },
      {
        "SubnetStatus": "Active",
        "SubnetIdentifier": "subnet-ac830e9",
        "SubnetAvailabilityZone":
          { "Name": "us-east-1b" }
      } ],
    "VpcId": "vpc-7e3fdd14",
    "SubnetGroupStatus": "Complete",
    "Description": "My subnet group",
    "ClusterSubnetGroupName": "mysubnetgroup"
  },
}
```

```
"ResponseMetadata": {  
  "RequestId": "8da93e89-8372-f936-93a8-873918938197a"  
}
```

- Para API obtener más información, consulte la Referencia [ModifyClusterSubnetGroup](#) de AWS CLI comandos.

## modify-cluster

En el siguiente ejemplo de código, se muestra cómo usar `modify-cluster`.

### AWS CLI

Asociar un grupo de seguridad a un ClusterThis ejemplo muestra cómo asociar un grupo de seguridad de clúster al clúster especificado. Comando:

```
aws redshift modify-cluster --cluster-identifier mycluster --cluster-security-groups  
mysecuritygroup
```

Modificar el período de mantenimiento por a ClusterThis muestra cómo cambiar el período de mantenimiento semanal preferido para un clúster para que sea el período mínimo de cuatro horas, que comienza los domingos a las 23:15 horas y termina los lunes a las 3:15 a.m.

Comando:

```
aws redshift modify-cluster --cluster-identifier mycluster --preferred-maintenance-  
window Sun:23:15-Mon:03:15
```

Cambiar la contraseña maestra del ClusterThis ejemplo muestra cómo cambiar la contraseña maestra de un cluster. Comando:

```
aws redshift modify-cluster --cluster-identifier mycluster --master-user-password  
A1b2c3d4
```

- Para API obtener más información, consulte la Referencia de [ModifyCluster](#) comandos AWS CLI .

## modify-event-subscription

En el siguiente ejemplo de código, se muestra cómo usar `modify-event-subscription`.

## AWS CLI

Para modificar la suscripción a un evento

En el siguiente `modify-event-subscription` ejemplo, se deshabilita la suscripción de notificaciones de eventos especificada.

```
aws redshift modify-event-subscription \  
  --subscription-name mysubscription \  
  --no-enabled
```

Salida:

```
{  
  "EventSubscription": {  
    "CustomerAwsId": "123456789012",  
    "CustSubscriptionId": "mysubscription",  
    "SnsTopicArn": "arn:aws:sns:us-west-2:123456789012:MySNSStopic",  
    "Status": "active",  
    "SubscriptionCreationTime": "2019-12-09T21:50:21.332Z",  
    "SourceIdsList": [],  
    "EventCategoriesList": [  
      "management"  
    ],  
    "Severity": "ERROR",  
    "Enabled": false,  
    "Tags": []  
  }  
}
```

Para obtener más información, consulte [Suscripción a las notificaciones de eventos de Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia de [ModifyEventSubscription](#) comandos AWS CLI .

## **modify-scheduled-action**

En el siguiente ejemplo de código, se muestra cómo usar `modify-scheduled-action`.

## AWS CLI

Para modificar una acción programada

En el siguiente `modify-scheduled-action` ejemplo, se agrega una descripción a la acción programada existente especificada.

```
aws redshift modify-scheduled-action \  
  --scheduled-action-name myscheduledaction \  
  --scheduled-action-description "My scheduled action"
```

Salida:

```
{  
  "ScheduledActionName": "myscheduledaction",  
  "TargetAction": {  
    "ResizeCluster": {  
      "ClusterIdentifier": "mycluster",  
      "NumberOfNodes": 2,  
      "Classic": false  
    }  
  },  
  "Schedule": "at(2019-12-25T00:00:00)",  
  "IamRole": "arn:aws:iam::123456789012:role/myRedshiftRole",  
  "ScheduledActionDescription": "My scheduled action",  
  "State": "ACTIVE",  
  "NextInvocations": [  
    "2019-12-25T00:00:00Z"  
  ]  
}
```

- Para API obtener más información, consulte [ModifyScheduledAction](#) la Referencia de AWS CLI comandos.

## `modify-snapshot-copy-retention-period`

En el siguiente ejemplo de código, se muestra cómo usar `modify-snapshot-copy-retention-period`.

## AWS CLI

Para modificar el período de retención de copias instantáneas

El siguiente `modify-snapshot-copy-retention-period` ejemplo modifica el número de días para conservar las instantáneas del clúster especificado en la AWS región de destino después de copiarlas de la región de origen AWS .

```
aws redshift modify-snapshot-copy-retention-period \  
  --cluster-identifier mycluster \  
  --retention-period 15
```

Salida:

```
{  
  "Cluster": {  
    "ClusterIdentifier": "mycluster",  
    "NodeType": "dc2.large",  
    "ClusterStatus": "available",  
    "ClusterAvailabilityStatus": "Available",  
    "MasterUsername": "adminuser",  
    "DBName": "dev",  
    "Endpoint": {  
      "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",  
      "Port": 5439  
    },  
    "ClusterCreateTime": "2019-12-05T18:44:36.991Z",  
    "AutomatedSnapshotRetentionPeriod": 3,  
    "ManualSnapshotRetentionPeriod": -1,  
    "ClusterSecurityGroups": [],  
    "VpcSecurityGroups": [  
      {  
        "VpcSecurityGroupId": "sh-a1a123ab",  
        "Status": "active"  
      }  
    ],  
    "ClusterParameterGroups": [  
      {  
        "ParameterGroupName": "default.redshift-1.0",  
        "ParameterApplyStatus": "in-sync"  
      }  
    ],  
    "ClusterSubnetGroupName": "default",  
    "VpcId": "vpc-b1fet7t9",  
    "AvailabilityZone": "us-west-2f",  
    "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",  
    "PendingModifiedValues": {
```



```

        "NodeType": "dc2.large",
        "NumberOfNodes": 2,
        "ClusterType": "multi-node"
    },
    "ClusterVersion": "1.0",
    "AllowVersionUpgrade": true,
    "NumberOfNodes": 4,
    "PubliclyAccessible": false,
    "Encrypted": false,
    "ClusterSnapshotCopyStatus": {
        "DestinationRegion": "us-west-1",
        "RetentionPeriod": 15,
        "ManualSnapshotRetentionPeriod": -1
    },
    "Tags": [
        {
            "Key": "mytags",
            "Value": "tag1"
        }
    ],
    "EnhancedVpcRouting": false,
    "IamRoles": [],
    "MaintenanceTrackName": "current",
    "DeferredMaintenanceWindows": [
        {
            "DeferMaintenanceIdentifier": "dfm-mUdVSfDcT1F4SGhw6fyF",
            "DeferMaintenanceStartTime": "2019-12-10T18:18:39.354Z",
            "DeferMaintenanceEndTime": "2020-01-09T18:18:39.354Z"
        }
    ],
    "NextMaintenanceWindowStartTime": "2020-01-11T16:00:00Z"
}
}

```

Para obtener más información, consulte [Snapshot Schedule Format](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [ModifySnapshotCopyRetentionPeriod](#) la Referencia de AWS CLI comandos.

## modify-snapshot-schedule

En el siguiente ejemplo de código, se muestra cómo usar `modify-snapshot-schedule`.

## AWS CLI

Para modificar la programación de instantáneas

En el siguiente `modify-snapshot-schedule` ejemplo, se modifica la frecuencia de la programación de instantáneas especificada para que pase a ser cada 10 horas.

```
aws redshift modify-snapshot-schedule \  
  --schedule-identifier mynapshotschedule \  
  --schedule-definitions "rate(10 hours)"
```

Salida:

```
{  
  "ScheduleDefinitions": [  
    "rate(10 hours)"  
  ],  
  "ScheduleIdentifier": "mynapshotschedule",  
  "ScheduleDescription": "My schedule description",  
  "Tags": []  
}
```

Para obtener más información, consulte [Snapshot Schedule Format](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [ModifySnapshotSchedule](#) la Referencia de AWS CLI comandos.

## **purchase-reserved-node-offering**

En el siguiente ejemplo de código, se muestra cómo usar `purchase-reserved-node-offering`.

### AWS CLI

El `NodeThis` ejemplo Comprar un nodo reservado muestra cómo comprar una oferta de nodo reservado. `reserved-node-offering-id` Se obtiene llamando a `describe-reserved-node-offerings`. Command:

```
aws redshift purchase-reserved-node-offering --reserved-node-offering-id ceb6a579-  
cf4c-4343-be8b-d832c45ab51c
```

Resultado:

```
{
  "ReservedNode": {
    "OfferingType": "Heavy Utilization",
    "FixedPrice": "",
    "NodeType": "dw.hs1.xlarge",
    "ReservedNodeId": "1ba8e2e3-bc01-4d65-b35d-a4a3e931547e",
    "UsagePrice": "",
    "RecurringCharges": [
      {
        "RecurringChargeAmount": "",
        "RecurringChargeFrequency": "Hourly"
      }
    ],
    "NodeCount": 1,
    "State": "payment-pending",
    "StartTime": "2013-02-13T17:08:39.051Z",
    "Duration": 31536000,
    "ReservedNodeOfferingId": "ceb6a579-cf4c-4343-be8b-d832c45ab51c"
  },
  "ResponseMetadata": {
    "RequestId": "01bda7bf-7600-11e2-b605-2568d7396e7f"
  }
}
```

- Para API obtener más información, consulte [PurchaseReservedNodeOffering](#) la Referencia de AWS CLI comandos.

## reboot-cluster

En el siguiente ejemplo de código, se muestra cómo usar `reboot-cluster`.

### AWS CLI

Reiniciar un ClusterThis ejemplo reinicia un clúster. De forma predeterminada, el resultado está en JSON formato. Comando:

```
aws redshift reboot-cluster --cluster-identifier mycluster
```

Resultado:

```
{
  "Cluster": {
```

```

    "NodeType": "dw.hs1.xlarge",
    "Endpoint": {
      "Port": 5439,
      "Address": "mycluster.coqoarplqhsn.us-east-1.redshift.amazonaws.com"
    },
    "ClusterVersion": "1.0",
    "PubliclyAccessible": "true",
    "MasterUsername": "adminuser",
    "ClusterParameterGroups": [
      {
        "ParameterApplyStatus": "in-sync",
        "ParameterGroupName": "default.redshift-1.0"
      }
    ],
    "ClusterSecurityGroups": [
      {
        "Status": "active",
        "ClusterSecurityGroupName": "default"
      }
    ],
    "AllowVersionUpgrade": true,
    "VpcSecurityGroups": [],
    "AvailabilityZone": "us-east-1a",
    "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
    "PreferredMaintenanceWindow": "sun:23:15-mon:03:15",
    "AutomatedSnapshotRetentionPeriod": 1,
    "ClusterStatus": "rebooting",
    "ClusterIdentifier": "mycluster",
    "DBName": "dev",
    "NumberOfNodes": 2,
    "PendingModifiedValues": {}
  },
  "ResponseMetadata": {
    "RequestId": "61c8b564-64e8-11e2-8f7d-3b939af52818"
  }
}

```

- Para API obtener más información, consulte la Referencia [RebootCluster](#) de AWS CLI comandos.

## reset-cluster-parameter-group

En el siguiente ejemplo de código, se muestra cómo usar `reset-cluster-parameter-group`.

## AWS CLI

El GroupThis ejemplo de restablecimiento de parámetros en un parámetro muestra cómo restablecer todos los parámetros de un grupo de parámetros. Comando:

```
aws redshift reset-cluster-parameter-group --parameter-group-name
myclusterparametergroup --reset-all-parameters
```

- Para API obtener más información, consulte la Referencia [ResetClusterParameterGroup](#) de AWS CLI comandos.

## resize-cluster

En el siguiente ejemplo de código, se muestra cómo usar `resize-cluster`.

### AWS CLI

Para cambiar el tamaño del clúster

En el siguiente `resize-cluster` ejemplo, se cambia el tamaño del clúster especificado.

```
aws redshift resize-cluster \  
  --cluster-identifier mycluster \  
  --cluster-type multi-node \  
  --node-type dc2.large \  
  --number-of-nodes 6 \  
  --classic
```

Salida:

```
{  
  "Cluster": {  
    "ClusterIdentifier": "mycluster",  
    "NodeType": "dc2.large",  
    "ClusterStatus": "resizing",  
    "ClusterAvailabilityStatus": "Modifying",  
    "MasterUsername": "adminuser",  
    "DBName": "dev",  
    "Endpoint": {  
      "Address": "mycluster.cmeaswquae.us-west-2.redshift.amazonaws.com",  
      "Port": 5439  
    }  
  },  
}
```

```
"ClusterCreateTime": "2019-12-05T18:44:36.991Z",
"AutomatedSnapshotRetentionPeriod": 3,
"ManualSnapshotRetentionPeriod": -1,
"ClusterSecurityGroups": [],
"VpcSecurityGroups": [
  {
    "VpcSecurityGroupId": "sh-a1a123ab",
    "Status": "active"
  }
],
"ClusterParameterGroups": [
  {
    "ParameterGroupName": "default.redshift-1.0",
    "ParameterApplyStatus": "in-sync"
  }
],
"ClusterSubnetGroupName": "default",
"VpcId": "vpc-a1abc1a1",
"AvailabilityZone": "us-west-2f",
"PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
"PendingModifiedValues": {
  "NodeType": "dc2.large",
  "NumberOfNodes": 6,
  "ClusterType": "multi-node"
},
"ClusterVersion": "1.0",
"AllowVersionUpgrade": true,
"NumberOfNodes": 4,
"PubliclyAccessible": false,
"Encrypted": false,
"ClusterSnapshotCopyStatus": {
  "DestinationRegion": "us-west-1",
  "RetentionPeriod": 15,
  "ManualSnapshotRetentionPeriod": -1
},
"Tags": [
  {
    "Key": "mytags",
    "Value": "tag1"
  }
],
"EnhancedVpcRouting": false,
"IamRoles": [],
"MaintenanceTrackName": "current",
```

```

    "DeferredMaintenanceWindows": [
      {
        "DeferMaintenanceIdentifier": "dfm-mUdVCfDcT1B4SGhw6fyF",
        "DeferMaintenanceStartTime": "2019-12-10T18:18:39.354Z",
        "DeferMaintenanceEndTime": "2020-01-09T18:18:39.354Z"
      }
    ],
    "NextMaintenanceWindowStartTime": "2020-01-11T16:00:00Z",
    "ResizeInfo": {
      "ResizeType": "ClassicResize",
      "AllowCancelResize": true
    }
  }
}

```

Para obtener más información, consulte Cambiar [el tamaño de un clúster](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [ResizeCluster](#) la Referencia de AWS CLI comandos.

## restore-from-cluster-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `restore-from-cluster-snapshot`.

### AWS CLI

Restaurar un clúster a partir de un SnapshotThis ejemplo restaura un clúster a partir de una instantánea. Comando:

```
aws redshift restore-from-cluster-snapshot --cluster-identifier mycluster-clone --
snapshot-identifier my-snapshot-id
```

### Resultado:

```

{
  "Cluster": {
    "NodeType": "dw.hs1.xlarge",
    "ClusterVersion": "1.0",
    "PubliclyAccessible": "true",
    "MasterUsername": "adminuser",
    "ClusterParameterGroups": [

```

```

    {
      "ParameterApplyStatus": "in-sync",
      "ParameterGroupName": "default.redshift-1.0"
    }
  ],
  "ClusterSecurityGroups": [
    {
      "Status": "active",
      "ClusterSecurityGroupName": "default"
    }
  ],
  "AllowVersionUpgrade": true,
  "VpcSecurityGroups": \[],
  "PreferredMaintenanceWindow": "sun:23:15-mon:03:15",
  "AutomatedSnapshotRetentionPeriod": 1,
  "ClusterStatus": "creating",
  "ClusterIdentifier": "mycluster-clone",
  "DBName": "dev",
  "NumberOfNodes": 2,
  "PendingModifiedValues": {}
},
"ResponseMetadata": {
  "RequestId": "77fd512b-64e3-11e2-8f5b-e90bd6c77476"
}
}

```

- Para API obtener más información, consulte la Referencia de [RestoreFromClusterSnapshot](#) comandos AWS CLI .

## restore-table-from-cluster-snapshot

En el siguiente ejemplo de código, se muestra cómo usar `restore-table-from-cluster-snapshot`.

### AWS CLI

Para restaurar una tabla a partir de una instantánea de un clúster

En el siguiente `restore-table-from-cluster-snapshot` ejemplo, se crea una tabla nueva a partir de la tabla especificada en la instantánea del clúster especificada.

```
aws redshift restore-table-from-cluster-snapshot /
```



```
--cluster-identifier mycluster /  
--snapshot-identifier mycluster-2019-11-19-16-17 /  
--source-database-name dev /  
--source-schema-name public /  
--source-table-name mytable /  
--target-database-name dev /  
--target-schema-name public /  
--new-table-name mytable-clone
```

Salida:

```
{  
  "TableRestoreStatus": {  
    "TableRestoreRequestId": "a123a12b-abc1-1a1a-a123-a1234ab12345",  
    "Status": "PENDING",  
    "RequestTime": "2019-12-20T00:20:16.402Z",  
    "ClusterIdentifier": "mycluster",  
    "SnapshotIdentifier": "mycluster-2019-11-19-16-17",  
    "SourceDatabaseName": "dev",  
    "SourceSchemaName": "public",  
    "SourceTableName": "mytable",  
    "TargetDatabaseName": "dev",  
    "TargetSchemaName": "public",  
    "NewTableName": "mytable-clone"  
  }  
}
```

Para obtener más información, consulte [Restauración de una tabla a partir de una instantánea](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte [RestoreTableFromClusterSnapshot](#) la Referencia de AWS CLI comandos.

## **revoke-cluster-security-group-ingress**

En el siguiente ejemplo de código, se muestra cómo usar `revoke-cluster-security-group-ingress`.

### AWS CLI

Revocar el acceso desde un GroupThis ejemplo de EC2 seguridad revoca el acceso a un grupo de EC2 seguridad de Amazon con nombre.Comando:

```
aws redshift revoke-cluster-security-group-ingress --cluster-security-group-name
mysecuritygroup --ec2-security-group-name myec2securitygroup --ec2-security-group-
owner-id 123445677890
```

Al revocar el acceso a un CIDR range Este ejemplo, se revoca el acceso a un rango. Comando:  
CIDR

```
aws redshift revoke-cluster-security-group-ingress --cluster-security-group-name
mysecuritygroup --cidrip 192.168.100.100/32
```

- Para obtener API más información, consulte la Referencia de comandos [RevokeClusterSecurityGroupIngress](#). AWS CLI

## revoke-snapshot-access

En el siguiente ejemplo de código, se muestra cómo usar `revoke-snapshot-access`.

### AWS CLI

Revocar la autorización de una AWS cuenta para restaurar un Snapshot Este ejemplo revoca la autorización de la AWS cuenta 444455556666 para restaurar la instantánea. `my-snapshot-id` De forma predeterminada, la salida está en JSON formato. Comando:

```
aws redshift revoke-snapshot-access --snapshot-id my-snapshot-id --account-with-
restore-access 444455556666
```

Resultado:

```
{
  "Snapshot": {
    "Status": "available",
    "SnapshotCreateTime": "2013-07-17T22:04:18.947Z",
    "EstimatedSecondsToCompletion": 0,
    "AvailabilityZone": "us-east-1a",
    "ClusterVersion": "1.0",
    "MasterUsername": "adminuser",
    "Encrypted": false,
    "OwnerAccount": "111122223333",
    "BackupProgressInMegabytes": 11.0,
    "ElapsedTimeInSeconds": 0,
    "DBName": "dev",
```

```

    "CurrentBackupRateInMegabytesPerSecond": 0.1534,
    "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
    "ActualIncrementalBackupSizeInMegabytes"; 11.0,
    "SnapshotType": "manual",
    "NodeType": "dw.hs1.xlarge",
    "ClusterIdentifier": "mycluster",
    "TotalBackupSizeInMegabytes": 20.0,
    "Port": 5439,
    "NumberOfNodes": 2,
    "SnapshotIdentifier": "my-snapshot-id"
  }
}

```

- Para API obtener más información, consulte la Referencia [RevokeSnapshotAccess](#) de AWS CLI comandos.

## rotate-encryption-key

En el siguiente ejemplo de código, se muestra cómo usar rotate-encryption-key.

### AWS CLI

Para rotar la clave de cifrado de un clúster

En el siguiente rotate-encryption-key ejemplo, se rota la clave de cifrado del clúster especificado.

```

aws redshift rotate-encryption-key \
  --cluster-identifier mycluster

```

Salida:

```

{
  "Cluster": {
    "ClusterIdentifier": "mycluster",
    "NodeType": "dc2.large",
    "ClusterStatus": "rotating-keys",
    "ClusterAvailabilityStatus": "Modifying",
    "MasterUsername": "adminuser",
    "DBName": "dev",
    "Endpoint": {
      "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",

```

```
    "Port": 5439
  },
  "ClusterCreateTime": "2019-12-10T19:25:45.886Z",
  "AutomatedSnapshotRetentionPeriod": 30,
  "ManualSnapshotRetentionPeriod": -1,
  "ClusterSecurityGroups": [],
  "VpcSecurityGroups": [
    {
      "VpcSecurityGroupId": "sh-a1a123ab",
      "Status": "active"
    }
  ],
  "ClusterParameterGroups": [
    {
      "ParameterGroupName": "default.redshift-1.0",
      "ParameterApplyStatus": "in-sync"
    }
  ],
  "ClusterSubnetGroupName": "default",
  "VpcId": "vpc-a1abc1a1",
  "AvailabilityZone": "us-west-2a",
  "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
  "PendingModifiedValues": {},
  "ClusterVersion": "1.0",
  "AllowVersionUpgrade": true,
  "NumberOfNodes": 2,
  "PubliclyAccessible": false,
  "Encrypted": true,
  "Tags": [],
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
bPxRfih3yCo8nvnvEXAMPLEKEY",
  "EnhancedVpcRouting": false,
  "IamRoles": [
    {
      "IamRoleArn": "arn:aws:iam::123456789012:role/myRedshiftRole",
      "ApplyStatus": "in-sync"
    }
  ],
  "MaintenanceTrackName": "current",
  "DeferredMaintenanceWindows": [],
  "NextMaintenanceWindowStartTime": "2019-12-14T16:00:00Z"
}
}
```

Para obtener más información, consulte el [cifrado de bases de datos de Amazon Redshift](#) en la Guía de administración de clústeres de Amazon Redshift.

- Para API obtener más información, consulte la Referencia [RotateEncryptionKey](#) de AWS CLI comandos.

## Ejemplos de Amazon Rekognition que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante Amazon AWS Command Line Interface Rekognition.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **compare-faces**

En el siguiente ejemplo de código, se muestra cómo usar `compare-faces`.

Para obtener información, consulte [Comparación de rostros en imágenes](#).

AWS CLI

Comparación de rostros en dos imágenes

El siguiente comando `compare-faces` compara rostros en dos imágenes almacenadas en un bucket de Amazon S3.

```
aws rekognition compare-faces \  
  --source-image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"source.jpg"}}' \  
  --target-image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"target.jpg"}}'
```

## Salida:

```
{
  "UnmatchedFaces": [],
  "FaceMatches": [
    {
      "Face": {
        "BoundingBox": {
          "Width": 0.12368916720151901,
          "Top": 0.16007372736930847,
          "Left": 0.5901257991790771,
          "Height": 0.25140416622161865
        },
        "Confidence": 100.0,
        "Pose": {
          "Yaw": -3.7351467609405518,
          "Roll": -0.10309021919965744,
          "Pitch": 0.8637830018997192
        },
        "Quality": {
          "Sharpness": 95.51618957519531,
          "Brightness": 65.29893493652344
        },
        "Landmarks": [
          {
            "Y": 0.26721030473709106,
            "X": 0.6204193830490112,
            "Type": "eyeLeft"
          },
          {
            "Y": 0.26831310987472534,
            "X": 0.6776827573776245,
            "Type": "eyeRight"
          },
          {
            "Y": 0.3514654338359833,
            "X": 0.6241428852081299,
            "Type": "mouthLeft"
          },
          {
            "Y": 0.35258132219314575,
            "X": 0.6713621020317078,
            "Type": "mouthRight"
          }
        ]
      }
    }
  ]
}
```

```

        {
            "Y": 0.3140771687030792,
            "X": 0.6428444981575012,
            "Type": "nose"
        }
    ],
    "Similarity": 100.0
}
],
"SourceImageFace": {
    "BoundingBox": {
        "Width": 0.12368916720151901,
        "Top": 0.16007372736930847,
        "Left": 0.5901257991790771,
        "Height": 0.25140416622161865
    },
    "Confidence": 100.0
}
}

```

Para obtener más información, consulte [Comparación de rostros en imágenes](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [CompareFaces](#) la Referencia de AWS CLI comandos.

## create-collection

En el siguiente ejemplo de código, se muestra cómo usar `create-collection`.

Para obtener información, consulte [Creación de una colección](#).

### AWS CLI

#### Creación de una colección

El siguiente comando `create-collection` crea una colección con el nombre especificado.

```
aws rekognition create-collection \
  --collection-id "MyCollection"
```

Salida:

```
{
  "CollectionArn": "aws:rekognition:us-west-2:123456789012:collection/
MyCollection",
  "FaceModelVersion": "4.0",
  "StatusCode": 200
}
```

Para obtener más información, consulte [Creación de una colección](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [CreateCollection](#) la Referencia de AWS CLI comandos.

## create-stream-processor

En el siguiente ejemplo de código, se muestra cómo usar `create-stream-processor`.

### AWS CLI

Para crear un nuevo procesador de flujo

El siguiente `create-stream-processor` ejemplo crea un nuevo procesador de transmisión con la configuración especificada.

```
aws rekognition create-stream-processor --name my-stream-processor\
--input '{"KinesisVideoStream":{"Arn":"arn:aws:kinesisvideo:us-
west-2:123456789012:stream/macwebcam/1530559711205"}}'\
--stream-processor-output '{"KinesisDataStream":{"Arn":"arn:aws:kinesis:us-
west-2:123456789012:stream/AmazonRekognitionRekStream"}}'\
--role-arn arn:aws:iam::123456789012:role/AmazonRekognitionDetect\
--settings '{"FaceSearch":
{"CollectionId":"MyCollection", "FaceMatchThreshold":85.5}}'
```

Salida:

```
{
  "StreamProcessorArn": "arn:aws:rekognition:us-
west-2:123456789012:streamprocessor/my-stream-processor"
}
```



Para obtener más información, consulte [Trabajar con vídeos en streaming](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [CreateStreamProcessor](#) comandos AWS CLI .

## **delete-collection**

En el siguiente ejemplo de código, se muestra cómo usar `delete-collection`.

Para obtener información, consulte [Eliminación de una colección](#).

### AWS CLI

Eliminación de una colección

El siguiente comando `delete-collection` elimina la colección especificada.

```
aws rekognition delete-collection \  
  --collection-id MyCollection
```

Salida:

```
{  
  "StatusCode": 200  
}
```

Para obtener más información, consulte [Eliminación de una colección](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [DeleteCollection](#) la Referencia de AWS CLI comandos.

## **delete-faces**

En el siguiente ejemplo de código, se muestra cómo usar `delete-faces`.

Para obtener información, consulte [Eliminación de rostros de una colección](#).

### AWS CLI

Eliminación de los rostros de una colección

El siguiente comando `delete-faces` elimina los rostros especificados de una colección.

```
aws rekognition delete-faces \  
  --collection-id MyCollection \  
  --face-ids '["0040279c-0178-436e-b70a-e61b074e96b0"]'
```

Salida:

```
{  
  "DeletedFaces": [  
    "0040279c-0178-436e-b70a-e61b074e96b0"  
  ]  
}
```

Para obtener más información, consulte [Eliminación de rostros de una colección](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [DeleteFaces](#) la Referencia de AWS CLI comandos.

## **delete-stream-processor**

En el siguiente ejemplo de código, se muestra cómo usar `delete-stream-processor`.

AWS CLI

Para eliminar un procesador de flujo

El siguiente `delete-stream-processor` comando elimina el procesador de flujo especificado.

```
aws rekognition delete-stream-processor \  
  --name my-stream-processor
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con vídeos en streaming](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [DeleteStreamProcessor](#) comandos AWS CLI .

## describe-collection

En el siguiente ejemplo de código, se muestra cómo usar `describe-collection`.

Para obtener información, consulte [Descripción de una colección](#).

### AWS CLI

Descripción de una colección

En el siguiente ejemplo de `describe-collection` se muestran los detalles de la colección especificada.

```
aws rekognition describe-collection \  
  --collection-id MyCollection
```

Salida:

```
{  
  "FaceCount": 200,  
  "CreationTimestamp": 1569444828.274,  
  "CollectionARN": "arn:aws:rekognition:us-west-2:123456789012:collection/  
MyCollection",  
  "FaceModelVersion": "4.0"  
}
```

Para obtener más información, consulte [Descripción de una colección](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [DescribeCollection](#) la Referencia de AWS CLI comandos.

## describe-stream-processor

En el siguiente ejemplo de código, se muestra cómo usar `describe-stream-processor`.

### AWS CLI

Para obtener información sobre un procesador de flujo

El siguiente `describe-stream-processor` comando muestra detalles sobre el procesador de flujo especificado.

```
aws rekognition describe-stream-processor \  
--name my-stream-processor
```

Salida:

```
{  
  "Status": "STOPPED",  
  "Name": "my-stream-processor",  
  "LastUpdateTimestamp": 1532449292.712,  
  "Settings": {  
    "FaceSearch": {  
      "FaceMatchThreshold": 80.0,  
      "CollectionId": "my-collection"  
    }  
  },  
  "RoleArn": "arn:aws:iam::123456789012:role/AmazonRekognitionDetectStream",  
  "StreamProcessorArn": "arn:aws:rekognition:us-west-2:123456789012:streamprocessor/my-stream-processpr",  
  "Output": {  
    "KinesisDataStream": {  
      "Arn": "arn:aws:kinesis:us-west-2:123456789012:stream/  
AmazonRekognitionRekStream"  
    }  
  },  
  "Input": {  
    "KinesisVideoStream": {  
      "Arn": "arn:aws:kinesisvideo:us-west-2:123456789012:stream/  
macwebcam/123456789012"  
    }  
  },  
  "CreationTimestamp": 1532449292.712  
}
```

Para obtener más información, consulte [Trabajar con vídeos en streaming](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [DescribeStreamProcessor](#) comandos AWS CLI .

## detect-faces

En el siguiente ejemplo de código, se muestra cómo usar detect-faces.

Para obtener información, consulte [Detección de rostros en una imagen](#).

## AWS CLI

### Detección de rostros en una imagen

El siguiente comando `detect-faces` detecta rostros en la imagen especificada almacenada en un bucket de Amazon S3.

```
aws rekognition detect-faces \  
  --image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"MyFriend.jpg"}}' \  
  --attributes "ALL"
```

Salida:

```
{  
  "FaceDetails": [  
    {  
      "Confidence": 100.0,  
      "Eyeglasses": {  
        "Confidence": 98.91107940673828,  
        "Value": false  
      },  
      "Sunglasses": {  
        "Confidence": 99.7966537475586,  
        "Value": false  
      },  
      "Gender": {  
        "Confidence": 99.56611633300781,  
        "Value": "Male"  
      },  
      "Landmarks": [  
        {  
          "Y": 0.26721030473709106,  
          "X": 0.6204193830490112,  
          "Type": "eyeLeft"  
        },  
        {  
          "Y": 0.26831310987472534,  
          "X": 0.6776827573776245,  
          "Type": "eyeRight"  
        }  
      ]  
    }  
  ]  
}
```

```
    "Y": 0.3514654338359833,  
    "X": 0.6241428852081299,  
    "Type": "mouthLeft"  
  },  
  {  
    "Y": 0.35258132219314575,  
    "X": 0.6713621020317078,  
    "Type": "mouthRight"  
  },  
  {  
    "Y": 0.3140771687030792,  
    "X": 0.6428444981575012,  
    "Type": "nose"  
  },  
  {  
    "Y": 0.24662546813488007,  
    "X": 0.6001564860343933,  
    "Type": "leftEyeBrowLeft"  
  },  
  {  
    "Y": 0.24326619505882263,  
    "X": 0.6303644776344299,  
    "Type": "leftEyeBrowRight"  
  },  
  {  
    "Y": 0.23818562924861908,  
    "X": 0.6146903038024902,  
    "Type": "leftEyeBrowUp"  
  },  
  {  
    "Y": 0.24373626708984375,  
    "X": 0.6640064716339111,  
    "Type": "rightEyeBrowLeft"  
  },  
  {  
    "Y": 0.24877218902111053,  
    "X": 0.7025929093360901,  
    "Type": "rightEyeBrowRight"  
  },  
  {  
    "Y": 0.23938551545143127,  
    "X": 0.6823262572288513,  
    "Type": "rightEyeBrowUp"  
  },  
  },
```

```
{
  "Y": 0.265746533870697,
  "X": 0.6112898588180542,
  "Type": "leftEyeLeft"
},
{
  "Y": 0.2676128149032593,
  "X": 0.6317071914672852,
  "Type": "leftEyeRight"
},
{
  "Y": 0.262735515832901,
  "X": 0.6201658248901367,
  "Type": "leftEyeUp"
},
{
  "Y": 0.27025148272514343,
  "X": 0.6206279993057251,
  "Type": "leftEyeDown"
},
{
  "Y": 0.268223375082016,
  "X": 0.6658390760421753,
  "Type": "rightEyeLeft"
},
{
  "Y": 0.2672517001628876,
  "X": 0.687832236289978,
  "Type": "rightEyeRight"
},
{
  "Y": 0.26383838057518005,
  "X": 0.6769183874130249,
  "Type": "rightEyeUp"
},
{
  "Y": 0.27138751745224,
  "X": 0.676596462726593,
  "Type": "rightEyeDown"
},
{
  "Y": 0.32283174991607666,
  "X": 0.6350004076957703,
  "Type": "noseLeft"
}
```

```
    },
    {
      "Y": 0.3219289481639862,
      "X": 0.6567046642303467,
      "Type": "noseRight"
    },
    {
      "Y": 0.3420318365097046,
      "X": 0.6450609564781189,
      "Type": "mouthUp"
    },
    {
      "Y": 0.3664324879646301,
      "X": 0.6455618143081665,
      "Type": "mouthDown"
    },
    {
      "Y": 0.26721030473709106,
      "X": 0.6204193830490112,
      "Type": "leftPupil"
    },
    {
      "Y": 0.26831310987472534,
      "X": 0.6776827573776245,
      "Type": "rightPupil"
    },
    {
      "Y": 0.26343393325805664,
      "X": 0.5946047306060791,
      "Type": "upperJawlineLeft"
    },
    {
      "Y": 0.3543180525302887,
      "X": 0.6044883728027344,
      "Type": "midJawlineLeft"
    },
    {
      "Y": 0.4084877669811249,
      "X": 0.6477024555206299,
      "Type": "chinBottom"
    },
    {
      "Y": 0.3562754988670349,
      "X": 0.707981526851654,
```



```
        "Type": "midJawlineRight"
      },
      {
        "Y": 0.26580461859703064,
        "X": 0.7234612107276917,
        "Type": "upperJawlineRight"
      }
    ],
    "Pose": {
      "Yaw": -3.7351467609405518,
      "Roll": -0.10309021919965744,
      "Pitch": 0.8637830018997192
    },
    "Emotions": [
      {
        "Confidence": 8.74203109741211,
        "Type": "SURPRISED"
      },
      {
        "Confidence": 2.501944065093994,
        "Type": "ANGRY"
      },
      {
        "Confidence": 0.7378743290901184,
        "Type": "DISGUSTED"
      },
      {
        "Confidence": 3.5296201705932617,
        "Type": "HAPPY"
      },
      {
        "Confidence": 1.7162904739379883,
        "Type": "SAD"
      },
      {
        "Confidence": 9.518536567687988,
        "Type": "CONFUSED"
      },
      {
        "Confidence": 0.45474427938461304,
        "Type": "FEAR"
      },
      {
        "Confidence": 72.79895782470703,
```

```
        "Type": "CALM"
      }
    ],
    "AgeRange": {
      "High": 48,
      "Low": 32
    },
    "EyesOpen": {
      "Confidence": 98.93987274169922,
      "Value": true
    },
    "BoundingBox": {
      "Width": 0.12368916720151901,
      "Top": 0.16007372736930847,
      "Left": 0.5901257991790771,
      "Height": 0.25140416622161865
    },
    "Smile": {
      "Confidence": 93.4493179321289,
      "Value": false
    },
    "MouthOpen": {
      "Confidence": 90.53053283691406,
      "Value": false
    },
    "Quality": {
      "Sharpness": 95.51618957519531,
      "Brightness": 65.29893493652344
    },
    "Mustache": {
      "Confidence": 89.85221099853516,
      "Value": false
    },
    "Beard": {
      "Confidence": 86.1991195678711,
      "Value": true
    }
  }
]
}
```

Para obtener más información, consulte [Detección de rostros en una imagen](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [DetectFaces](#) la Referencia de AWS CLI comandos.

## detect-labels

En el siguiente ejemplo de código, se muestra cómo usar `detect-labels`.

Para obtener información, consulte [Detección de etiquetas en una imagen](#).

### AWS CLI

Detección de una etiqueta en una imagen

En el siguiente ejemplo de `detect-labels` se detectan escenas y objetos en una imagen almacenada en un bucket de Amazon S3.

```
aws rekognition detect-labels \  
  --image '{"S3Object":{"Bucket":"bucket","Name":"image"}}'
```

Salida:

```
{  
  "Labels": [  
    {  
      "Instances": [],  
      "Confidence": 99.15271759033203,  
      "Parents": [  
        {  
          "Name": "Vehicle"  
        },  
        {  
          "Name": "Transportation"  
        }  
      ],  
      "Name": "Automobile"  
    },  
    {  
      "Instances": [],  
      "Confidence": 99.15271759033203,  
      "Parents": [  
        {  
          "Name": "Transportation"  
        }  
      ],  
    }  
  ],  
}
```

```
    "Name": "Vehicle"
  },
  {
    "Instances": [],
    "Confidence": 99.15271759033203,
    "Parents": [],
    "Name": "Transportation"
  },
  {
    "Instances": [
      {
        "BoundingBox": {
          "Width": 0.10616336017847061,
          "Top": 0.5039216876029968,
          "Left": 0.0037978808395564556,
          "Height": 0.18528179824352264
        },
        "Confidence": 99.15271759033203
      },
      {
        "BoundingBox": {
          "Width": 0.2429988533258438,
          "Top": 0.5251884460449219,
          "Left": 0.7309805154800415,
          "Height": 0.21577216684818268
        },
        "Confidence": 99.1286392211914
      },
      {
        "BoundingBox": {
          "Width": 0.14233611524105072,
          "Top": 0.5333095788955688,
          "Left": 0.6494812965393066,
          "Height": 0.15528248250484467
        },
        "Confidence": 98.48368072509766
      },
      {
        "BoundingBox": {
          "Width": 0.11086395382881165,
          "Top": 0.5354844927787781,
          "Left": 0.10355594009160995,
          "Height": 0.10271988064050674
        },

```

```
    "Confidence": 96.45606231689453
  },
  {
    "BoundingBox": {
      "Width": 0.06254628300666809,
      "Top": 0.5573825240135193,
      "Left": 0.46083059906959534,
      "Height": 0.053911514580249786
    },
    "Confidence": 93.65448760986328
  },
  {
    "BoundingBox": {
      "Width": 0.10105438530445099,
      "Top": 0.534368634223938,
      "Left": 0.5743985772132874,
      "Height": 0.12226245552301407
    },
    "Confidence": 93.06217193603516
  },
  {
    "BoundingBox": {
      "Width": 0.056389667093753815,
      "Top": 0.5235804319381714,
      "Left": 0.9427769780158997,
      "Height": 0.17163699865341187
    },
    "Confidence": 92.6864013671875
  },
  {
    "BoundingBox": {
      "Width": 0.06003860384225845,
      "Top": 0.5441341400146484,
      "Left": 0.22409997880458832,
      "Height": 0.06737709045410156
    },
    "Confidence": 90.4227066040039
  },
  {
    "BoundingBox": {
      "Width": 0.02848697081208229,
      "Top": 0.5107086896896362,
      "Left": 0,
      "Height": 0.19150497019290924
    }
  }
}
```

```
    },
    "Confidence": 86.65286254882812
  },
  {
    "BoundingBox": {
      "Width": 0.04067881405353546,
      "Top": 0.5566273927688599,
      "Left": 0.316415935754776,
      "Height": 0.03428703173995018
    },
    "Confidence": 85.36471557617188
  },
  {
    "BoundingBox": {
      "Width": 0.043411049991846085,
      "Top": 0.5394920110702515,
      "Left": 0.18293385207653046,
      "Height": 0.0893595889210701
    },
    "Confidence": 82.21705627441406
  },
  {
    "BoundingBox": {
      "Width": 0.031183116137981415,
      "Top": 0.5579366683959961,
      "Left": 0.2853088080883026,
      "Height": 0.03989990055561066
    },
    "Confidence": 81.0157470703125
  },
  {
    "BoundingBox": {
      "Width": 0.031113790348172188,
      "Top": 0.5504819750785828,
      "Left": 0.2580395042896271,
      "Height": 0.056484755128622055
    },
    "Confidence": 56.13441467285156
  },
  {
    "BoundingBox": {
      "Width": 0.08586374670267105,
      "Top": 0.5438792705535889,
      "Left": 0.5128012895584106,
```

```
        "Height": 0.08550430089235306
      },
      "Confidence": 52.37760925292969
    }
  ],
  "Confidence": 99.15271759033203,
  "Parents": [
    {
      "Name": "Vehicle"
    },
    {
      "Name": "Transportation"
    }
  ],
  "Name": "Car"
},
{
  "Instances": [],
  "Confidence": 98.9914321899414,
  "Parents": [],
  "Name": "Human"
},
{
  "Instances": [
    {
      "BoundingBox": {
        "Width": 0.19360728561878204,
        "Top": 0.35072067379951477,
        "Left": 0.43734854459762573,
        "Height": 0.2742200493812561
      },
      "Confidence": 98.9914321899414
    },
    {
      "BoundingBox": {
        "Width": 0.03801717236638069,
        "Top": 0.5010883808135986,
        "Left": 0.9155802130699158,
        "Height": 0.06597328186035156
      },
      "Confidence": 85.02790832519531
    }
  ],
  "Confidence": 98.9914321899414,
```

```
    "Parents": [],
    "Name": "Person"
  },
  {
    "Instances": [],
    "Confidence": 93.24951934814453,
    "Parents": [],
    "Name": "Machine"
  },
  {
    "Instances": [
      {
        "BoundingBox": {
          "Width": 0.03561960905790329,
          "Top": 0.6468243598937988,
          "Left": 0.7850857377052307,
          "Height": 0.08878646790981293
        },
        "Confidence": 93.24951934814453
      },
      {
        "BoundingBox": {
          "Width": 0.02217046171426773,
          "Top": 0.6149078607559204,
          "Left": 0.04757237061858177,
          "Height": 0.07136218994855881
        },
        "Confidence": 91.5025863647461
      },
      {
        "BoundingBox": {
          "Width": 0.016197510063648224,
          "Top": 0.6274210214614868,
          "Left": 0.6472989320755005,
          "Height": 0.04955997318029404
        },
        "Confidence": 85.14686584472656
      },
      {
        "BoundingBox": {
          "Width": 0.020207518711686134,
          "Top": 0.6348286867141724,
          "Left": 0.7295016646385193,
          "Height": 0.07059963047504425
        }
      }
    ]
  }
}
```



```
    },
    "Confidence": 83.34547424316406
  },
  {
    "BoundingBox": {
      "Width": 0.020280985161662102,
      "Top": 0.6171894669532776,
      "Left": 0.08744934946298599,
      "Height": 0.05297485366463661
    },
    "Confidence": 79.9981460571289
  },
  {
    "BoundingBox": {
      "Width": 0.018318990245461464,
      "Top": 0.623889148235321,
      "Left": 0.6836880445480347,
      "Height": 0.06730121374130249
    },
    "Confidence": 78.87144470214844
  },
  {
    "BoundingBox": {
      "Width": 0.021310249343514442,
      "Top": 0.6167286038398743,
      "Left": 0.004064912907779217,
      "Height": 0.08317798376083374
    },
    "Confidence": 75.89361572265625
  },
  {
    "BoundingBox": {
      "Width": 0.03604431077837944,
      "Top": 0.7030032277107239,
      "Left": 0.9254803657531738,
      "Height": 0.04569442570209503
    },
    "Confidence": 64.402587890625
  },
  {
    "BoundingBox": {
      "Width": 0.009834849275648594,
      "Top": 0.5821820497512817,
      "Left": 0.28094568848609924,
```

```
        "Height": 0.01964157074689865
      },
      "Confidence": 62.79907989501953
    },
    {
      "BoundingBox": {
        "Width": 0.01475677452981472,
        "Top": 0.6137543320655823,
        "Left": 0.5950819253921509,
        "Height": 0.039063986390829086
      },
      "Confidence": 59.40483474731445
    }
  ],
  "Confidence": 93.24951934814453,
  "Parents": [
    {
      "Name": "Machine"
    }
  ],
  "Name": "Wheel"
},
{
  "Instances": [],
  "Confidence": 92.61514282226562,
  "Parents": [],
  "Name": "Road"
},
{
  "Instances": [],
  "Confidence": 92.37877655029297,
  "Parents": [
    {
      "Name": "Person"
    }
  ],
  "Name": "Sport"
},
{
  "Instances": [],
  "Confidence": 92.37877655029297,
  "Parents": [
    {
      "Name": "Person"
    }
  ]
}
```

```
    }
  ],
  "Name": "Sports"
},
{
  "Instances": [
    {
      "BoundingBox": {
        "Width": 0.12326609343290329,
        "Top": 0.6332163214683533,
        "Left": 0.44815489649772644,
        "Height": 0.058117982000112534
      },
      "Confidence": 92.37877655029297
    }
  ],
  "Confidence": 92.37877655029297,
  "Parents": [
    {
      "Name": "Person"
    },
    {
      "Name": "Sport"
    }
  ],
  "Name": "Skateboard"
},
{
  "Instances": [],
  "Confidence": 90.62931060791016,
  "Parents": [
    {
      "Name": "Person"
    }
  ],
  "Name": "Pedestrian"
},
{
  "Instances": [],
  "Confidence": 88.81334686279297,
  "Parents": [],
  "Name": "Asphalt"
},
{
```

```
    "Instances": [],
    "Confidence": 88.81334686279297,
    "Parents": [],
    "Name": "Tarmac"
  },
  {
    "Instances": [],
    "Confidence": 88.23201751708984,
    "Parents": [],
    "Name": "Path"
  },
  {
    "Instances": [],
    "Confidence": 80.26520538330078,
    "Parents": [],
    "Name": "Urban"
  },
  {
    "Instances": [],
    "Confidence": 80.26520538330078,
    "Parents": [
      {
        "Name": "Building"
      },
      {
        "Name": "Urban"
      }
    ],
    "Name": "Town"
  },
  {
    "Instances": [],
    "Confidence": 80.26520538330078,
    "Parents": [],
    "Name": "Building"
  },
  {
    "Instances": [],
    "Confidence": 80.26520538330078,
    "Parents": [
      {
        "Name": "Building"
      },
      {

```

```
        "Name": "Urban"
      }
    ],
    "Name": "City"
  },
  {
    "Instances": [],
    "Confidence": 78.37934875488281,
    "Parents": [
      {
        "Name": "Car"
      },
      {
        "Name": "Vehicle"
      },
      {
        "Name": "Transportation"
      }
    ],
    "Name": "Parking Lot"
  },
  {
    "Instances": [],
    "Confidence": 78.37934875488281,
    "Parents": [
      {
        "Name": "Car"
      },
      {
        "Name": "Vehicle"
      },
      {
        "Name": "Transportation"
      }
    ],
    "Name": "Parking"
  },
  {
    "Instances": [],
    "Confidence": 74.37590026855469,
    "Parents": [
      {
        "Name": "Building"
      },

```

```
    {
      "Name": "Urban"
    },
    {
      "Name": "City"
    }
  ],
  "Name": "Downtown"
},
{
  "Instances": [],
  "Confidence": 69.84622955322266,
  "Parents": [
    {
      "Name": "Road"
    }
  ],
  "Name": "Intersection"
},
{
  "Instances": [],
  "Confidence": 57.68518829345703,
  "Parents": [
    {
      "Name": "Sports Car"
    },
    {
      "Name": "Car"
    },
    {
      "Name": "Vehicle"
    },
    {
      "Name": "Transportation"
    }
  ],
  "Name": "Coupe"
},
{
  "Instances": [],
  "Confidence": 57.68518829345703,
  "Parents": [
    {
      "Name": "Car"
    }
  ]
}
```

```
    },
    {
      "Name": "Vehicle"
    },
    {
      "Name": "Transportation"
    }
  ],
  "Name": "Sports Car"
},
{
  "Instances": [],
  "Confidence": 56.59492111206055,
  "Parents": [
    {
      "Name": "Path"
    }
  ],
  "Name": "Sidewalk"
},
{
  "Instances": [],
  "Confidence": 56.59492111206055,
  "Parents": [
    {
      "Name": "Path"
    }
  ],
  "Name": "Pavement"
},
{
  "Instances": [],
  "Confidence": 55.58770751953125,
  "Parents": [
    {
      "Name": "Building"
    },
    {
      "Name": "Urban"
    }
  ],
  "Name": "Neighborhood"
}
],
```

```
"LabelModelVersion": "2.0"  
}
```

Para obtener más información, consulte [Detección de etiquetas en una imagen](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [DetectLabels](#) la Referencia de AWS CLI comandos.

## detect-moderation-labels

En el siguiente ejemplo de código, se muestra cómo usar `detect-moderation-labels`.

Para obtener información, consulte [Detección de imágenes inapropiadas](#).

### AWS CLI

Detección de contenido no seguro en una imagen

El siguiente comando `detect-moderation-labels` detecta contenido no seguro en la imagen especificada almacenada en un bucket de Amazon S3.

```
aws rekognition detect-moderation-labels \  
  --image "S3Object={Bucket=MyImageS3Bucket,Name=gun.jpg}"
```

Salida:

```
{  
  "ModerationModelVersion": "3.0",  
  "ModerationLabels": [  
    {  
      "Confidence": 97.29618072509766,  
      "ParentName": "Violence",  
      "Name": "Weapon Violence"  
    },  
    {  
      "Confidence": 97.29618072509766,  
      "ParentName": "",  
      "Name": "Violence"  
    }  
  ]  
}
```



Para obtener más información, consulte [Detección de imágenes no seguras](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [DetectModerationLabels](#) la Referencia de AWS CLI comandos.

## detect-text

En el siguiente ejemplo de código, se muestra cómo usar detect-text.

Para obtener información, consulte [Detección de texto en una imagen](#).

### AWS CLI

Detección de texto en una imagen

El siguiente comando detect-text detecta el texto de la imagen especificada.

```
aws rekognition detect-text \  
  --image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"ExamplePicture.jpg"}}'
```

Salida:

```
{  
  "TextDetections": [  
    {  
      "Geometry": {  
        "BoundingBox": {  
          "Width": 0.24624845385551453,  
          "Top": 0.28288066387176514,  
          "Left": 0.391388863325119,  
          "Height": 0.022687450051307678  
        },  
        "Polygon": [  
          {  
            "Y": 0.28288066387176514,  
            "X": 0.391388863325119  
          },  
          {  
            "Y": 0.2826388478279114,  
            "X": 0.6376373171806335  
          },  
          {  
            "Y": 0.2826388478279114,  
            "X": 0.391388863325119  
          },  
          {  
            "Y": 0.28288066387176514,  
            "X": 0.391388863325119  
          }  
        ]  
      }  
    }  
  ]  
}
```

```
        "Y": 0.30532628297805786,  
        "X": 0.637677013874054  
    },  
    {  
        "Y": 0.305568128824234,  
        "X": 0.39142853021621704  
    }  
    ]  
},  
"Confidence": 94.35709381103516,  
"DetectedText": "ESTD 1882",  
"Type": "LINE",  
"Id": 0  
},  
{  
    "Geometry": {  
        "BoundingBox": {  
            "Width": 0.33933889865875244,  
            "Top": 0.32603850960731506,  
            "Left": 0.34534579515457153,  
            "Height": 0.07126858830451965  
        },  
        "Polygon": [  
            {  
                "Y": 0.32603850960731506,  
                "X": 0.34534579515457153  
            },  
            {  
                "Y": 0.32633158564567566,  
                "X": 0.684684693813324  
            },  
            {  
                "Y": 0.3976001739501953,  
                "X": 0.684575080871582  
            },  
            {  
                "Y": 0.3973070979118347,  
                "X": 0.345236212015152  
            }  
        ]  
    },  
    "Confidence": 99.95779418945312,  
    "DetectedText": "BRAINS",  
    "Type": "LINE",
```

```
    "Id": 1
  },
  {
    "Confidence": 97.22098541259766,
    "Geometry": {
      "BoundingBox": {
        "Width": 0.061079490929841995,
        "Top": 0.2843210697174072,
        "Left": 0.391391396522522,
        "Height": 0.021029088646173477
      },
      "Polygon": [
        {
          "Y": 0.2843210697174072,
          "X": 0.391391396522522
        },
        {
          "Y": 0.2828207015991211,
          "X": 0.4524524509906769
        },
        {
          "Y": 0.3038259446620941,
          "X": 0.4534534513950348
        },
        {
          "Y": 0.30532634258270264,
          "X": 0.3923923969268799
        }
      ]
    },
    "DetectedText": "ESTD",
    "ParentId": 0,
    "Type": "WORD",
    "Id": 2
  },
  {
    "Confidence": 91.49320983886719,
    "Geometry": {
      "BoundingBox": {
        "Width": 0.07007007300853729,
        "Top": 0.2828207015991211,
        "Left": 0.5675675868988037,
        "Height": 0.02250562608242035
      },
```

```

        "Polygon": [
            {
                "Y": 0.2828207015991211,
                "X": 0.5675675868988037
            },
            {
                "Y": 0.2828207015991211,
                "X": 0.6376376152038574
            },
            {
                "Y": 0.30532634258270264,
                "X": 0.6376376152038574
            },
            {
                "Y": 0.30532634258270264,
                "X": 0.5675675868988037
            }
        ]
    },
    "DetectedText": "1882",
    "ParentId": 0,
    "Type": "WORD",
    "Id": 3
},
{
    "Confidence": 99.95779418945312,
    "Geometry": {
        "BoundingBox": {
            "Width": 0.33933934569358826,
            "Top": 0.32633158564567566,
            "Left": 0.3453453481197357,
            "Height": 0.07127484679222107
        },
        "Polygon": [
            {
                "Y": 0.32633158564567566,
                "X": 0.3453453481197357
            },
            {
                "Y": 0.32633158564567566,
                "X": 0.684684693813324
            },
            {
                "Y": 0.39759939908981323,

```

```

        "X": 0.6836836934089661
      },
      {
        "Y": 0.39684921503067017,
        "X": 0.3453453481197357
      }
    ]
  },
  "DetectedText": "BRAINS",
  "ParentId": 1,
  "Type": "WORD",
  "Id": 4
}
]
}

```

- Para API obtener más información, consulte [DetectText](#) la Referencia de AWS CLI comandos.

## disassociate-faces

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-faces`.

### AWS CLI

```
aws rekognition disassociate-faces --face-ids list-of-face-ids
--user-id user-id --collection-id collection-name --region region-name
```

- Para API obtener más información, consulte [DisassociateFaces](#) la Referencia de AWS CLI comandos.

## get-celebrity-info

En el siguiente ejemplo de código, se muestra cómo usar `get-celebrity-info`.

### AWS CLI

Obtención de información sobre un famoso

El siguiente comando `get-celebrity-info` muestra información sobre el famoso especificado. El parámetro `id` procede de una llamada anterior a `recognize-celebrities`.

```
aws rekognition get-celebrity-info --id nnnnnnn
```

Salida:

```
{
  "Name": "Celeb A",
  "Urls": [
    "www.imdb.com/name/aaaaaaaa"
  ]
}
```

Para obtener más información, consulte [Obtención de información sobre un famoso](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [GetCelebrityInfo](#) la Referencia de AWS CLI comandos.

## get-celebrity-recognition

En el siguiente ejemplo de código, se muestra cómo usar `get-celebrity-recognition`.

AWS CLI

Para obtener los resultados de una operación de reconocimiento de celebridades

El siguiente `get-celebrity-recognition` comando muestra los resultados de una operación de reconocimiento de celebridades a la que ya ha llamado `start-celebrity-recognition`.

```
aws rekognition get-celebrity-recognition \
  --job-id 1234567890abcdef1234567890abcdef1234567890abcdef
```

Salida:

```
{
  "NextToken": "3D01Clx1CiT31VsRDkA03IybLb/h5AtDWSGuhYi
+N1FIJwwPtAkuKzDhL2rV3GcwmNt77+12",
  "Celebrities": [
    {
      "Timestamp": 0,
      "Celebrity": {
        "Confidence": 96.0,

```

```
"Face": {
  "BoundingBox": {
    "Width": 0.70333331823349,
    "Top": 0.16750000417232513,
    "Left": 0.19555555284023285,
    "Height": 0.3956249952316284
  },
  "Landmarks": [
    {
      "Y": 0.31031012535095215,
      "X": 0.441436767578125,
      "Type": "eyeLeft"
    },
    {
      "Y": 0.3081788718700409,
      "X": 0.6437258720397949,
      "Type": "eyeRight"
    },
    {
      "Y": 0.39542075991630554,
      "X": 0.5572493076324463,
      "Type": "nose"
    },
    {
      "Y": 0.4597957134246826,
      "X": 0.4579732120037079,
      "Type": "mouthLeft"
    },
    {
      "Y": 0.45688048005104065,
      "X": 0.6349081993103027,
      "Type": "mouthRight"
    }
  ],
  "Pose": {
    "Yaw": 8.943398475646973,
    "Roll": -2.0309247970581055,
    "Pitch": -0.5674862861633301
  },
  "Quality": {
    "Sharpness": 99.40211486816406,
    "Brightness": 89.47132110595703
  },
  "Confidence": 99.99861145019531
}
```

```
    },
    "Name": "CelebrityA",
    "Urls": [
      "www.imdb.com/name/111111111"
    ],
    "Id": "nnnnnn"
  }
},
{
  "Timestamp": 467,
  "Celebrity": {
    "Confidence": 99.0,
    "Face": {
      "BoundingBox": {
        "Width": 0.6877777576446533,
        "Top": 0.18437500298023224,
        "Left": 0.20555555820465088,
        "Height": 0.3868750035762787
      },
      "Landmarks": [
        {
          "Y": 0.31895750761032104,
          "X": 0.4411413371562958,
          "Type": "eyeLeft"
        },
        {
          "Y": 0.3140959143638611,
          "X": 0.6523157954216003,
          "Type": "eyeRight"
        },
        {
          "Y": 0.4016456604003906,
          "X": 0.5682755708694458,
          "Type": "nose"
        },
        {
          "Y": 0.46894142031669617,
          "X": 0.4597797095775604,
          "Type": "mouthLeft"
        },
        {
          "Y": 0.46971091628074646,
          "X": 0.6286435127258301,
          "Type": "mouthRight"
        }
      ]
    }
  }
}
```



```

        }
      ],
      "Pose": {
        "Yaw": 10.433465957641602,
        "Roll": -3.347442388534546,
        "Pitch": 1.3709543943405151
      },
      "Quality": {
        "Sharpness": 99.5531005859375,
        "Brightness": 88.5764389038086
      },
      "Confidence": 99.99148559570312
    },
    "Name": "Jane Celebrity",
    "Urls": [
      "www.imdb.com/name/111111111"
    ],
    "Id": "nnnnnn"
  }
}
],
"JobStatus": "SUCCEEDED",
"VideoMetadata": {
  "Format": "QuickTime / MOV",
  "FrameRate": 29.978118896484375,
  "Codec": "h264",
  "DurationMillis": 4570,
  "FrameHeight": 1920,
  "FrameWidth": 1080
}
}

```

Para obtener más información, consulte [Reconocimiento de celebridades en un vídeo almacenado](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [GetCelebrityRecognition](#) comandos AWS CLI .

## get-content-moderation

En el siguiente ejemplo de código, se muestra cómo usar `get-content-moderation`.

## AWS CLI

Para obtener los resultados de una operación de contenido no seguro

El siguiente `get-content-moderation` comando muestra los resultados de una operación de contenido no seguro a la que se inició anteriormente mediante una llamada `start-content-moderation`.

```
aws rekognition get-content-moderation \  
  --job-id 1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef
```

Salida:

```
{  
  "NextToken": "dlhcKMHMzpCBGFukz6I03JMcWiJAamCVhXht3r6b4b5Tfbyw3q7o+Jeezt  
+ZpgfOnW9FCCgQ",  
  "ModerationLabels": [  
    {  
      "Timestamp": 0,  
      "ModerationLabel": {  
        "Confidence": 97.39583587646484,  
        "ParentName": "",  
        "Name": "Violence"  
      }  
    },  
    {  
      "Timestamp": 0,  
      "ModerationLabel": {  
        "Confidence": 97.39583587646484,  
        "ParentName": "Violence",  
        "Name": "Weapon Violence"  
      }  
    }  
  ],  
  "JobStatus": "SUCCEEDED",  
  "VideoMetadata": {  
    "Format": "QuickTime / MOV",  
    "FrameRate": 29.97515869140625,  
    "Codec": "h264",  
    "DurationMillis": 6039,  
    "FrameHeight": 1920,  
    "FrameWidth": 1080  
  }  
}
```

```
}
```

Para obtener más información, consulte [Detección de vídeos almacenados no seguros](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [GetContentModeration](#) comandos AWS CLI .

## get-face-detection

En el siguiente ejemplo de código, se muestra cómo usar `get-face-detection`.

### AWS CLI

Para obtener los resultados de una operación de detección de rostros

El siguiente `get-face-detection` comando muestra los resultados de una operación de detección de rostros que usted inició anteriormente con una llamada `start-face-detection`.

```
aws rekognition get-face-detection \  
  --job-id 1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef
```

Salida:

```
{  
  "Faces": [  
    {  
      "Timestamp": 467,  
      "Face": {  
        "BoundingBox": {  
          "Width": 0.1560753583908081,  
          "Top": 0.13555361330509186,  
          "Left": -0.0952017530798912,  
          "Height": 0.6934483051300049  
        },  
        "Landmarks": [  
          {  
            "Y": 0.4013825058937073,  
            "X": -0.041750285774469376,  
            "Type": "eyeLeft"  
          },  
          {  
            "Y": 0.41695496439933777,
```

```
        "X": 0.027979329228401184,  
        "Type": "eyeRight"  
    },  
    {  
        "Y": 0.6375303268432617,  
        "X": -0.04034662991762161,  
        "Type": "mouthLeft"  
    },  
    {  
        "Y": 0.6497718691825867,  
        "X": 0.013960429467260838,  
        "Type": "mouthRight"  
    },  
    {  
        "Y": 0.5238034129142761,  
        "X": 0.008022055961191654,  
        "Type": "nose"  
    }  
],  
"Pose": {  
    "Yaw": -58.07863998413086,  
    "Roll": 1.9384294748306274,  
    "Pitch": -24.66305160522461  
},  
"Quality": {  
    "Sharpness": 83.14741516113281,  
    "Brightness": 25.75942611694336  
},  
"Confidence": 87.7622299194336  
}  
},  
{  
    "Timestamp": 967,  
    "Face": {  
        "BoundingBox": {  
            "Width": 0.28559377789497375,  
            "Top": 0.19436298310756683,  
            "Left": 0.024553587660193443,  
            "Height": 0.7216082215309143  
        },  
        "Landmarks": [  
            {  
                "Y": 0.4650231599807739,  
                "X": 0.16269078850746155,
```

```
        "Type": "eyeLeft"
      },
      {
        "Y": 0.4843238294124603,
        "X": 0.2782580852508545,
        "Type": "eyeRight"
      },
      {
        "Y": 0.71530681848526,
        "X": 0.1741468608379364,
        "Type": "mouthLeft"
      },
      {
        "Y": 0.7310671210289001,
        "X": 0.26857468485832214,
        "Type": "mouthRight"
      },
      {
        "Y": 0.582602322101593,
        "X": 0.2566150426864624,
        "Type": "nose"
      }
    ],
    "Pose": {
      "Yaw": 11.487052917480469,
      "Roll": 5.074230670928955,
      "Pitch": 15.396159172058105
    },
    "Quality": {
      "Sharpness": 73.32209777832031,
      "Brightness": 54.96497344970703
    },
    "Confidence": 99.99998474121094
  }
}
],
"NextToken":
"0zL223pDKy91160/02KXRqFIEAwxy4PkgYcm3hSo0rdysbXg5Ex0eFgTGEj0ADEac6S037U",
"JobStatus": "SUCCEEDED",
"VideoMetadata": {
  "Format": "QuickTime / MOV",
  "FrameRate": 29.970617294311523,
  "Codec": "h264",
  "DurationMillis": 6806,
```

```
    "FrameHeight": 1080,  
    "FrameWidth": 1920  
  }  
}
```

Para obtener más información, consulte [Detección de rostros en un vídeo almacenado](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [GetFaceDetection](#) comandos AWS CLI .

## get-face-search

En el siguiente ejemplo de código, se muestra cómo usar `get-face-search`.

### AWS CLI

Para obtener los resultados de una operación de búsqueda de rostros

El siguiente `get-face-search` comando muestra los resultados de una operación de búsqueda de rostros que usted inició anteriormente con una llamada `start-face-search`.

```
aws rekognition get-face-search \  
  --job-id 1234567890abcdef1234567890abcdef1234567890abcdef
```

Salida:

```
{  
  "Persons": [  
    {  
      "Timestamp": 467,  
      "FaceMatches": [],  
      "Person": {  
        "Index": 0,  
        "Face": {  
          "BoundingBox": {  
            "Width": 0.1560753583908081,  
            "Top": 0.13555361330509186,  
            "Left": -0.0952017530798912,  
            "Height": 0.6934483051300049  
          },  
          "Landmarks": [  
            {
```

```
        "Y": 0.4013825058937073,
        "X": -0.041750285774469376,
        "Type": "eyeLeft"
    },
    {
        "Y": 0.41695496439933777,
        "X": 0.027979329228401184,
        "Type": "eyeRight"
    },
    {
        "Y": 0.6375303268432617,
        "X": -0.04034662991762161,
        "Type": "mouthLeft"
    },
    {
        "Y": 0.6497718691825867,
        "X": 0.013960429467260838,
        "Type": "mouthRight"
    },
    {
        "Y": 0.5238034129142761,
        "X": 0.008022055961191654,
        "Type": "nose"
    }
  ],
  "Pose": {
    "Yaw": -58.07863998413086,
    "Roll": 1.9384294748306274,
    "Pitch": -24.66305160522461
  },
  "Quality": {
    "Sharpness": 83.14741516113281,
    "Brightness": 25.75942611694336
  },
  "Confidence": 87.7622299194336
}
},
{
  "Timestamp": 967,
  "FaceMatches": [
    {
      "Face": {
        "BoundingBox": {
```

```
        "Width": 0.12368900328874588,
        "Top": 0.16007399559020996,
        "Left": 0.5901259779930115,
        "Height": 0.2514039874076843
    },
    "FaceId": "056a95fa-2060-4159-9cab-7ed4daa030fa",
    "ExternalImageId": "image3.jpg",
    "Confidence": 100.0,
    "ImageId": "08f8a078-8929-37fd-8e8f-aadf690e8232"
},
"Similarity": 98.44476318359375
}
],
"Person": {
  "Index": 1,
  "Face": {
    "BoundingBox": {
      "Width": 0.28559377789497375,
      "Top": 0.19436298310756683,
      "Left": 0.024553587660193443,
      "Height": 0.7216082215309143
    },
    "Landmarks": [
      {
        "Y": 0.4650231599807739,
        "X": 0.16269078850746155,
        "Type": "eyeLeft"
      },
      {
        "Y": 0.4843238294124603,
        "X": 0.2782580852508545,
        "Type": "eyeRight"
      },
      {
        "Y": 0.71530681848526,
        "X": 0.1741468608379364,
        "Type": "mouthLeft"
      },
      {
        "Y": 0.7310671210289001,
        "X": 0.26857468485832214,
        "Type": "mouthRight"
      }
    ]
  }
}
```



```

        "Y": 0.582602322101593,
        "X": 0.2566150426864624,
        "Type": "nose"
    }
  ],
  "Pose": {
    "Yaw": 11.487052917480469,
    "Roll": 5.074230670928955,
    "Pitch": 15.396159172058105
  },
  "Quality": {
    "Sharpness": 73.32209777832031,
    "Brightness": 54.96497344970703
  },
  "Confidence": 99.99998474121094
}
}
}
],
"NextToken": "5bkgcezyuaqhtWk3C80TW6cjRghrwV9XDMivm5B3MXm+Lv6G+L+GejyFHPhoNa/
ldXIC4c/d",
"JobStatus": "SUCCEEDED",
"VideoMetadata": {
  "Format": "QuickTime / MOV",
  "FrameRate": 29.970617294311523,
  "Codec": "h264",
  "DurationMillis": 6806,
  "FrameHeight": 1080,
  "FrameWidth": 1920
}
}
}

```

Para obtener más información, consulte [Búsqueda de rostros en vídeos almacenados en la Guía para desarrolladores de Amazon Rekognition](#).

- Para API obtener más información, consulte la Referencia de [GetFaceSearch](#) comandos AWS CLI .

## get-label-detection

En el siguiente ejemplo de código, se muestra cómo usar `get-label-detection`.

## AWS CLI

Para obtener los resultados de una operación de detección de objetos y escenas

El siguiente `get-label-detection` comando muestra los resultados de una operación de detección de objetos y escenas que ya había iniciado con una llamada `start-label-detection`.

```
aws rekognition get-label-detection \
  --job-id 1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef
```

Salida:

```
{
  "Labels": [
    {
      "Timestamp": 0,
      "Label": {
        "Instances": [],
        "Confidence": 50.19071578979492,
        "Parents": [
          {
            "Name": "Person"
          },
          {
            "Name": "Crowd"
          }
        ],
        "Name": "Audience"
      }
    },
    {
      "Timestamp": 0,
      "Label": {
        "Instances": [],
        "Confidence": 55.74115753173828,
        "Parents": [
          {
            "Name": "Room"
          },
          {
            "Name": "Indoors"
          }
        ],

```

```

        {
            "Name": "School"
        }
    ],
    "Name": "Classroom"
}
}
],
"JobStatus": "SUCCEEDED",
"LabelModelVersion": "2.0",
"VideoMetadata": {
    "Format": "QuickTime / MOV",
    "FrameRate": 29.970617294311523,
    "Codec": "h264",
    "DurationMillis": 6806,
    "FrameHeight": 1080,
    "FrameWidth": 1920
},
"NextToken": "BMugzAi4L72IERzQdbpyMQuEFBsjl05W0Yx3mfG+sR9mm98E1/
Cp0benspRfs/5FBQFs4X7G"
}

```

Para obtener más información, consulte [Detección de etiquetas en un vídeo](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [GetLabelDetection](#) comandos AWS CLI .

## get-person-tracking

En el siguiente ejemplo de código, se muestra cómo usar `get-person-tracking`.

### AWS CLI

Para obtener los resultados de una operación de localización de personas

El siguiente `get-person-tracking` comando muestra los resultados de una operación de localización de personas a la que se inició anteriormente al llamar. `start-person-tracking`

```

aws rekognition get-person-tracking \
  --job-id 1234567890abcdef1234567890abcdef1234567890abcdef

```

Salida:

```
{
  "Persons": [
    {
      "Timestamp": 500,
      "Person": {
        "BoundingBox": {
          "Width": 0.4151041805744171,
          "Top": 0.07870370149612427,
          "Left": 0.0,
          "Height": 0.9212962985038757
        },
        "Index": 0
      }
    },
    {
      "Timestamp": 567,
      "Person": {
        "BoundingBox": {
          "Width": 0.4755208194255829,
          "Top": 0.07777778059244156,
          "Left": 0.0,
          "Height": 0.9194444417953491
        },
        "Index": 0
      }
    }
  ],
  "NextToken": "D/vRIYnyhG79ugdta3f+8cRg9oSro+HigG0uxRiYpTn0ExnqTi1CJektVAc4HrAXDv25eHYk",
  "JobStatus": "SUCCEEDED",
  "VideoMetadata": {
    "Format": "QuickTime / MOV",
    "FrameRate": 29.970617294311523,
    "Codec": "h264",
    "DurationMillis": 6806,
    "FrameHeight": 1080,
    "FrameWidth": 1920
  }
}
```

Para obtener más información, consulte [People Pathing](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para obtener API más información, consulte la Referencia de comandos [GetPersonTracking](#).AWS CLI

## index-faces

En el siguiente ejemplo de código, se muestra cómo usar `index-faces`.

Para obtener información, consulte [Adición de rostros a una colección](#).

### AWS CLI

#### Adición de rostros a una colección

El siguiente comando `index-faces` añade los rostros que se encuentran en una imagen a la colección especificada.

```
aws rekognition index-faces \  
  --image '{"S3Object":{"Bucket":"MyVideoS3Bucket","Name":"MyPicture.jpg"}}' \  
  --collection-id MyCollection \  
  --max-faces 1 \  
  --quality-filter "AUTO" \  
  --detection-attributes "ALL" \  
  --external-image-id "MyPicture.jpg"
```

#### Salida:

```
{  
  "FaceRecords": [  
    {  
      "FaceDetail": {  
        "Confidence": 99.993408203125,  
        "Eyeglasses": {  
          "Confidence": 99.11750030517578,  
          "Value": false  
        },  
        "Sunglasses": {  
          "Confidence": 99.98249053955078,  
          "Value": false  
        },  
        "Gender": {  
          "Confidence": 99.92769622802734,  
          "Value": "Male"  
        }  
      },  
    },  
  ],  
}
```

```
"Landmarks": [  
  {  
    "Y": 0.26750367879867554,  
    "X": 0.6202793717384338,  
    "Type": "eyeLeft"  
  },  
  {  
    "Y": 0.26642778515815735,  
    "X": 0.6787431836128235,  
    "Type": "eyeRight"  
  },  
  {  
    "Y": 0.31361380219459534,  
    "X": 0.6421601176261902,  
    "Type": "nose"  
  },  
  {  
    "Y": 0.3495299220085144,  
    "X": 0.6216195225715637,  
    "Type": "mouthLeft"  
  },  
  {  
    "Y": 0.35194727778434753,  
    "X": 0.669899046421051,  
    "Type": "mouthRight"  
  },  
  {  
    "Y": 0.26844894886016846,  
    "X": 0.6210268139839172,  
    "Type": "leftPupil"  
  },  
  {  
    "Y": 0.26707562804222107,  
    "X": 0.6817160844802856,  
    "Type": "rightPupil"  
  },  
  {  
    "Y": 0.24834522604942322,  
    "X": 0.6018546223640442,  
    "Type": "leftEyeBrowLeft"  
  },  
  {  
    "Y": 0.24397172033786774,  
    "X": 0.6172008514404297,
```

```
        "Type": "leftEyeBrowUp"
    },
    {
        "Y": 0.24677404761314392,
        "X": 0.6339119076728821,
        "Type": "leftEyeBrowRight"
    },
    {
        "Y": 0.24582654237747192,
        "X": 0.6619398593902588,
        "Type": "rightEyeBrowLeft"
    },
    {
        "Y": 0.23973053693771362,
        "X": 0.6804757118225098,
        "Type": "rightEyeBrowUp"
    },
    {
        "Y": 0.24441994726657867,
        "X": 0.6978968977928162,
        "Type": "rightEyeBrowRight"
    },
    {
        "Y": 0.2695908546447754,
        "X": 0.6085202693939209,
        "Type": "leftEyeLeft"
    },
    {
        "Y": 0.26716896891593933,
        "X": 0.6315826177597046,
        "Type": "leftEyeRight"
    },
    {
        "Y": 0.26289820671081543,
        "X": 0.6202316880226135,
        "Type": "leftEyeUp"
    },
    {
        "Y": 0.27123287320137024,
        "X": 0.6205548048019409,
        "Type": "leftEyeDown"
    },
    {
        "Y": 0.2668408751487732,
```

```
        "X": 0.6663622260093689,  
        "Type": "rightEyeLeft"  
    },  
    {  
        "Y": 0.26741549372673035,  
        "X": 0.6910083889961243,  
        "Type": "rightEyeRight"  
    },  
    {  
        "Y": 0.2614026665687561,  
        "X": 0.6785826086997986,  
        "Type": "rightEyeUp"  
    },  
    {  
        "Y": 0.27075251936912537,  
        "X": 0.6789616942405701,  
        "Type": "rightEyeDown"  
    },  
    {  
        "Y": 0.3211299479007721,  
        "X": 0.6324167847633362,  
        "Type": "noseLeft"  
    },  
    {  
        "Y": 0.32276326417922974,  
        "X": 0.6558475494384766,  
        "Type": "noseRight"  
    },  
    {  
        "Y": 0.34385165572166443,  
        "X": 0.6444970965385437,  
        "Type": "mouthUp"  
    },  
    {  
        "Y": 0.3671635091304779,  
        "X": 0.6459195017814636,  
        "Type": "mouthDown"  
    }  
],  
"Pose": {  
    "Yaw": -9.54541015625,  
    "Roll": -0.5709401965141296,  
    "Pitch": 0.6045494675636292  
},
```



```
"Emotions": [  
  {  
    "Confidence": 39.90074157714844,  
    "Type": "HAPPY"  
  },  
  {  
    "Confidence": 23.38753890991211,  
    "Type": "CALM"  
  },  
  {  
    "Confidence": 5.840933322906494,  
    "Type": "CONFUSED"  
  }  
],  
"AgeRange": {  
  "High": 63,  
  "Low": 45  
},  
"EyesOpen": {  
  "Confidence": 99.80887603759766,  
  "Value": true  
},  
"BoundingBox": {  
  "Width": 0.18562500178813934,  
  "Top": 0.1618015021085739,  
  "Left": 0.5575000047683716,  
  "Height": 0.24770642817020416  
},  
"Smile": {  
  "Confidence": 99.69740295410156,  
  "Value": false  
},  
"MouthOpen": {  
  "Confidence": 99.97393798828125,  
  "Value": false  
},  
"Quality": {  
  "Sharpness": 95.54405975341797,  
  "Brightness": 63.867706298828125  
},  
"Mustache": {  
  "Confidence": 97.05007934570312,  
  "Value": false  
},  
},
```

```
        "Beard": {
            "Confidence": 87.34505462646484,
            "Value": false
        },
        "Face": {
            "BoundingBox": {
                "Width": 0.18562500178813934,
                "Top": 0.1618015021085739,
                "Left": 0.5575000047683716,
                "Height": 0.24770642817020416
            },
            "FaceId": "ce7ed422-2132-4a11-ab14-06c5c410f29f",
            "ExternalImageId": "example-image.jpg",
            "Confidence": 99.993408203125,
            "ImageId": "8d67061e-90d2-598f-9fbd-29c8497039c0"
        }
    ],
    "UnindexedFaces": [],
    "FaceModelVersion": "3.0",
    "OrientationCorrection": "ROTATE_0"
}
```

Para obtener más información, consulte [Agregar rostros a una colección](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [IndexFaces](#) la Referencia de AWS CLI comandos.

## list-collections

En el siguiente ejemplo de código, se muestra cómo usar `list-collections`.

Para obtener información, consulte [Enumerar colecciones](#).

### AWS CLI

Creación de una lista de las colecciones disponibles

El siguiente `list-collections` comando muestra las colecciones disponibles en la AWS cuenta.

```
aws rekognition list-collections
```

Salida:

```
{
  "FaceModelVersions": [
    "2.0",
    "3.0",
    "3.0",
    "3.0",
    "4.0",
    "1.0",
    "3.0",
    "4.0",
    "4.0",
    "4.0"
  ],
  "CollectionIds": [
    "MyCollection1",
    "MyCollection2",
    "MyCollection3",
    "MyCollection4",
    "MyCollection5",
    "MyCollection6",
    "MyCollection7",
    "MyCollection8",
    "MyCollection9",
    "MyCollection10"
  ]
}
```

Para obtener más información, consulte [Listado de colecciones](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [ListCollections](#) la Referencia de AWS CLI comandos.

## **list-faces**

En el siguiente ejemplo de código, se muestra cómo usar `list-faces`.

Para obtener información, consulte [Enumerar rostros en una colección](#).

## AWS CLI

### Creación de una lista de rostros en una colección

El siguiente comando `list-faces` muestra los rostros de la colección especificada.

```
aws rekognition list-faces \  
  --collection-id MyCollection
```

Salida:

```
{  
  "FaceModelVersion": "3.0",  
  "Faces": [  
    {  
      "BoundingBox": {  
        "Width": 0.5216310024261475,  
        "Top": 0.3256250023841858,  
        "Left": 0.13394300639629364,  
        "Height": 0.3918749988079071  
      },  
      "FaceId": "0040279c-0178-436e-b70a-e61b074e96b0",  
      "ExternalImageId": "image1.jpg",  
      "Confidence": 100.0,  
      "ImageId": "f976e487-3719-5e2d-be8b-ea2724c26991"  
    },  
    {  
      "BoundingBox": {  
        "Width": 0.5074880123138428,  
        "Top": 0.3774999976158142,  
        "Left": 0.18302799761295319,  
        "Height": 0.3812499940395355  
      },  
      "FaceId": "086261e8-6deb-4bc0-ac73-ab22323cc38d",  
      "ExternalImageId": "image2.jpg",  
      "Confidence": 99.99930572509766,  
      "ImageId": "ae1593b0-a8f6-5e24-a306-abf529e276fa"  
    },  
    {  
      "BoundingBox": {  
        "Width": 0.5574039816856384,  
        "Top": 0.37187498807907104,  
        "Left": 0.14559100568294525,
```

```
        "Height": 0.4181250035762787
      },
      "FaceId": "11c4bd3c-19c5-4eb8-aecc-24feb93a26e1",
      "ExternalImageId": "image3.jpg",
      "Confidence": 99.99960327148438,
      "ImageId": "80739b4d-883f-5b78-97cf-5124038e26b9"
    },
    {
      "BoundingBox": {
        "Width": 0.18562500178813934,
        "Top": 0.1618019938468933,
        "Left": 0.5575000047683716,
        "Height": 0.24770599603652954
      },
      "FaceId": "13692fe4-990a-4679-b14a-5ac23d135eab",
      "ExternalImageId": "image4.jpg",
      "Confidence": 99.99340057373047,
      "ImageId": "8df18239-9ad1-5acd-a46a-6581ff98f51b"
    },
    {
      "BoundingBox": {
        "Width": 0.5307819843292236,
        "Top": 0.2862499952316284,
        "Left": 0.1564060002565384,
        "Height": 0.3987500071525574
      },
      "FaceId": "2eb5f3fd-e2a9-4b1c-a89f-afa0a518fe06",
      "ExternalImageId": "image5.jpg",
      "Confidence": 99.99970245361328,
      "ImageId": "3c314792-197d-528d-bbb6-798ed012c150"
    },
    {
      "BoundingBox": {
        "Width": 0.5773710012435913,
        "Top": 0.34437501430511475,
        "Left": 0.12396000325679779,
        "Height": 0.4337500035762787
      },
      "FaceId": "57189455-42b0-4839-a86c-abda48b13174",
      "ExternalImageId": "image6.jpg",
      "Confidence": 100.0,
      "ImageId": "0aff2f37-e7a2-5dbc-a3a3-4ef6ec18eaa0"
    }
  ],
  {
```

```
    "BoundingBox": {
      "Width": 0.5349419713020325,
      "Top": 0.29124999046325684,
      "Left": 0.16389399766921997,
      "Height": 0.40187498927116394
    },
    "FaceId": "745f7509-b1fa-44e0-8b95-367b1359638a",
    "ExternalImageId": "image7.jpg",
    "Confidence": 99.99979400634766,
    "ImageId": "67a34327-48d1-5179-b042-01e52ccfeada"
  },
  {
    "BoundingBox": {
      "Width": 0.41499999165534973,
      "Top": 0.09187500178813934,
      "Left": 0.28083300590515137,
      "Height": 0.3112500011920929
    },
    "FaceId": "8d3cfc70-4ba8-4b36-9644-90fba29c2dac",
    "ExternalImageId": "image8.jpg",
    "Confidence": 99.99769592285156,
    "ImageId": "a294da46-2cb1-5cc4-9045-61d7ca567662"
  },
  {
    "BoundingBox": {
      "Width": 0.48166701197624207,
      "Top": 0.20999999344348907,
      "Left": 0.21250000596046448,
      "Height": 0.36125001311302185
    },
    "FaceId": "bd4ceb4d-9acc-4ab7-8ef8-1c2d2ba0a66a",
    "ExternalImageId": "image9.jpg",
    "Confidence": 99.99949645996094,
    "ImageId": "5e1a7588-e5a0-5ee3-bd00-c642518dfe3a"
  },
  {
    "BoundingBox": {
      "Width": 0.18562500178813934,
      "Top": 0.1618019938468933,
      "Left": 0.5575000047683716,
      "Height": 0.24770599603652954
    },
    "FaceId": "ce7ed422-2132-4a11-ab14-06c5c410f29f",
    "ExternalImageId": "image10.jpg",
```

```
        "Confidence": 99.99340057373047,  
        "ImageId": "8d67061e-90d2-598f-9fbd-29c8497039c0"  
    }  
]  
}
```

Para obtener más información, consulte [Listado de rostros en una colección](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [ListFaces](#) la Referencia de AWS CLI comandos.

## list-stream-processors

En el siguiente ejemplo de código, se muestra cómo usar `list-stream-processors`.

### AWS CLI

Para ver una lista de los procesadores de streaming de su cuenta

El siguiente `list-stream-processors` comando muestra los procesadores de transmisión de su cuenta y el estado de cada uno de ellos.

```
aws rekognition list-stream-processors
```

Salida:

```
{  
  "StreamProcessors": [  
    {  
      "Status": "STOPPED",  
      "Name": "my-stream-processor"  
    }  
  ]  
}
```

Para obtener más información, consulte [Trabajar con vídeos en streaming](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [ListStreamProcessors](#) comandos AWS CLI .

## recognize-celebrities

En el siguiente ejemplo de código, se muestra cómo usar `recognize-celebrities`.

Para obtener información, consulte [Reconocimiento de famosos en una imagen](#).

### AWS CLI

#### Reconocimiento de famosos en una imagen

El siguiente comando `recognize-celebrities` reconoce a famosos en la imagen especificada almacenada en un bucket de Amazon S3:

```
aws rekognition recognize-celebrities \  
  --image "S3Object={Bucket=MyImageS3Bucket,Name=moviestars.jpg}"
```

Salida:

```
{  
  "UnrecognizedFaces": [  
    {  
      "BoundingBox": {  
        "Width": 0.14416666328907013,  
        "Top": 0.077777778059244156,  
        "Left": 0.625,  
        "Height": 0.2746031880378723  
      },  
      "Confidence": 99.9990234375,  
      "Pose": {  
        "Yaw": 10.80408763885498,  
        "Roll": -12.761146545410156,  
        "Pitch": 10.96889877319336  
      },  
      "Quality": {  
        "Sharpness": 94.1185531616211,  
        "Brightness": 79.18367004394531  
      },  
      "Landmarks": [  
        {  
          "Y": 0.18220913410186768,  
          "X": 0.6702951788902283,  
          "Type": "eyeLeft"  
        },  
        {
```



```
        "Y": 0.16337193548679352,
        "X": 0.7188183665275574,
        "Type": "eyeRight"
    },
    {
        "Y": 0.20739148557186127,
        "X": 0.7055801749229431,
        "Type": "nose"
    },
    {
        "Y": 0.2889308035373688,
        "X": 0.687512218952179,
        "Type": "mouthLeft"
    },
    {
        "Y": 0.2706988751888275,
        "X": 0.7250053286552429,
        "Type": "mouthRight"
    }
  ]
},
"CelebrityFaces": [
  {
    "MatchConfidence": 100.0,
    "Face": {
      "BoundingBox": {
        "Width": 0.14000000059604645,
        "Top": 0.1190476194024086,
        "Left": 0.82833331823349,
        "Height": 0.2666666805744171
      },
      "Confidence": 99.99359130859375,
      "Pose": {
        "Yaw": -10.509642601013184,
        "Roll": -14.51749324798584,
        "Pitch": 13.799399375915527
      },
      "Quality": {
        "Sharpness": 78.74752044677734,
        "Brightness": 42.201324462890625
      },
      "Landmarks": [
        {
```

```
        "Y": 0.2290833294391632,  
        "X": 0.8709492087364197,  
        "Type": "eyeLeft"  
    },  
    {  
        "Y": 0.20639978349208832,  
        "X": 0.9153988361358643,  
        "Type": "eyeRight"  
    },  
    {  
        "Y": 0.25417643785476685,  
        "X": 0.8907724022865295,  
        "Type": "nose"  
    },  
    {  
        "Y": 0.32729196548461914,  
        "X": 0.8876466155052185,  
        "Type": "mouthLeft"  
    },  
    {  
        "Y": 0.3115464746952057,  
        "X": 0.9238573312759399,  
        "Type": "mouthRight"  
    }  
    ]  
},  
"Name": "Celeb A",  
"Urls": [  
    "www.imdb.com/name/aaaaaaaaa"  
],  
"Id": "1111111"  
},  
{  
    "MatchConfidence": 97.0,  
    "Face": {  
        "BoundingBox": {  
            "Width": 0.13333334028720856,  
            "Top": 0.24920634925365448,  
            "Left": 0.4449999928474426,  
            "Height": 0.2539682686328888  
        },  
        "Confidence": 99.99979400634766,  
        "Pose": {  
            "Yaw": 6.557040691375732,
```

```
        "Roll": -7.316643714904785,  
        "Pitch": 9.272967338562012  
    },  
    "Quality": {  
        "Sharpness": 83.23492431640625,  
        "Brightness": 78.83267974853516  
    },  
    "Landmarks": [  
        {  
            "Y": 0.3625510632991791,  
            "X": 0.48898839950561523,  
            "Type": "eyeLeft"  
        },  
        {  
            "Y": 0.35366007685661316,  
            "X": 0.5313721299171448,  
            "Type": "eyeRight"  
        },  
        {  
            "Y": 0.3894785940647125,  
            "X": 0.5173314809799194,  
            "Type": "nose"  
        },  
        {  
            "Y": 0.44889405369758606,  
            "X": 0.5020005702972412,  
            "Type": "mouthLeft"  
        },  
        {  
            "Y": 0.4408611059188843,  
            "X": 0.5351271629333496,  
            "Type": "mouthRight"  
        }  
    ]  
},  
"Name": "Celeb B",  
"Urls": [  
    "www.imdb.com/name/bbbbbbbbbb"  
],  
"Id": "2222222"  
},  
{  
    "MatchConfidence": 100.0,  
    "Face": {
```

```
"BoundingBox": {
  "Width": 0.12416666746139526,
  "Top": 0.2968254089355469,
  "Left": 0.2150000035762787,
  "Height": 0.23650793731212616
},
"Confidence": 99.99958801269531,
"Pose": {
  "Yaw": 7.801797866821289,
  "Roll": -8.326810836791992,
  "Pitch": 7.844768047332764
},
"Quality": {
  "Sharpness": 86.93206024169922,
  "Brightness": 79.81291198730469
},
"Landmarks": [
  {
    "Y": 0.4027804136276245,
    "X": 0.2575301229953766,
    "Type": "eyeLeft"
  },
  {
    "Y": 0.3934555947780609,
    "X": 0.2956969439983368,
    "Type": "eyeRight"
  },
  {
    "Y": 0.4309830069541931,
    "X": 0.2837020754814148,
    "Type": "nose"
  },
  {
    "Y": 0.48186683654785156,
    "X": 0.26812544465065,
    "Type": "mouthLeft"
  },
  {
    "Y": 0.47338807582855225,
    "X": 0.29905644059181213,
    "Type": "mouthRight"
  }
]
},
```

```
"Name": "Celeb C",
"Urls": [
  "www.imdb.com/name/ccccccccc"
],
"Id": "3333333"
},
{
  "MatchConfidence": 97.0,
  "Face": {
    "BoundingBox": {
      "Width": 0.11916666477918625,
      "Top": 0.3698412775993347,
      "Left": 0.008333333767950535,
      "Height": 0.22698412835597992
    },
    "Confidence": 99.99999237060547,
    "Pose": {
      "Yaw": 16.38478660583496,
      "Roll": -1.0260354280471802,
      "Pitch": 5.975185394287109
    },
    "Quality": {
      "Sharpness": 83.23492431640625,
      "Brightness": 61.408443450927734
    },
    "Landmarks": [
      {
        "Y": 0.4632347822189331,
        "X": 0.049406956881284714,
        "Type": "eyeLeft"
      },
      {
        "Y": 0.46388113498687744,
        "X": 0.08722897619009018,
        "Type": "eyeRight"
      },
      {
        "Y": 0.5020678639411926,
        "X": 0.0758260041475296,
        "Type": "nose"
      },
      {
        "Y": 0.544157862663269,
        "X": 0.054029736667871475,
```

```

        "Type": "mouthLeft"
      },
      {
        "Y": 0.5463630557060242,
        "X": 0.08464983850717545,
        "Type": "mouthRight"
      }
    ]
  },
  "Name": "Celeb D",
  "Urls": [
    "www.imdb.com/name/ddddddddd"
  ],
  "Id": "44444444"
}
]
}

```

Para obtener más información, consulte [Reconocimiento de famosos en una imagen](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [RecognizeCelebrities](#) la Referencia de AWS CLI comandos.

## search-faces-by-image

En el siguiente ejemplo de código, se muestra cómo usar `search-faces-by-image`.

Para obtener información, consulte [Búsqueda de un rostro \(imagen\)](#).

### AWS CLI

Búsqueda de rostros en una colección que coincida con el rostro de mayor tamaño en una imagen.

El siguiente comando `search-faces-by-image` busca rostros en una colección que coincidan con el rostro más grande de la imagen especificada:

```

aws rekognition search-faces-by-image \
  --image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"ExamplePerson.jpg"}}' \
  --collection-id MyFaceImageCollection
{

```

```
"SearchedFaceBoundingBox": {
  "Width": 0.18562500178813934,
  "Top": 0.1618015021085739,
  "Left": 0.5575000047683716,
  "Height": 0.24770642817020416
},
"SearchedFaceConfidence": 99.993408203125,
"FaceMatches": [
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.18562500178813934,
        "Top": 0.1618019938468933,
        "Left": 0.5575000047683716,
        "Height": 0.24770599603652954
      },
      "FaceId": "ce7ed422-2132-4a11-ab14-06c5c410f29f",
      "ExternalImageId": "example-image.jpg",
      "Confidence": 99.99340057373047,
      "ImageId": "8d67061e-90d2-598f-9fbd-29c8497039c0"
    },
    "Similarity": 99.97913360595703
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.18562500178813934,
        "Top": 0.1618019938468933,
        "Left": 0.5575000047683716,
        "Height": 0.24770599603652954
      },
      "FaceId": "13692fe4-990a-4679-b14a-5ac23d135eab",
      "ExternalImageId": "image3.jpg",
      "Confidence": 99.99340057373047,
      "ImageId": "8df18239-9ad1-5acd-a46a-6581ff98f51b"
    },
    "Similarity": 99.97913360595703
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.41499999165534973,
        "Top": 0.09187500178813934,
        "Left": 0.28083300590515137,
```

```
        "Height": 0.3112500011920929
      },
      "FaceId": "8d3cfc70-4ba8-4b36-9644-90fba29c2dac",
      "ExternalImageId": "image2.jpg",
      "Confidence": 99.99769592285156,
      "ImageId": "a294da46-2cb1-5cc4-9045-61d7ca567662"
    },
    "Similarity": 99.18069458007812
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.48166701197624207,
        "Top": 0.20999999344348907,
        "Left": 0.21250000596046448,
        "Height": 0.36125001311302185
      },
      "FaceId": "bd4ceb4d-9acc-4ab7-8ef8-1c2d2ba0a66a",
      "ExternalImageId": "image1.jpg",
      "Confidence": 99.99949645996094,
      "ImageId": "5e1a7588-e5a0-5ee3-bd00-c642518dfe3a"
    },
    "Similarity": 98.66607666015625
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.5349419713020325,
        "Top": 0.29124999046325684,
        "Left": 0.16389399766921997,
        "Height": 0.40187498927116394
      },
      "FaceId": "745f7509-b1fa-44e0-8b95-367b1359638a",
      "ExternalImageId": "image9.jpg",
      "Confidence": 99.99979400634766,
      "ImageId": "67a34327-48d1-5179-b042-01e52ccfeada"
    },
    "Similarity": 98.24278259277344
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.5307819843292236,
        "Top": 0.2862499952316284,
```



```
        "Left": 0.1564060002565384,  
        "Height": 0.3987500071525574  
    },  
    "FaceId": "2eb5f3fd-e2a9-4b1c-a89f-afa0a518fe06",  
    "ExternalImageId": "image10.jpg",  
    "Confidence": 99.99970245361328,  
    "ImageId": "3c314792-197d-528d-bbb6-798ed012c150"  
},  
"Similarity": 98.10665893554688  
},  
{  
  "Face": {  
    "BoundingBox": {  
      "Width": 0.5074880123138428,  
      "Top": 0.3774999976158142,  
      "Left": 0.18302799761295319,  
      "Height": 0.3812499940395355  
    },  
    "FaceId": "086261e8-6deb-4bc0-ac73-ab22323cc38d",  
    "ExternalImageId": "image6.jpg",  
    "Confidence": 99.99930572509766,  
    "ImageId": "ae1593b0-a8f6-5e24-a306-abf529e276fa"  
  },  
  "Similarity": 98.10526275634766  
},  
{  
  "Face": {  
    "BoundingBox": {  
      "Width": 0.5574039816856384,  
      "Top": 0.37187498807907104,  
      "Left": 0.14559100568294525,  
      "Height": 0.4181250035762787  
    },  
    "FaceId": "11c4bd3c-19c5-4eb8-aecc-24feb93a26e1",  
    "ExternalImageId": "image5.jpg",  
    "Confidence": 99.99960327148438,  
    "ImageId": "80739b4d-883f-5b78-97cf-5124038e26b9"  
  },  
  "Similarity": 97.94659423828125  
},  
{  
  "Face": {  
    "BoundingBox": {  
      "Width": 0.5773710012435913,
```

```

        "Top": 0.34437501430511475,
        "Left": 0.12396000325679779,
        "Height": 0.4337500035762787
    },
    "FaceId": "57189455-42b0-4839-a86c-abda48b13174",
    "ExternalImageId": "image8.jpg",
    "Confidence": 100.0,
    "ImageId": "0aff2f37-e7a2-5dbc-a3a3-4ef6ec18eaa0"
},
"Similarity": 97.93476867675781
],
"FaceModelVersion": "3.0"
}

```

Para obtener más información, consulte [Búsqueda de un rostro utilizando una imagen](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [SearchFacesByImage](#) la Referencia de AWS CLI comandos.

## search-faces

En el siguiente ejemplo de código, se muestra cómo usar search-faces.

Para obtener información, consulte [Búsqueda de un rostro \(ID de rostro\)](#).

### AWS CLI

Búsqueda de rostros en una colección que coincidan con un ID de rostro.

El siguiente comando search-faces busca rostros en una colección que coincidan con el ID de rostro especificado.

```

aws rekognition search-faces \
  --face-id 8d3cfc70-4ba8-4b36-9644-90fba29c2dac \
  --collection-id MyCollection

```

Salida:

```

{
  "SearchedFaceId": "8d3cfc70-4ba8-4b36-9644-90fba29c2dac",

```

```
"FaceModelVersion": "3.0",
"FaceMatches": [
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.48166701197624207,
        "Top": 0.20999999344348907,
        "Left": 0.21250000596046448,
        "Height": 0.36125001311302185
      },
      "FaceId": "bd4ceb4d-9acc-4ab7-8ef8-1c2d2ba0a66a",
      "ExternalImageId": "image1.jpg",
      "Confidence": 99.99949645996094,
      "ImageId": "5e1a7588-e5a0-5ee3-bd00-c642518dfe3a"
    },
    "Similarity": 99.30997467041016
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.18562500178813934,
        "Top": 0.1618019938468933,
        "Left": 0.5575000047683716,
        "Height": 0.24770599603652954
      },
      "FaceId": "ce7ed422-2132-4a11-ab14-06c5c410f29f",
      "ExternalImageId": "example-image.jpg",
      "Confidence": 99.99340057373047,
      "ImageId": "8d67061e-90d2-598f-9fbd-29c8497039c0"
    },
    "Similarity": 99.24862670898438
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.18562500178813934,
        "Top": 0.1618019938468933,
        "Left": 0.5575000047683716,
        "Height": 0.24770599603652954
      },
      "FaceId": "13692fe4-990a-4679-b14a-5ac23d135eab",
      "ExternalImageId": "image3.jpg",
      "Confidence": 99.99340057373047,
      "ImageId": "8df18239-9ad1-5acd-a46a-6581ff98f51b"
    }
  }
]
```

```
    },
    "Similarity": 99.24862670898438
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.5349419713020325,
        "Top": 0.29124999046325684,
        "Left": 0.16389399766921997,
        "Height": 0.40187498927116394
      },
      "FaceId": "745f7509-b1fa-44e0-8b95-367b1359638a",
      "ExternalImageId": "image9.jpg",
      "Confidence": 99.99979400634766,
      "ImageId": "67a34327-48d1-5179-b042-01e52ccfeada"
    },
    "Similarity": 96.73158264160156
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.5307819843292236,
        "Top": 0.2862499952316284,
        "Left": 0.1564060002565384,
        "Height": 0.3987500071525574
      },
      "FaceId": "2eb5f3fd-e2a9-4b1c-a89f-afa0a518fe06",
      "ExternalImageId": "image10.jpg",
      "Confidence": 99.99970245361328,
      "ImageId": "3c314792-197d-528d-bbb6-798ed012c150"
    },
    "Similarity": 96.48291015625
  },
  {
    "Face": {
      "BoundingBox": {
        "Width": 0.5074880123138428,
        "Top": 0.3774999976158142,
        "Left": 0.18302799761295319,
        "Height": 0.3812499940395355
      },
      "FaceId": "086261e8-6deb-4bc0-ac73-ab22323cc38d",
      "ExternalImageId": "image6.jpg",
      "Confidence": 99.99930572509766,
    }
  }
}
```

```

        "ImageId": "ae1593b0-a8f6-5e24-a306-abf529e276fa"
    },
    "Similarity": 96.43287658691406
},
{
    "Face": {
        "BoundingBox": {
            "Width": 0.5574039816856384,
            "Top": 0.37187498807907104,
            "Left": 0.14559100568294525,
            "Height": 0.4181250035762787
        },
        "FaceId": "11c4bd3c-19c5-4eb8-aecc-24feb93a26e1",
        "ExternalImageId": "image5.jpg",
        "Confidence": 99.99960327148438,
        "ImageId": "80739b4d-883f-5b78-97cf-5124038e26b9"
    },
    "Similarity": 95.25305938720703
},
{
    "Face": {
        "BoundingBox": {
            "Width": 0.5773710012435913,
            "Top": 0.34437501430511475,
            "Left": 0.12396000325679779,
            "Height": 0.4337500035762787
        },
        "FaceId": "57189455-42b0-4839-a86c-abda48b13174",
        "ExternalImageId": "image8.jpg",
        "Confidence": 100.0,
        "ImageId": "0aff2f37-e7a2-5dbc-a3a3-4ef6ec18eaa0"
    },
    "Similarity": 95.22837829589844
}
]
}

```

Para obtener más información, consulte [Búsqueda de un rostro utilizando su ID de rostro](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte [SearchFaces](#) la Referencia de AWS CLI comandos.

## start-celebrity-recognition

En el siguiente ejemplo de código, se muestra cómo usar `start-celebrity-recognition`.

### AWS CLI

Para iniciar el reconocimiento de celebridades en un vídeo almacenado

El siguiente `start-celebrity-recognition` comando inicia un trabajo para buscar celebridades en el archivo de vídeo especificado almacenado en un bucket de Amazon S3.

```
aws rekognition start-celebrity-recognition \  
  --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Salida:

```
{  
  "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"  
}
```

Para obtener más información, consulte [Reconocimiento de celebridades en un vídeo almacenado](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [StartCelebrityRecognition](#) comandos AWS CLI .

## start-content-moderation

En el siguiente ejemplo de código, se muestra cómo usar `start-content-moderation`.

### AWS CLI

Para iniciar el reconocimiento del contenido no seguro en un vídeo almacenado

El siguiente `start-content-moderation` comando inicia un trabajo para detectar contenido no seguro en el archivo de vídeo especificado almacenado en un bucket de Amazon S3.

```
aws rekognition start-content-moderation \  
  --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Salida:

```
{
  "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"
}
```

Para obtener más información, consulte [Detección de vídeos almacenados no seguros](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [StartContentModeration](#) comandos AWS CLI .

## start-face-detection

En el siguiente ejemplo de código, se muestra cómo usar `start-face-detection`.

### AWS CLI

Para detectar rostros en un vídeo

El siguiente `start-face-detection` comando inicia un trabajo para detectar rostros en el archivo de vídeo especificado almacenado en un bucket de Amazon S3.

```
aws rekognition start-face-detection
  --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Salida:

```
{
  "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"
}
```

Para obtener más información, consulte [Detección de rostros en un vídeo almacenado](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [StartFaceDetection](#) comandos AWS CLI .

## start-face-search

En el siguiente ejemplo de código, se muestra cómo usar `start-face-search`.

## AWS CLI

Para buscar rostros en una colección que coincidan con los rostros detectados en un vídeo

El siguiente `start-face-search` comando inicia un trabajo para buscar rostros en una colección que coincidan con los rostros detectados en el archivo de vídeo especificado en un bucket de Amazon S3.

```
aws rekognition start-face-search \  
  --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}" \  
  --collection-id collection
```

Salida:

```
{  
  "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"  
}
```

Para obtener más información, consulte [Búsqueda de rostros en vídeos almacenados en la Guía para desarrolladores de Amazon Rekognition](#).

- Para API obtener más información, consulte la Referencia de [StartFaceSearch](#) comandos AWS CLI .

## start-label-detection

En el siguiente ejemplo de código, se muestra cómo usar `start-label-detection`.

## AWS CLI

Para detectar objetos y escenas en un vídeo

El siguiente `start-label-detection` comando inicia un trabajo para detectar objetos y escenas en el archivo de vídeo especificado almacenado en un bucket de Amazon S3.

```
aws rekognition start-label-detection \  
  --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Salida:

```
{  
  "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"
```



```
}
```

Para obtener más información, consulte [Detección de etiquetas en un vídeo](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [StartLabelDetection](#) comandos AWS CLI .

## start-person-tracking

En el siguiente ejemplo de código, se muestra cómo usar `start-person-tracking`.

### AWS CLI

Para iniciar la búsqueda de rutas de las personas en un vídeo almacenado

El siguiente `start-person-tracking` comando inicia un trabajo para rastrear las rutas que toman las personas en el archivo de vídeo especificado almacenado en un bucket de Amazon S3. :

```
aws rekognition start-person-tracking \  
  --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Salida:

```
{  
  "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"  
}
```

Para obtener más información, consulte [People Pathing](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para obtener API más información, consulte la Referencia de comandos [StartPersonTracking](#).AWS CLI

## start-stream-processor

En el siguiente ejemplo de código, se muestra cómo usar `start-stream-processor`.

### AWS CLI

Para iniciar un procesador de streaming

El siguiente `start-stream-processor` comando inicia el procesador de transmisión de vídeo especificado.

```
aws rekognition start-stream-processor \  
  --name my-stream-processor
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con vídeos en streaming](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [StartStreamProcessor](#) comandos AWS CLI .

## stop-stream-processor

En el siguiente ejemplo de código, se muestra cómo usar `stop-stream-processor`.

### AWS CLI

Para detener un procesador de flujo en ejecución

El siguiente `stop-stream-processor` comando detiene el procesador de flujo en ejecución especificado.

```
aws rekognition stop-stream-processor \  
  --name my-stream-processor
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con vídeos en streaming](#) en la Guía para desarrolladores de Amazon Rekognition.

- Para API obtener más información, consulte la Referencia de [StopStreamProcessor](#) comandos AWS CLI .

## AWS RAM ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS RAM.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **accept-resource-share-invitation**

En el siguiente ejemplo de código, se muestra cómo usar `accept-resource-share-invitation`.

AWS CLI

Para aceptar una invitación a compartir recursos

En el siguiente `accept-resource-share-invitation` ejemplo, se acepta la invitación a compartir recursos especificada. Los directores de la cuenta invitada pueden empezar a utilizar inmediatamente los recursos del recurso compartido.

```
aws ram accept-resource-share-invitation \
  --resource-share-invitation-arn arn:aws:ram:us-west-2:111111111111:resource-
  share-invitation/1e3477be-4a95-46b4-bbe0-c4001EXAMPLE
```

Salida:

```
{
  "resourceShareInvitation": {
    "resourceShareInvitationArn": "arn:aws:ram:us-west-2:111111111111:resource-
  share-invitation/1e3477be-4a95-46b4-bbe0-c4001EXAMPLE",
    "resourceShareName": "MyLicenseShare",
    "resourceShareArn": "arn:aws:ram:us-west-2:111111111111:resource-
  share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE",
    "senderAccountId": "111111111111",
    "receiverAccountId": "222222222222",
    "invitationTimestamp": "2021-09-22T15:07:35.620000-07:00",
```

```

    "status": "ACCEPTED"
  }
}

```

- Para API obtener más información, consulte [AcceptResourceShareInvitation](#) la Referencia de AWS CLI comandos.

## associate-resource-share-permission

En el siguiente ejemplo de código, se muestra cómo usar `associate-resource-share-permission`.

### AWS CLI

Para asociar un permiso RAM gestionado a un recurso compartido

En el siguiente `associate-resource-share-permission` ejemplo, se reemplaza el permiso administrado existente para el tipo de recurso correspondiente por el permiso administrado especificado. El acceso a todos los recursos del tipo de recurso correspondiente se rige por el nuevo permiso.

```

aws ram associate-resource-share-permission \
  --permission-arn arn:aws:ram::aws:permission/
AWSRAMPermissionGlueDatabaseReadWrite \
  --replace \
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE

```

Salida:

```

{
  "returnValue": true
}

```

- Para API obtener más información, consulte [AssociateResourceSharePermission](#) la Referencia de AWS CLI comandos.

## associate-resource-share

En el siguiente ejemplo de código, se muestra cómo usar `associate-resource-share`.

## AWS CLI

Ejemplo 1: Para asociar un recurso a un recurso compartido

En el siguiente `associate-resource-share` ejemplo, se agrega una configuración de licencia al recurso compartido especificado.

```
aws ram associate-resource-share \  
  --resource-share arn:aws:ram:us-west-2:123456789012:resource-  
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE \  
  --resource-arns arn:aws:license-manager:us-west-2:123456789012:license-  
configuration:lic-36be0485f5ae379cc74cf8e92EXAMPLE
```

Salida:

```
{  
  "resourceShareAssociations": [  
    {  
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-  
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE",  
      "associatedEntity": "arn:aws:license-manager:us-  
west-2:123456789012:license-configuration:lic-36be0485f5ae379cc74cf8e92EXAMPLE",  
      "associationType": "RESOURCE",  
      "status": "ASSOCIATING",  
      "external": false  
    }  
  ]  
}
```

Ejemplo 2: Para asociar un principal a un recurso compartido

En el siguiente `associate-resource-share` ejemplo, se concede acceso al recurso compartido especificado a todas las cuentas de la unidad organizativa especificada.

```
aws ram associate-resource-share \  
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-  
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE \  
  --principals arn:aws:organizations::123456789012:ou/o-63bEXAMPLE/ou-46xi-  
rEXAMPLE
```

Salida:

```
{
  "resourceShareAssociations": [
    {
      "status": "ASSOCIATING",
      "associationType": "PRINCIPAL",
      "associatedEntity": "arn:aws:organizations::123456789012:ou/o-63bEXAMPLE/ou-46xi-rEXAMPLE",
      "external": false,
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE"
    }
  ]
}
```

- Para API obtener más información, consulte [AssociateResourceShare](#) la Referencia de AWS CLI comandos.

## create-resource-share

En el siguiente ejemplo de código, se muestra cómo usar `create-resource-share`.

### AWS CLI

Ejemplo 1: Para crear un recurso compartido

En el siguiente `create-resource-share` ejemplo, se crea un recurso compartido vacío con el nombre especificado. Debe agregar los recursos, las entidades principales y los permisos al recurso compartido por separado.

```
aws ram create-resource-share \
  --name MyNewResourceShare
```

Salida:

```
{
  "resourceShare": {
    "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-share/4476c27d-8feb-4b21-afe9-7de23EXAMPLE",
    "name": "MyNewResourceShare",
    "owningAccountId": "123456789012",
    "allowExternalPrincipals": true,
  }
}
```

```

    "status": "ACTIVE",
    "creationTime": 1634586271.302,
    "lastUpdatedTime": 1634586271.302
  }
}

```

Ejemplo 2: Para crear un recurso compartido con AWS las cuentas como principales

En el siguiente `create-resource-share` ejemplo, se crea un recurso compartido y se concede acceso a la AWS cuenta especificada (222222222222). Si los directores especificados no forman parte de la misma AWS organización, las invitaciones se envían y deben aceptarse antes de conceder el acceso.

```

aws ram create-resource-share \
  --name MyNewResourceShare \
  --principals 222222222222

```

Ejemplo 3: Para crear un recurso compartido restringido a su organización AWS

En el siguiente `create-resource-share` ejemplo, se crea un recurso compartido que está restringido a las cuentas de la AWS organización a la que pertenece su cuenta y se agrega la OU especificada como principal. Todas las cuentas de esa OU pueden usar los recursos del recurso compartido.

```

aws ram create-resource-share \
  --name MyNewResourceShare \
  --no-allow-external-principals \
  --principals arn:aws:organizations::123456789012:ou/o-63bEXAMPLE/ou-46xi-  
rEXAMPLE

```

Salida:

```

{
  "resourceShare": {
    "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-  
share/7be8694e-095c-41ca-9ce8-7be4aEXAMPLE",
    "name": "MyNewResourceShare",
    "owningAccountId": "123456789012",
    "allowExternalPrincipals": false,
    "status": "ACTIVE",
    "creationTime": 1634587042.49,
    "lastUpdatedTime": 1634587042.49
  }
}

```

```
}  
}
```

- Para API obtener más información, consulte [CreateResourceShare](#) la Referencia de AWS CLI comandos.

## **delete-resource-share**

En el siguiente ejemplo de código, se muestra cómo usar `delete-resource-share`.

### AWS CLI

Para eliminar un recurso compartido

En el siguiente `delete-resource-share` ejemplo, se elimina el recurso compartido especificado.

```
aws ram delete-resource-share \  
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-  
b505-7e2a-420d-6f5d3EXAMPLE
```

El siguiente resultado indica que se ha realizado correctamente:

```
{  
  "returnValue": true  
}
```

- Para API obtener más información, consulte [DeleteResourceShare](#) la Referencia de AWS CLI comandos.

## **disassociate-resource-share-permission**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-resource-share-permission`.

### AWS CLI

Para eliminar un permiso RAM administrado para un tipo de recurso de un recurso compartido

En el siguiente `disassociate-resource-share-permission` ejemplo, se quita el permiso RAM administrado para las bases de datos de Glue del recurso compartido especificado.



```
aws ram disassociate-resource-share-permission \
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE \
  --permission-arn arn:aws:ram::aws:permission/
AWSRAMPermissionGlueDatabaseReadWrite
```

Salida:

```
{
  "returnValue": true
}
```

- Para API obtener más información, consulte [DisassociateResourceSharePermission](#) la Referencia de AWS CLI comandos.

## disassociate-resource-share

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-resource-share`.

AWS CLI

Para eliminar un recurso de un recurso compartido

En el siguiente `disassociate-resource-share` ejemplo, se quita el recurso especificado, en este caso una VPC subred, del recurso compartido especificado. Los principales con acceso al recurso compartido ya no pueden realizar operaciones en ese recurso.

```
aws ram disassociate-resource-share \
  --resource-arns arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0250c25a1fEXAMPLE \
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-
b505-7e2a-420d-6f5d3EXAMPLE
```

Salida:

```
{
  "resourceShareAssociations": [
    {
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
      "associatedEntity": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0250c25a1fEXAMPLE",
    }
  ]
}
```

```
    "associationType": "RESOURCE",
    "status": "DISASSOCIATING",
    "external": false
  ]
}
```

- Para API obtener más información, consulte [DisassociateResourceShare](#) la Referencia de AWS CLI comandos.

## **enable-sharing-with-aws-organization**

En el siguiente ejemplo de código, se muestra cómo usar `enable-sharing-with-aws-organization`.

### AWS CLI

Para permitir el uso compartido de recursos entre AWS Organizations

El siguiente `enable-sharing-with-aws-organization` ejemplo permite compartir recursos entre la organización y las unidades organizativas.

```
aws ram enable-sharing-with-aws-organization
```

El siguiente resultado indica éxito.

```
{
  "returnValue": true
}
```

- Para API obtener más información, consulte [EnableSharingWithAwsOrganization](#) la Referencia de AWS CLI comandos.

## **get-permission**

En el siguiente ejemplo de código, se muestra cómo usar `get-permission`.

### AWS CLI

Para recuperar los detalles de un permiso RAM gestionado

En el siguiente `get-permission` ejemplo, se muestran los detalles de la versión predeterminada del permiso RAM administrado especificado.

```
aws ram get-permission \
  --permission-arn arn:aws:ram::aws:permission/
AWSRAMPermissionGlueTableReadWriteForDatabase
```

Salida:

```
{
  "permission": {
    "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionGlueTableReadWriteForDatabase",
    "version": "2",
    "defaultVersion": true,
    "name": "AWSRAMPermissionGlueTableReadWriteForDatabase",
    "resourceType": "glue:Database",
    "permission": "{\"Effect\":\"Allow\",\"Action\":[\"glue:GetTable
\", \"glue:UpdateTable\", \"glue>DeleteTable\", \"glue:BatchDeleteTable\",
\", \"glue:BatchDeleteTableVersion\", \"glue:GetTableVersion\", \"glue:GetTableVersions
\", \"glue:GetPartition\", \"glue:GetPartitions\", \"glue:BatchGetPartition\",
\", \"glue:BatchCreatePartition\", \"glue>CreatePartition\", \"glue:UpdatePartition
\", \"glue:BatchDeletePartition\", \"glue>DeletePartition\", \"glue:GetTables\",
\", \"glue:SearchTables\"]}",
    "creationTime": 1624912434.431,
    "lastUpdatedTime": 1624912434.431,
    "isResourceTypeDefault": false
  }
}
```

- Para API obtener más información, consulte [GetPermission](#) la Referencia de AWS CLI comandos.

## **get-resource-policies**

En el siguiente ejemplo de código, se muestra cómo usar `get-resource-policies`.

AWS CLI

Para obtener las políticas de un recurso

En el siguiente `get-resource-policies` ejemplo, se muestran las políticas de permisos basadas en recursos para el recurso especificado asociado a un recurso compartido.

```
aws ram get-resource-policies \
  --resource-arns arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0250c25a1fEXAMPLE
```

Salida:

```
{
  "policies": [
    [{"Version": "2008-10-17", "Statement": [{"Sid": "RamStatement1",
      "Effect": "Allow", "Principal": {"AWS": []}, "Action": ["ec2:RunInstances",
      "ec2:CreateNetworkInterface", "ec2:DescribeSubnets"], "Resource":
      "arn:aws:ec2:us-west-2:123456789012:subnet/subnet-0250c25a1fEXAMPLE"}]}]
  ]
}
```

- Para API obtener más información, consulte la Referencia [GetResourcePolicies](#) de AWS CLI comandos.

## get-resource-share-associations

En el siguiente ejemplo de código, se muestra cómo usar `get-resource-share-associations`.

AWS CLI

Ejemplo 1: Para enumerar todas las asociaciones de recursos para todos los tipos de recursos

En el siguiente `get-resource-share-associations` ejemplo, se enumeran las asociaciones de recursos para todos los tipos de recursos en todos los recursos compartidos.

```
aws ram get-resource-share-associations \
  --association-type RESOURCE
```

Salida:

```
{
  "resourceShareAssociations": [
    {
```

```

        "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
        "associatedEntity": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0250c25a1fEXAMPLE",
        "resourceShareName": "MySubnetShare",
        "associationType": "RESOURCE",
        "status": "ASSOCIATED",
        "creationTime": 1565303590.973,
        "lastUpdatedTime": 1565303591.695,
        "external": false
    },
    {
        "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/8167bdfe-4480-4a01-8632-315e0EXAMPLE",
        "associatedEntity": "arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-36be0485f5ae379cc74cf8e92EXAMPLE",
        "resourceShareName": "MyLicenseShare",
        "associationType": "RESOURCE",
        "status": "ASSOCIATED",
        "creationTime": 1632342958.457,
        "lastUpdatedTime": 1632342958.907,
        "external": false
    }
]
}

```

Ejemplo 2: Para enumerar las asociaciones principales de un recurso compartido

En el siguiente `get-resource-share-associations` ejemplo, se enumeran solo las asociaciones principales únicamente para el recurso compartido especificado.

```

aws ram get-resource-share-associations \
  --resource-share-arns arn:aws:ram:us-west-2:123456789012:resource-
share/7be8694e-095c-41ca-9ce8-7be4aEXAMPLE \
  --association-type PRINCIPAL

```

Salida:

```

{
  "resourceShareAssociations": [
    {
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7be8694e-095c-41ca-9ce8-7be4aEXAMPLE",

```

```

        "resourceShareName": "MyNewResourceShare",
        "associatedEntity": "arn:aws:organizations::123456789012:ou/
o-63bEXAMPLE/ou-46xi-rEXAMPLE",
        "associationType": "PRINCIPAL",
        "status": "ASSOCIATED",
        "creationTime": 1634587042.49,
        "lastUpdatedTime": 1634587044.291,
        "external": false
    }
]
}

```

- Para API obtener más información, consulte [GetResourceShareAssociations](#) la Referencia de AWS CLI comandos.

## get-resource-share-invitations

En el siguiente ejemplo de código, se muestra cómo usar `get-resource-share-invitations`.

### AWS CLI

Para enumerar tus recursos, comparte invitaciones

En el siguiente `get-resource-share-invitations` ejemplo, se muestran las invitaciones actuales para compartir recursos.

```
aws ram get-resource-share-invitations
```

Salida:

```

{
  "resourceShareInvitations": [
    {
      "resourceShareInvitationArn": "arn:aws:ram:us-
west2-1:111111111111:resource-share-invitation/32b639f0-14b8-7e8f-55ea-
e6117EXAMPLE",
      "resourceShareName": "project-resource-share",
      "resourceShareArn": "arn:aws:ram:us-west-2:111111111111:resource-share/
fcb639f0-1449-4744-35bc-a983fEXAMPLE",
      "senderAccountId": "111111111111",
      "receiverAccountId": "222222222222",
      "invitationTimestamp": 1565312166.258,

```

```
        "status": "PENDING"
      }
    ]
  }
```

- Para API obtener más información, consulte [GetResourceShareInvitations](#) la Referencia de AWS CLI comandos.

## get-resource-shares

En el siguiente ejemplo de código, se muestra cómo usar `get-resource-shares`.

### AWS CLI

Ejemplo 1: Para enumerar los recursos compartidos que te pertenecen y los compartes con otros

En el siguiente `get-resource-shares` ejemplo, se enumeran los recursos compartidos que se han creado y que se están compartiendo con otros.

```
aws ram get-resource-shares \
  --resource-owner SELF
```

Salida:

```
{
  "resourceShares": [
    {
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/3ab63985-99d9-1cd2-7d24-75e93EXAMPLE",
      "name": "my-resource-share",
      "owningAccountId": "123456789012",
      "allowExternalPrincipals": false,
      "status": "ACTIVE",
      "tags": [
        {
          "key": "project",
          "value": "lima"
        }
      ]
      "creationTime": 1565295733.282,
      "lastUpdatedTime": 1565295733.282
    }
  ]
}
```

```
    },
    {
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
      "name": "my-resource-share",
      "owningAccountId": "123456789012",
      "allowExternalPrincipals": true,
      "status": "ACTIVE",
      "creationTime": 1565295733.282,
      "lastUpdatedTime": 1565295733.282
    }
  ]
}
```

Ejemplo 2: Para enumerar los recursos compartidos que son propiedad de otros y que se han compartido con usted

En el siguiente `get-resource-shares` ejemplo, se enumeran los recursos compartidos que otros han creado y compartido con usted. En este ejemplo, no hay ninguno.

```
aws ram get-resource-shares \
  --resource-owner OTHER-ACCOUNTS
```

Salida:

```
{
  "resourceShares": []
}
```

- Para API obtener más información, consulte [GetResourceShares](#) la Referencia de AWS CLI comandos.

## **list-pending-invitation-resources**

En el siguiente ejemplo de código, se muestra cómo usar `list-pending-invitation-resources`.

AWS CLI

Para enumerar los recursos que están disponibles en un recurso compartido pendiente



En el siguiente `list-pending-invitation-resources` ejemplo, se enumeran todos los recursos que se encuentran en el recurso compartido asociado a la invitación especificada.

```
aws ram list-pending-invitation-resources \
  --resource-share-invitation-arn arn:aws:ram:us-west-2:123456789012:resource-
  share-invitation/1e3477be-4a95-46b4-bbe0-c4001EXAMPLE
```

Salida:

```
{
  "resources": [
    {
      "arn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-04a555b0e6EXAMPLE",
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7be8694e-095c-41ca-9ce8-7be4aEXAMPLE",
      "creationTime": 1634676051.269,
      "lastUpdatedTime": 1634676052.07,
      "status": "AVAILABLE",
      "type": "ec2:Subnet"
    },
    {
      "arn": "arn:aws:license-manager:us-west-2:123456789012:license-
configuration:lic-36be0485f5ae379cc74cf8e92EXAMPLE",
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
      "creationTime": 1624912434.431,
      "lastUpdatedTime": 1624912434.431,
      "status": "AVAILABLE",
      "type": "license-manager:LicenseConfiguration"
    }
  ]
}
```

- Para API obtener más información, consulte [ListPendingInvitationResources](#) la Referencia de AWS CLI comandos.

## list-permissions

En el siguiente ejemplo de código, se muestra cómo usar `list-permissions`.

## AWS CLI

Para ver una lista de los permisos RAM gestionados disponibles

En el siguiente `list-permissions` ejemplo, se enumeran todos los permisos RAM administrados disponibles únicamente para el tipo de recurso de la base de datos AWS Glue.

```
aws ram list-permissions \  
  --resource-type glue:Database
```

Salida:

```
{  
  "permissions": [  
    {  
      "arn": "arn:aws:ram::aws:permission/  
AWSRAMDefaultPermissionGlueDatabase",  
      "version": "1",  
      "defaultVersion": true,  
      "name": "AWSRAMDefaultPermissionGlueDatabase",  
      "resourceType": "glue:Database",  
      "creationTime": 1592007820.935,  
      "lastUpdatedTime": 1592007820.935,  
      "isResourceTypeDefault": true  
    },  
    {  
      "arn": "arn:aws:ram::aws:permission/  
AWSRAMPermissionGlueAllTablesReadWriteForDatabase",  
      "version": "2",  
      "defaultVersion": true,  
      "name": "AWSRAMPermissionGlueAllTablesReadWriteForDatabase",  
      "resourceType": "glue:Database",  
      "creationTime": 1624912413.323,  
      "lastUpdatedTime": 1624912413.323,  
      "isResourceTypeDefault": false  
    },  
    {  
      "arn": "arn:aws:ram::aws:permission/  
AWSRAMPermissionGlueDatabaseReadWrite",  
      "version": "2",  
      "defaultVersion": true,  
      "name": "AWSRAMPermissionGlueDatabaseReadWrite",  
      "resourceType": "glue:Database",
```

```

        "creationTime": 1624912417.4,
        "lastUpdatedTime": 1624912417.4,
        "isResourceTypeDefault": false
    },
    {
        "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionGlueTableReadWriteForDatabase",
        "version": "2",
        "defaultVersion": true,
        "name": "AWSRAMPermissionGlueTableReadWriteForDatabase",
        "resourceType": "glue:Database",
        "creationTime": 1624912434.431,
        "lastUpdatedTime": 1624912434.431,
        "isResourceTypeDefault": false
    }
]
}

```

En el siguiente `list-permissions` ejemplo, se muestran los permisos RAM administrados disponibles para todos los tipos de recursos.

```
aws ram list-permissions
```

Salida:

```

{
  "permissions": [
    {
      "arn": "arn:aws:ram::aws:permission/
AWSRAMBlankEndEntityCertificateAPICSRPassthroughIssuanceCertificateAuthority",
      "version": "1",
      "defaultVersion": true,
      "name":
"AWSRAMBlankEndEntityCertificateAPICSRPassthroughIssuanceCertificateAuthority",
      "resourceType": "acm-pca:CertificateAuthority",
      "creationTime": 1623264861.085,
      "lastUpdatedTime": 1623264861.085,
      "isResourceTypeDefault": false
    },
    {
      "arn": "arn:aws:ram::aws:permission/AWSRAMDefaultPermissionAppMesh",
      "version": "1",
      "defaultVersion": true,

```

```

        "name": "AWSRAMDefaultPermissionAppMesh",
        "resourceType": "appmesh:Mesh",
        "creationTime": 1589307188.584,
        "lastUpdatedTime": 1589307188.584,
        "isResourceTypeDefault": true
    },
    ...TRUNCATED FOR BREVITY...
    {
        "arn": "arn:aws:ram::aws:permission/
AWSRAMSubordinateCACertificatePathLen0IssuanceCertificateAuthority",
        "version": "1",
        "defaultVersion": true,
        "name":
    "AWSRAMSubordinateCACertificatePathLen0IssuanceCertificateAuthority",
        "resourceType": "acm-pca:CertificateAuthority",
        "creationTime": 1623264876.75,
        "lastUpdatedTime": 1623264876.75,
        "isResourceTypeDefault": false
    }
]
}

```

- Para API obtener más información, consulte [ListPermissions](#) la Referencia de AWS CLI comandos.

## list-principals

En el siguiente ejemplo de código, se muestra cómo usar `list-principals`.

### AWS CLI

Para enumerar los directores con acceso a un recurso

En el siguiente `list-principals` ejemplo, se muestra una lista de los principales que pueden acceder a los recursos del tipo especificado a través de cualquier recurso compartido.

```
aws ram list-principals \
  --resource-type ec2:Subnet
```

Salida:

```
{
```

```

    "principals": [
      {
        "id": "arn:aws:organizations::123456789012:ou/o-gx7EXAMPLE/ou-29c5-
zEXAMPLE",
        "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
        "creationTime": 1565298209.737,
        "lastUpdatedTime": 1565298211.019,
        "external": false
      }
    ]
  }

```

- Para API obtener más información, consulte [ListPrincipals](#) la Referencia de AWS CLI comandos.

## list-resource-share-permissions

En el siguiente ejemplo de código, se muestra cómo usar `list-resource-share-permissions`.

### AWS CLI

Para ver una lista de todos los permisos RAM gestionados actualmente asociados a un recurso compartido

En el siguiente `list-resource-share-permissions` ejemplo, se enumeran todos los permisos RAM administrados que están asociados al recurso compartido especificado.

```

aws ram list-resource-share-permissions \
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE

```

Salida:

```

{
  "permissions": [
    {
      "arn": "arn:aws:ram::aws:permission/
AWSRAMDefaultPermissionLicenseConfiguration",
      "version": "1",
      "resourceType": "license-manager:LicenseConfiguration",
      "status": "ASSOCIATED",

```

```

        "lastUpdatedTime": 1632342984.234
      },
      {
        "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionGlueDatabaseReadWrite",
        "version": "2",
        "resourceType": "glue:Database",
        "status": "ASSOCIATED",
        "lastUpdatedTime": 1632512462.297
      }
    ]
  }
}

```

- Para API obtener más información, consulte [ListResourceSharePermissions](#) la Referencia de AWS CLI comandos.

## list-resource-types

En el siguiente ejemplo de código, se muestra cómo usar `list-resource-types`.

### AWS CLI

Para enumerar los tipos de recursos compatibles con AWS RAM

En el siguiente `list-resource-types` ejemplo se enumeran todos los tipos de recursos que admite actualmente AWS RAM.

```
aws ram list-resource-types
```

Salida:

```

{
  "resourceTypes": [
    {
      "resourceType": "route53resolver:FirewallRuleGroup",
      "serviceName": "route53resolver"
    },
    {
      "resourceType": "ec2:LocalGatewayRouteTable",
      "serviceName": "ec2"
    },
    ...OUTPUT TRUNCATED FOR BREVITY...
  ]
}

```

```

    {
      "resourceType": "ec2:Subnet",
      "serviceName": "ec2"
    },
    {
      "resourceType": "ec2:TransitGatewayMulticastDomain",
      "serviceName": "ec2"
    }
  ]
}

```

- Para API obtener más información, consulte [ListResourceTypes](#) la Referencia de AWS CLI comandos.

## list-resources

En el siguiente ejemplo de código, se muestra cómo usar `list-resources`.

### AWS CLI

Para enumerar los recursos asociados a un recurso compartido

En el siguiente `list-resources` ejemplo, se enumeran todos los recursos del recurso compartido especificado que son del tipo de recurso especificado.

```

aws ram list-resources \
  --resource-type ec2:Subnet \
  --resource-owner SELF \
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE

```

Salida:

```

{
  "resources": [
    {
      "arn": "aarn:aws:ec2:us-west-2:123456789012:subnet/subnet-0250c25a1f4e15235",
      "type": "ec2:Subnet",
      "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
      "creationTime": 1565301545.023,
    }
  ]
}

```

```
        "lastUpdatedTime": 1565301545.947
      }
    ]
  }
```

- Para API obtener más información, consulte [ListResources](#) la Referencia de AWS CLI comandos.

## **promote-resource-share-created-from-policy**

En el siguiente ejemplo de código, se muestra cómo usar `promote-resource-share-created-from-policy`.

### AWS CLI

Para promover un uso compartido de recursos basado en políticas de recursos hasta alcanzar una funcionalidad completa en AWS RAM

En el siguiente `promote-resource-share-created-from-policy` ejemplo, se toma un recurso compartido que se creó implícitamente al adjuntar una política basada en recursos y se convierte en un recurso compartido totalmente funcional con la AWS RAM consola y sus operaciones. CLI API

```
aws ram promote-resource-share-created-from-policy \
  --resource-share-arn arn:aws:ram:us-east-1:123456789012:resource-
share/91fa8429-2d06-4032-909a-90909EXAMPLE
```

Salida:

```
{
  "returnValue": true
}
```

- Para obtener API más información, consulte la Referencia de comandos [PromoteResourceShareCreatedFromPolicy](#).AWS CLI

## **reject-resource-share-invitation**

En el siguiente ejemplo de código, se muestra cómo usar `reject-resource-share-invitation`.



## AWS CLI

Para rechazar una invitación a compartir recursos

En el siguiente `reject-resource-share-invitation` ejemplo, se rechaza la invitación a compartir recursos especificada.

```
aws ram reject-resource-share-invitation \  
  --resource-share-invitation-arn arn:aws:ram:us-west-2:111111111111:resource-  
share-invitation/32b639f0-14b8-7e8f-55ea-e6117EXAMPLE
```

Salida:

```
"resourceShareInvitations": [  
  {  
    "resourceShareInvitationArn": "arn:aws:ram:us-west2-1:111111111111:resource-  
share-invitation/32b639f0-14b8-7e8f-55ea-e6117EXAMPLE",  
    "resourceShareName": "project-resource-share",  
    "resourceShareArn": "arn:aws:ram:us-west-2:111111111111:resource-share/  
fcb639f0-1449-4744-35bc-a983fEXAMPLE",  
    "senderAccountId": "111111111111",  
    "receiverAccountId": "222222222222",  
    "invitationTimestamp": 1565319592.463,  
    "status": "REJECTED"  
  }  
]
```

- Para API obtener más información, consulte [RejectResourceShareInvitation](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

## AWS CLI

Para añadir etiquetas a un recurso compartido

En el siguiente `tag-resource` ejemplo, se agrega una clave de etiqueta `project` y un valor asociado `lima` al recurso compartido especificado.

```
aws ram tag-resource \  
  --tags key=project,value=Lima \  
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-  
b505-7e2a-420d-6f5d3EXAMPLE
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar etiquetas de un recurso compartido

En el siguiente `untag-resource` ejemplo, se eliminan la clave de `project` etiqueta y el valor asociado del recurso compartido especificado.

```
aws ram untag-resource \  
  --tag-keys project \  
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-  
b505-7e2a-420d-6f5d3EXAMPLE
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-resource-share

En el siguiente ejemplo de código, se muestra cómo usar `update-resource-share`.

### AWS CLI

Para actualizar un recurso compartido

En el siguiente `update-resource-share` ejemplo, se cambia el recurso compartido especificado para permitir el uso de directores externos que no estén en una AWS organización.

```
aws ram update-resource-share \  
  --allow-external-principals \  
  --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-  
b505-7e2a-420d-6f5d3EXAMPLE
```

Salida:

```
{  
  "resourceShare": {  
    "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-  
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",  
    "name": "my-resource-share",  
    "owningAccountId": "123456789012",  
    "allowExternalPrincipals": true,  
    "status": "ACTIVE",  
    "creationTime": 1565295733.282,  
    "lastUpdatedTime": 1565303080.023  
  }  
}
```

- Para API obtener más información, consulte [UpdateResourceShare](#) la Referencia de AWS CLI comandos.

## Ejemplos de Resource Explorer que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el Explorador AWS Command Line Interface de recursos.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **associate-default-view**

En el siguiente ejemplo de código, se muestra cómo usar `associate-default-view`.

#### AWS CLI

Para configurar una vista del Explorador de recursos como predeterminada para su AWS región

En el siguiente `associate-default-view` ejemplo, se establece una vista, tal como lo especificaARN, como la vista predeterminada de la AWS región en la que se llama a la operación.

```
aws resource-explorer-2 associate-default-view \  
  --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-View/  
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Salida:

```
{  
  "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-  
View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"  
}
```

Para obtener más información, consulte [Configurar una vista predeterminada en una AWS región en la](#) Guía del usuario AWS de Resource Explorer.

- Para API obtener más información, consulte [AssociateDefaultView](#) la Referencia de AWS CLI comandos.

### **batch-get-view**

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-view`.

#### AWS CLI

Para recuperar detalles sobre varias vistas del Explorador de recursos

En el siguiente `batch-get-view` ejemplo, se muestran los detalles de dos vistas especificadas por susARNs. Utilice espacios para separar el múltiplo del ARNs parámetro `--view-arn`.

```
aws resource-explorer-2 batch-get-view \  
  --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111 arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-View/EXAMPLE9-01ab-cdef-fedc-EXAMPLE11111
```

```
--view-arns arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222, \
            arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Salida:

```
{
  "Views": [
    {
      "Filters": {
        "FilterString": "service:ec2"
      },
      "IncludedProperties": [
        {
          "Name": "tags"
        }
      ],
      "LastUpdatedAt": "2022-07-13T21:33:45.249000+00:00",
      "Owner": "123456789012",
      "Scope": "arn:aws:iam::123456789012:root",
      "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222"
    },
    {
      "Filters": {
        "FilterString": ""
      },
      "IncludedProperties": [
        {
          "Name": "tags"
        }
      ],
      "LastUpdatedAt": "2022-07-13T20:34:11.314000+00:00",
      "Owner": "123456789012",
      "Scope": "arn:aws:iam::123456789012:root",
      "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
    }
  ]
  "Errors": []
}
```

Para obtener más información sobre las vistas, consulte [Acerca de las vistas del Explorador de recursos en la Guía del usuario del Explorador](#) de AWS recursos.

- Para API obtener más información, consulte [BatchGetView](#) la Referencia de AWS CLI comandos.

## create-index

En el siguiente ejemplo de código, se muestra cómo usar `create-index`.

### AWS CLI

Para activar el explorador de recursos en una AWS región mediante la creación de un índice

El siguiente `create-index` ejemplo crea un índice local en la AWS región en la que se llama a la operación. Genera AWS CLI automáticamente un valor de `client-token` parámetro aleatorio y lo incluye en la llamada a AWS si no se especifica ningún valor.

```
aws resource-explorer-2 create-index \  
  --region us-east-1
```

Salida:

```
{  
  "Arn": "arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-  
cdef-fedc-EXAMPLE22222c",  
  "CreatedAt": "2022-11-01T20:00:59.149Z",  
  "State": "CREATING"  
}
```

Tras crear un índice local, puede convertirlo en el índice agregador de la cuenta ejecutando el [update-index-type](#) comando.

Para obtener más información, consulte [Activar el explorador de recursos en una AWS región para indexar los recursos](#) en la Guía del usuario del explorador de AWS recursos.

- Para API obtener más información, consulte [CreateIndex](#) la Referencia de AWS CLI comandos.

## create-view

En el siguiente ejemplo de código, se muestra cómo usar `create-view`.

## AWS CLI

Ejemplo 1: Para crear una vista sin filtrar del índice de una AWS región

El siguiente `create-view` ejemplo crea una vista en la AWS región especificada que devuelve todos los resultados de la región sin ningún tipo de filtrado. La vista incluye el campo Etiquetas opcional en los resultados devueltos. Como esta vista se crea en la región que contiene el índice del agregador, puede incluir los resultados de todas las regiones de la cuenta que contienen un índice del explorador de recursos.

```
aws resource-explorer-2 create-view \  
  --view-name My-Main-View \  
  --included-properties Name=tags \  
  --region us-east-1
```

Salida:

```
{  
  "View": {  
    "Filters": {  
      "FilterString": ""  
    },  
    "IncludedProperties": [  
      {  
        "Name": "tags"  
      }  
    ],  
    "LastUpdatedAt": "2022-07-13T20:34:11.314000+00:00",  
    "Owner": "123456789012",  
    "Scope": "arn:aws:iam::123456789012:root",  
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"  
  }  
}
```

Ejemplo 2: Para crear una vista que devuelva solo los recursos asociados a Amazon EC2

A continuación, se `create-view` crea una vista en AWS la región `us-east-1` que muestra solo los recursos de la región que están asociados al EC2 servicio de Amazon. La vista incluye el Tags campo opcional de los resultados devueltos. Como esta vista se crea en la región que contiene el índice del agregador, puede incluir los resultados de todas las regiones de la cuenta que contienen un índice del explorador de recursos.

```
aws resource-explorer-2 create-view \  
  --view-name My-EC2-Only-View \  
  --included-properties Name=tags \  
  --filters FilterString="service:ec2" \  
  --region us-east-1
```

Salida:

```
{  
  "View": {  
    "Filters": {  
      "FilterString": "service:ec2"  
    },  
    "IncludedProperties": [  
      {  
        "Name": "tags"  
      }  
    ],  
    "LastUpdatedAt": "2022-07-13T21:35:09.059Z",  
    "Owner": "123456789012",  
    "Scope": "arn:aws:iam::123456789012:root",  
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-  
Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222"  
  }  
}
```

Para obtener más información, consulte [Crear vistas para la búsqueda](#) en la Guía del usuario AWS de Resource Explorer.

- Para API obtener más información, consulte [CreateView](#) la Referencia de AWS CLI comandos.

## delete-index

En el siguiente ejemplo de código, se muestra cómo usar `delete-index`.

### AWS CLI

Para desactivar el Explorador de recursos en una AWS región mediante la eliminación de su índice

En el siguiente `delete-index` ejemplo, se elimina el índice del explorador de recursos especificado en la AWS región en la que se realiza la solicitud.



```
aws resource-explorer-2 delete-index \  
  --arn arn:aws:resource-explorer-2:us-west-2:123456789012:index/EXAMPLE8-90ab-  
cdef-fedc-EXAMPLE22222 \  
  --region us-west-2
```

Salida:

```
{  
  "Arn": "arn:aws:resource-explorer-2:us-west-2:123456789012:index/EXAMPLE8-90ab-  
cdef-fedc-EXAMPLE22222",  
  "State": "DELETING"  
}
```

Para obtener más información sobre la eliminación de un índice, consulte [Desactivar el Explorador de AWS recursos en una AWS región](#) en la Guía del usuario del Explorador de AWS recursos.

- Para API obtener más información, consulte [DeleteIndex](#) la Referencia de AWS CLI comandos.

## delete-view

En el siguiente ejemplo de código, se muestra cómo usar delete-view.

### AWS CLI

Para eliminar una vista del explorador de recursos

En el siguiente delete-view ejemplo, se elimina una vista especificada por suARN.

```
aws resource-explorer-2 delete-view \  
  --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-  
View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Salida:

```
{  
  "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-  
View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"  
}
```

Para obtener más información, consulte [Eliminar vistas](#) en la Guía del usuario AWS de Resource Explorer.

- Para API obtener más información, consulte [DeleteView](#) la Referencia de AWS CLI comandos.

## **disassociate-default-view**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-default-view`.

### AWS CLI

Para eliminar la vista predeterminada del Explorador de recursos de una AWS región

Lo siguiente `disassociate-default-view` elimina la vista predeterminada del explorador de recursos de la AWS región en la que se llama a la operación. Tras realizar esta operación, todas las operaciones de búsqueda de la región deben especificar explícitamente una vista; de lo contrario, la operación fallará.

```
aws resource-explorer-2 disassociate-default-view
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configurar una vista predeterminada en una AWS región en la](#) Guía del usuario AWS de Resource Explorer.

- Para API obtener más información, consulte [DisassociateDefaultView](#) la Referencia de AWS CLI comandos.

## **get-default-view**

En el siguiente ejemplo de código, se muestra cómo usar `get-default-view`.

### AWS CLI

Para recuperar la vista del Explorador de recursos, que es la vista predeterminada para su AWS región

En el siguiente `get-default-view` ejemplo, se recupera la vista predeterminada ARN de la AWS región en la que se llama a la operación.

```
aws resource-explorer-2 get-default-view
```

Salida:

```
{
  "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/default-
view/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
}
```

Para obtener más información, consulte [Configurar una vista predeterminada en una AWS región en la](#) Guía del usuario de AWS Resource Explorer.

- Para API obtener más información, consulte [GetDefaultView](#) la Referencia de AWS CLI comandos.

## get-index

En el siguiente ejemplo de código, se muestra cómo usar `get-index`.

### AWS CLI

Ejemplo 1: Para recuperar los detalles de un índice agregador de Resource Explorer

En el siguiente `get-index` ejemplo, se muestran los detalles del índice del explorador de recursos en la AWS región especificada. Como la región especificada contiene el índice agregador de la cuenta, el resultado muestra las regiones que replican los datos en el índice de esta región.

```
aws resource-explorer-2 get-index \
  --region us-east-1
```

Salida:

```
{
  "Arn": "arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-
cdef-fedc-EXAMPLE11111",
  "CreatedAt": "2022-07-12T18:59:10.503000+00:00",
  "LastUpdatedAt": "2022-07-13T18:41:58.799000+00:00",
  "ReplicatingFrom": [
    "ap-south-1",
    "us-west-2"
  ],
  "State": "ACTIVE",
  "Tags": {},
}
```

```
"Type": "AGGREGATOR"
}
```

Ejemplo 2: Para recuperar los detalles de un índice local de Resource Explorer

En el siguiente `get-index` ejemplo, se muestran los detalles del índice del Explorador de recursos en la AWS región especificada. Como la región especificada contiene un índice local, el resultado muestra la región en la que se replican los datos del índice de esta región.

```
aws resource-explorer-2 get-index \
  --region us-west-2
```

Salida:

```
{
  "Arn": "arn:aws:resource-explorer-2:us-west-2:123456789012:index/EXAMPLE8-90ab-
cdef-fedc-EXAMPLE22222",
  "CreatedAt": "2022-07-12T18:59:10.503000+00:00",
  "LastUpdatedAt": "2022-07-13T18:41:58.799000+00:00",
  "ReplicatingTo": [
    "us-west-2"
  ],
  "State": "ACTIVE",
  "Tags": {},
  "Type": "LOCAL"
}
```

Para obtener más información sobre los índices, consulte [Comprobar qué AWS regiones tienen activado el Explorador de recursos en](#) la Guía del usuario del Explorador de AWS recursos.

- Para API obtener más información, consulte [GetIndex](#) la Referencia de AWS CLI comandos.

## get-view

En el siguiente ejemplo de código, se muestra cómo usar `get-view`.

### AWS CLI

Para recuperar detalles sobre una vista del explorador de recursos

En el siguiente `get-view` ejemplo, se muestran los detalles de una vista especificada por suARN.

```
aws resource-explorer-2 get-view \
  --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Salida:

```
{
  "Tags" : {},
  "View" : {
    "Filters" : {
      "FilterString" : "service:ec2"
    },
    "IncludedProperties" : [
      {
        "Name" : "tags"
      }
    ],
    "LastUpdatedAt" : "2022-07-13T21:33:45.249Z",
    "Owner" : "123456789012",
    "Scope" : "arn:aws:iam::123456789012:root",
    "ViewArn" : "arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
  }
}
```

Para obtener más información sobre las vistas, consulte [Acerca de las vistas del Explorador de recursos](#) en la Guía del usuario del Explorador de AWS recursos.

- Para API obtener más información, consulte [GetView](#) la Referencia de AWS CLI comandos.

## list-indexes

En el siguiente ejemplo de código, se muestra cómo usar `list-indexes`.

### AWS CLI

Para enumerar las AWS regiones en las que Resource Explorer tiene índices

El siguiente `list-indexes` ejemplo muestra los índices de todas las regiones en las que Resource Explorer tiene un índice. La respuesta especifica el tipo de cada índice, su AWS región y suARN.

```
aws resource-explorer-2 list-indexes
```

Salida:

```
{
  "Indexes": [
    {
      "Arn": "arn:aws:resource-explorer-2:us-west-2:123456789012:index/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111",
      "Region": "us-west-2",
      "Type": "AGGREGATOR"
    },
    {
      "Arn": "arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222",
      "Region": "us-east-1",
      "Type": "LOCAL"
    },
    {
      "Arn": "arn:aws:resource-explorer-2:us-east-2:123456789012:index/EXAMPLE8-90ab-cdef-fedc-EXAMPLE33333",
      "Region": "us-east-2",
      "Type": "LOCAL"
    },
    {
      "Arn": "arn:aws:resource-explorer-2:us-west-1:123456789012:index/EXAMPLE8-90ab-cdef-fedc-EXAMPLE44444",
      "Region": "us-west-1",
      "Type": "LOCAL"
    }
  ]
}
```

Para obtener más información sobre los índices, consulte [Comprobar qué AWS regiones tienen activado el Explorador de recursos en](#) la Guía del usuario del Explorador de AWS recursos.

- Para API obtener más información, consulte [ListIndexes](#) la Referencia de AWS CLI comandos.

## list-supported-resource-types

En el siguiente ejemplo de código, se muestra cómo usar `list-supported-resource-types`.

## AWS CLI

Para enumerar las AWS regiones en las que Resource Explorer tiene índices

El siguiente `list-supported-resource-types` ejemplo muestra todos los tipos de recursos que actualmente admite &AREXlong;. La respuesta de ejemplo incluye un `NextToken` valor que indica que hay más resultados disponibles para recuperar con llamadas adicionales.

```
aws resource-explorer-2 list-supported-resource-types \  
  --max-items 10
```

Salida:

```
{  
  "ResourceTypes": [  
    {  
      "ResourceType": "cloudfront:cache-policy",  
      "Service": "cloudfront"  
    },  
    {  
      "ResourceType": "cloudfront:distribution",  
      "Service": "cloudfront"  
    },  
    {  
      "ResourceType": "cloudfront:function",  
      "Service": "cloudfront"  
    },  
    {  
      "ResourceType": "cloudfront:origin-access-identity",  
      "Service": "cloudfront"  
    },  
    {  
      "ResourceType": "cloudfront:origin-request-policy",  
      "Service": "cloudfront"  
    },  
    {  
      "ResourceType": "cloudfront:realtime-log-config",  
      "Service": "cloudfront"  
    },  
    {  
      "ResourceType": "cloudfront:response-headers-policy",  
      "Service": "cloudfront"  
    },  
  ],  
}
```

```

    {
      "ResourceType": "cloudwatch:alarm",
      "Service": "cloudwatch"
    },
    {
      "ResourceType": "cloudwatch:dashboard",
      "Service": "cloudwatch"
    },
    {
      "ResourceType": "cloudwatch:insight-rule",
      "Service": "cloudwatch"
    }
  ],
  "NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxMH0="
}

```

Para obtener la siguiente parte del resultado, vuelva a llamar a la operación y pase el valor de `NextToken` respuesta de la llamada anterior como valor de `--starting-token`. Repita el procedimiento hasta `NextToken` que no aparezca en la respuesta.

```

aws resource-explorer-2 list-supported-resource-types \
  --max-items 10 \
  --starting-
token eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxMH0=

```

Salida:

```

{
  "ResourceTypes": [
    {
      "ResourceType": "cloudwatch:metric-stream",
      "Service": "cloudwatch"
    },
    {
      "ResourceType": "dynamodb:table",
      "Service": "dynamodb"
    },
    {
      "ResourceType": "ec2:capacity-reservation",
      "Service": "ec2"
    },
    {

```



```
    "ResourceType": "ec2:capacity-reservation-fleet",
    "Service": "ec2"
  },
  {
    "ResourceType": "ec2:client-vpn-endpoint",
    "Service": "ec2"
  },
  {
    "ResourceType": "ec2:customer-gateway",
    "Service": "ec2"
  },
  {
    "ResourceType": "ec2:dedicated-host",
    "Service": "ec2"
  },
  {
    "ResourceType": "ec2:dhcp-options",
    "Service": "ec2"
  },
  {
    "ResourceType": "ec2:egress-only-internet-gateway",
    "Service": "ec2"
  },
  {
    "ResourceType": "ec2:elastic-gpu",
    "Service": "ec2"
  }
],
"NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyMH0="
}
```

Para obtener más información sobre los índices, consulte [Comprobar qué AWS regiones tienen activado el Explorador de recursos en](#) la Guía del usuario del Explorador de AWS recursos.

- Para API obtener más información, consulte [ListSupportedResourceTypes](#) la Referencia de AWS CLI comandos.

## **list-tags-for-resource**

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

## AWS CLI

Para enumerar las etiquetas adjuntas a una vista o índice del Explorador de recursos

En el siguiente `list-tags-for-resource` ejemplo, se muestran los pares de etiquetas, claves y valores adjuntos a la vista con los especificadosARN. Debe llamar a la operación desde la AWS región que contiene el recurso.

```
aws resource-explorer-2 list-tags-for-resource \
  --resource-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Salida:

```
{
  "Tags": {
    "application": "MainCorpApp",
    "department": "1234"
  }
}
```

Para obtener más información sobre el etiquetado de vistas, consulte [Etiquetar vistas para el control de acceso](#) en la Guía del usuario de AWS Resource Explorer.

- Para API obtener más información, consulte la Referencia [ListTagsForResource](#) de AWS CLI comandos.

## **list-views**

En el siguiente ejemplo de código, se muestra cómo usar `list-views`.

## AWS CLI

Para ver una lista de las vistas del explorador de recursos disponibles en una AWS región

En el siguiente `list-views` ejemplo, se enumeran todas las vistas disponibles en la región en la que se invoca la operación.

```
aws resource-explorer-2 list-views
```

Salida:

```
{
  "Views": [
    "arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111",
    "arn:aws:resource-explorer-2:us-east-1:123456789012:view/Default-All-Resources-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222",
    "arn:aws:resource-explorer-2:us-east-1:123456789012:view/Production-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE33333"
  ]
}
```

Para obtener más información sobre las vistas, consulte [Acerca de las vistas del Explorador de recursos](#) en la Guía del usuario del Explorador de AWS recursos.

- Para API obtener más información, consulte [ListViews](#) la Referencia de AWS CLI comandos.

## search

En el siguiente ejemplo de código, se muestra cómo usar search.

### AWS CLI

Ejemplo 1: Para buscar utilizando la vista por defecto

En el siguiente search ejemplo, se muestran todos los recursos de los especificados que están asociados al servicio. La búsqueda utiliza la vista predeterminada de la región. La respuesta de ejemplo incluye un NextToken valor que indica que hay más resultados disponibles para recuperar con llamadas adicionales.

```
aws resource-explorer-2 search \
  --query-string "service:iam"
```

Salida:

```
{
  "Count": {
    "Complete": true,
    "TotalResources": 55
  },
  "NextToken":
  "AG9V0EF1KLEXAMPLE0hJHVwo5chEXAMPLER5XiEpNrgsEXAMPLE...b0Cm0F0ryHEXAMPLE",
  "Resources": [{
```

```

    "Arn": "arn:aws:iam::123456789012:policy/service-role/Some-Policy-For-A-
Service-Role",
    "LastReportedAt": "2022-07-21T12:34:42Z",
    "OwningAccountId": "123456789012",
    "Properties": [],
    "Region": "global",
    "ResourceType": "iam:policy",
    "Service": "iam"
  }, {
    "Arn": "arn:aws:iam::123456789012:policy/service-role/Another-Policy-For-A-
Service-Role",
    "LastReportedAt": "2022-07-21T12:34:42Z",
    "OwningAccountId": "123456789012",
    "Properties": [],
    "Region": "global",
    "ResourceType": "iam:policy",
    "Service": "iam"
  }, {
    ... TRUNCATED FOR BREVITY ...
  }],
  "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/my-default-
view/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
}

```

## Ejemplo 2: Para buscar mediante una vista específica

El siguiente search ejemplo de búsqueda muestra todos los recursos («\*») de la AWS región especificada que están visibles a través de la vista especificada. Los resultados incluyen solo los recursos asociados EC2 a Amazon debido a los filtros adjuntos a la vista.

```

aws resource-explorer-2 search \
  -- query-string "*" \
  -- view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-view/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222

```

## Salida:

```

HTTP/1.1 200 OK
Date: Tue, 01 Nov 2022 20:00:59 GMT
Content-Type: application/json
Content-Length: <PayloadSizeBytes>

{

```

```
"Count": {
  "Complete": true,
  "TotalResources": 67
},
"Resources": [{
  "Arn": "arn:aws:ec2:us-east-1:123456789012:network-acl/acl-1a2b3c4d",
  "LastReportedAt": "2022-07-21T18:52:02Z",
  "OwningAccountId": "123456789012",
  "Properties": [{
    "Data": [{
      "Key": "Department",
      "Value": "AppDevelopment"
    }, {
      "Key": "Environment",
      "Value": "Production"
    }
  ]],
  "LastReportedAt": "2021-11-15T14:48:29Z",
  "Name": "tags"
}],
"Region": "us-east-1",
"ResourceType": "ec2:network-acl",
"Service": "ec2"
}, {
  "Arn": "arn:aws:ec2:us-east-1:123456789012:subnet/subnet-1a2b3c4d",
  "LastReportedAt": "2022-07-21T21:22:23Z",
  "OwningAccountId": "123456789012",
  "Properties": [{
    "Data": [{
      "Key": "Department",
      "Value": "AppDevelopment"
    }, {
      "Key": "Environment",
      "Value": "Production"
    }
  ]],
  "LastReportedAt": "2021-07-29T19:02:39Z",
  "Name": "tags"
}],
"Region": "us-east-1",
"ResourceType": "ec2:subnet",
"Service": "ec2"
}, {
  "Arn": "arn:aws:ec2:us-east-1:123456789012:dhcp-options/dopt-1a2b3c4d",
  "LastReportedAt": "2022-07-21T06:08:53Z",
  "OwningAccountId": "123456789012",
```

```

    "Properties": [{
      "Data": [{
        "Key": "Department",
        "Value": "AppDevelopment"
      }, {
        "Key": "Environment",
        "Value": "Production"
      }],
      "LastReportedAt": "2021-11-15T15:11:05Z",
      "Name": "tags"
    }],
    "Region": "us-east-1",
    "ResourceType": "ec2:dhcptions",
    "Service": "ec2"
  }, {
    ... TRUNCATED FOR BREVITY ...
  }],
  "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-
view/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222"
}

```

Para obtener más información, consulte [Uso del Explorador de AWS recursos para buscar recursos](#) en la Guía del usuario del Explorador de AWS recursos.

- Para API obtener más información, consulte [Buscar](#) en la referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

### AWS CLI

Para etiquetar una vista del explorador de recursos

En el siguiente tag-resource ejemplo, se agrega la clave de etiqueta «entorno» con el valor «producción» a la vista con el valor especificadoARN.

```

aws resource-explorer-2 tag-resource \
  --resource-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View//  

EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111 \
  --tags environment=production

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar vistas para el control de acceso](#) en la Guía del usuario AWS de Resource Explorer.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar una etiqueta de una vista del Explorador de recursos

En el siguiente `untag-resource` ejemplo, se elimina cualquier etiqueta con el nombre clave «entorno» de la vista con el valor especificadoARN.

```
aws resource-explorer-2 untag-resource \  
  --resource-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View//  
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111 \  
  --tag-keys environment
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar vistas para el control de acceso](#) en la Guía del usuario AWS de Resource Explorer.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-index-type

En el siguiente ejemplo de código, se muestra cómo usar `update-index-type`.

### AWS CLI

Para cambiar el tipo de índice de Resource Explorer

En el siguiente `update-index-type` ejemplo, se convierte el índice especificado de un tipo `local aggregator` a otro para activar la posibilidad de buscar recursos en todas AWS las regiones de la cuenta. Debe enviar la solicitud a la AWS región que contiene el índice que desea actualizar.

```
aws resource-explorer-2 update-index-type \  
  --arn arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-  
cdef-fedc-EXAMPLE11111 \  
  --type aggregator \  
  --region us-east-1
```

Salida:

```
{  
  "Arn": "arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-  
cdef-fedc-EXAMPLE11111",  
  "LastUpdatedAt": "2022-07-13T18:41:58.799Z",  
  "State": "updating",  
  "Type": "aggregator"  
}
```

Para obtener más información sobre cómo cambiar el tipo de un índice, consulte [Activar la búsqueda entre regiones mediante la creación de un índice agregador](#) en la Guía del usuario de AWS Resource Explorer.

- Para API obtener más información, consulte la Referencia [UpdateIndexType](#) de AWS CLI comandos.

## update-view

En el siguiente ejemplo de código, se muestra cómo usar update-view.

### AWS CLI

Ejemplo 1: Para actualizar el IncludedProperties campo para una vista del explorador de recursos

En el siguiente update-view ejemplo, se actualiza la vista especificada agregándola `tags` a la opcional `IncludedProperties`. Tras ejecutar esta operación, las operaciones de búsqueda que utilizan esta vista incluyen información sobre las etiquetas adjuntas a los recursos que aparecen en los resultados.

```
aws resource-explorer-2 update-view \  
  --included-properties Name=tags \  
  --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View/  
EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222
```



**Salida:**

```
{
  "View": {
    "Filters": {
      "FilterString": ""
    },
    "IncludedProperties": [
      {
        "Name": "tags"
      }
    ],
    "LastUpdatedAt": "2022-07-19T17:41:21.710000+00:00",
    "Owner": "123456789012",
    "Scope": "arn:aws:iam::123456789012:root",
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
  }
}
```

**Ejemplo 2: Para actualizar los filtros adjuntos a una vista**

En el siguiente `update-view` ejemplo, se actualiza la vista especificada para utilizar un filtro que limite los resultados únicamente a los tipos de recursos asociados al EC2 servicio de Amazon.

```
aws resource-explorer-2 update-view \
  --filters FilterString="service:ec2" \
  --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222
```

**Salida:**

```
{
  "View": {
    "Filters": {
      "FilterString": "service:ec2"
    },
    "IncludedProperties": [],
    "LastUpdatedAt": "2022-07-19T17:41:21.710000+00:00",
    "Owner": "123456789012",
    "Scope": "arn:aws:iam::123456789012:root",
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222"
  }
}
```

```
}  
}
```

Para obtener más información sobre las vistas, consulte [Acerca de las vistas de Resource Explorer](#) en la Guía del usuario de AWS Resource Explorer.

- Para API obtener más información, consulte [UpdateView](#) la Referencia de AWS CLI comandos.

## Ejemplos de Resource Groups que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface with Resource Groups.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-group**

En el siguiente ejemplo de código, se muestra cómo usar `create-group`.

AWS CLI

Ejemplo 1: Para crear un grupo de recursos basado en etiquetas

En el siguiente `create-group` ejemplo, se crea un grupo de recursos de EC2 instancias de Amazon basado en etiquetas en la región actual. Se basa en una consulta de recursos que están etiquetados con la clave `Name` y el valor `WebServers`. El nombre del grupo es `tbq-WebServer`. La consulta se encuentra en un JSON archivo independiente que se pasa al comando.

```
aws resource-groups create-group \  
  --name tbq-WebServer \  
  --tags Key=Name,Value=WebServers \  
  --cli-input-file tbq-WebServer.json
```

```
--resource-query file://query.json
```

Contenidos de `query.json`:

```
{
  "Type": "TAG_FILTERS_1_0",
  "Query": "{\"ResourceTypeFilters\": [\"AWS::EC2::Instance\"], \"TagFilters\": [ { \"Key\": \"Name\", \"Values\": [\"WebServers\"] } ]}"
}
```

Salida:

```
{
  "Group": {
    "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer",
    "Name": "tbq-WebServer"
  },
  "ResourceQuery": {
    "Type": "TAG_FILTERS_1_0",
    "Query": "{\"ResourceTypeFilters\": [\"AWS::EC2::Instance\"], \"TagFilters\": [ { \"Key\": \"Name\", \"Values\": [\"WebServers\"] } ]}"
  }
}
```

Ejemplo 2: Para crear un grupo de CloudFormation recursos basado en pilas

En el siguiente `create-group` ejemplo, se crea un grupo de recursos AWS CloudFormation basado en pilas denominado `sampleCFNstackgroup`. La consulta incluye todos los recursos de la CloudFormation pila especificada que son compatibles con AWS Resource Groups.

```
aws resource-groups create-group \
  --name cbq-CFNstackgroup \
  --resource-query file://query.json
```

Contenidos de `query.json`:

```
{
  "Type": "CLOUDFORMATION_STACK_1_0",
  "Query": "{\"ResourceTypeFilters\": [\"AWS::AllSupported\"], \"StackIdentifier\": \"arn:aws:cloudformation:us-west-2:123456789012:stack/MyCFNStack/1415z9z0-z39z-11z8-97z5-500z212zz6fz\"}"
}
```

```
}

```

Salida:

```
{
  "Group": {
    "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/cbq-
CFNstackgroup",
    "Name": "cbq-CFNstackgroup"
  },
  "ResourceQuery": {
    "Type": "CLOUDFORMATION_STACK_1_0",
    "Query": "{\"ResourceTypeFilters\":[\"AWS::AllSupported\"],\"StackIdentifier
\": \"arn:aws:cloudformation:us-east-2:123456789012:stack/MyCFNStack/1415z9z0-
z39z-11z8-97z5-500z212zz6fz\"}"
  }
}
```

Para obtener más información, consulte [Crear grupos](#) en la Guía del usuario AWS de Resource Groups.

- Para API obtener más información, consulte [CreateGroup](#) la Referencia de AWS CLI comandos.

## delete-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-group`.

AWS CLI

Para actualizar la descripción de un grupo de recursos

En el siguiente `delete-group` ejemplo, se actualiza el grupo de recursos especificado.

```
aws resource-groups delete-group \
  --group-name tbq-WebServer
```

Salida:

```
{
  "Group": {
    "GroupArn": "arn:aws:resource-groups:us-west-2:1234567890:group/tbq-
WebServer",
```

```

    "Name": "tbq-WebServer"
  }
}

```

Para obtener más información, consulte [Eliminar grupos](#) en la Guía del usuario AWS de Resource Groups.

- Para API obtener más información, consulte [DeleteGroup](#) la Referencia de AWS CLI comandos.

## get-group-query

En el siguiente ejemplo de código, se muestra cómo usar `get-group-query`.

### AWS CLI

Para adjuntar la consulta a un grupo de recursos

En el siguiente `get-group-query` ejemplo, se muestra la consulta adjunta al grupo de recursos especificado.

```

aws resource-groups get-group-query \
  --group-name tbq-WebServer

```

Salida:

```

{
  "GroupQuery": {
    "GroupName": "tbq-WebServer",
    "ResourceQuery": {
      "Type": "TAG_FILTERS_1_0",
      "Query": "{\"ResourceTypeFilters\": [\"AWS::EC2::Instance\"], \"TagFilters\": [{\"Key\": \"Name\", \"Values\": [\"WebServers\"]}]}"
    }
  }
}

```

- Para API obtener más información, consulte [GetGroupQuery](#) la Referencia de AWS CLI comandos.

## get-group

En el siguiente ejemplo de código, se muestra cómo usar `get-group`.

## AWS CLI

Para obtener información sobre un grupo de recursos

En el siguiente `get-group` ejemplo, se muestran detalles sobre el grupo de recursos especificado. Para adjuntar la consulta al grupo, utilice `get-group-query`.

```
aws resource-groups get-group \  
  --group-name tbq-WebServer
```

Salida:

```
{  
  "Group": {  
    "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-  
WebServer",  
    "Name": "tbq-WebServer",  
    "Description": "A tag-based query resource group of WebServers."  
  }  
}
```

- Para API obtener más información, consulte [GetGroup](#) la Referencia de AWS CLI comandos.

## get-tags

En el siguiente ejemplo de código, se muestra cómo usar `get-tags`.

## AWS CLI

Para recuperar las etiquetas adjuntas a un grupo de recursos

En el siguiente `get-tags` ejemplo, se muestran los pares de etiquetas, clave y valores adjuntos al grupo de recursos especificado (el grupo en sí, no sus miembros).

```
aws resource-groups get-tags \  
  --arn arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer
```

Salida:

```
{  
  "Arn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer",
```

```

    "Tags": {
      "QueryType": "tags",
      "QueryResources": "ec2-instances"
    }
  }
}

```

- Para API obtener más información, consulte [GetTags](#) la Referencia de AWS CLI comandos.

## list-group-resources

En el siguiente ejemplo de código, se muestra cómo usar `list-group-resources`.

### AWS CLI

Para enumerar todos los recursos de un grupo de recursos

Ejemplo 1: En el siguiente `list-resource-groups` ejemplo se enumeran todos los recursos que forman parte del grupo de recursos especificado.

```

aws resource-groups list-group-resources \
  --group-name tbq-WebServer

```

Salida:

```

{
  "ResourceIdentifiers": [
    {
      "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/i-09f77fa38c12345ab",
      "ResourceType": "AWS::EC2::Instance"
    }
  ]
}

```

Ejemplo 2: El siguiente ejemplo muestra todos los recursos del grupo que también tienen un «tipo de recurso» de tipo 'AWS:: EC2 :Instancia' . :

```

aws resource-groups list-group-resources --group-name tbq- WebServer --filters name=Tipo de recurso, Values>:: :Instance AWS EC2

```

- API Para obtener más información [ListGroupResources](#), AWS CLI consulte la Referencia de comandos.

## list-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-groups`.

### AWS CLI

Para enumerar los grupos de recursos disponibles

En el siguiente `list-groups` ejemplo, se muestra una lista de todos los grupos de recursos.

```
aws resource-groups list-groups
```

Salida:

```
{
  "GroupIdentifiers": [
    {
      "GroupName": "tbq-WebServer",
      "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer3"
    },
    {
      "GroupName": "cbq-CFNStackQuery",
      "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/cbq-CFNStackQuery"
    }
  ],
  "Groups": [
    {
      "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer",
      "Name": "tbq-WebServer"
    },
    {
      "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/cbq-CFNStackQuery",
      "Name": "cbq-CFNStackQuery"
    }
  ]
}
```

- Para API obtener más información, consulte [ListGroups](#) la Referencia de AWS CLI comandos.



## list-resource-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-resource-groups`.

### AWS CLI

Para enumerar todos los recursos de un grupo de recursos

En el siguiente `list-resource-groups` ejemplo, se enumeran todos los recursos que forman parte del grupo de recursos especificado.

```
aws resource-groups list-group-resources \  
  --group-name tbq-WebServer
```

Salida:

```
{  
  "ResourceIdentifiers": [  
    {  
      "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/  
i-09f77fa38c12345ab",  
      "ResourceType": "AWS::EC2::Instance"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListResourceGroups](#) la Referencia de AWS CLI comandos.

## put-group-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-group-configuration`.

### AWS CLI

Para adjuntar una configuración de servicio a un grupo de recursos

Ejemplo 1: El siguiente `put-group-configuration` ejemplo especifica que el grupo de recursos debe contener solo las reservas de EC2 capacidad de Amazon para las instancias de las M5 familias C5 o.

```
aws resource-groups put-group-configuration \  
  --group MyTestGroup \  
  --configuration file://config.json
```

Contenidos de config.json:

```
[  
  {  
    "Type": "AWS::EC2::HostManagement",  
    "Parameters": [  
      {  
        "Name": "allowed-host-families",  
        "Values": [ "c5", "m5" ]  
      },  
      {  
        "Name": "any-host-based-license-configuration",  
        "Values": [ "true" ]  
      }  
    ]  
  },  
  {  
    "Type": "AWS::ResourceGroups::Generic",  
    "Parameters": [  
      {  
        "Name": "allowed-resource-types",  
        "Values": [ "AWS::EC2::Host" ]  
      },  
      {  
        "Name": "deletion-protection",  
        "Values": [ "UNLESS_EMPTY" ]  
      }  
    ]  
  }  
]
```

Este comando no genera ninguna salida si se realiza correctamente.

Para obtener más información, consulte [Configuraciones de servicios para grupos de recursos](#) en la Guía de API referencia de Resource Groups.

- Para API obtener más información, consulte [PutGroupConfiguration](#) la Referencia de AWS CLI comandos.

## search-resources

En el siguiente ejemplo de código, se muestra cómo usar `search-resources`.

### AWS CLI

Para buscar recursos que coincidan con una consulta

En el siguiente `search-resources` ejemplo, se recupera una lista de todos los AWS recursos que coinciden con la consulta especificada.

```
aws resource-groups search-resources \  
  --resource-query file://query.json
```

Contenidos de `query.json`:

```
{  
  "Type": "TAG_FILTERS_1_0",  
  "Query": "{\"ResourceTypeFilters\": [\"AWS::EC2::Instance\"], \"TagFilters\":  
  [{\"Key\": \"Patch Group\", \"Values\": [\"Dev\"]}]}"  
}
```

Salida:

```
{  
  "ResourceIdentifiers": [  
    {  
      "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/  
i-01a23bc45d67890ef",  
      "ResourceType": "AWS::EC2::Instance"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [SearchResources](#) la Referencia de AWS CLI comandos.

## tag

En el siguiente ejemplo de código, se muestra cómo usar `tag`.

## AWS CLI

Para adjuntar una etiqueta a un grupo de recursos

En el siguiente tag ejemplo, se adjuntan los pares de clave y valor de etiqueta especificados al grupo de recursos especificado (al grupo en sí, no a sus miembros).

```
aws resource-groups tag \  
  --tags QueryType=tags,QueryResources=ec2-instances \  
  --arn arn:aws:resource-groups:us-west-2:128716708097:group/tbq-WebServer
```

Salida:

```
{  
  "Arn": "arn:aws:resource-groups:us-west-2:128716708097:group/tbq-WebServer",  
  "Tags": {  
    "QueryType": "tags",  
    "QueryResources": "ec2-instances"  
  }  
}
```

Para obtener más información, consulte [Administrar etiquetas](#) en la Guía del usuario AWS de Resource Groups.

- Para API obtener más información, consulte [Etiqueta](#) en la referencia de AWS CLI comandos.

## untag

En el siguiente ejemplo de código, se muestra cómo usar untag.

## AWS CLI

Para eliminar etiquetas de un grupo de recursos

En el siguiente untags ejemplo, se quita cualquier etiqueta con la clave especificada del propio grupo de recursos, no de sus miembros.

```
aws resource-groups untag \  
  --arn arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer \  
  --keys QueryType
```

Salida:

```
{
  "Arn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer",
  "Keys": [
    "QueryType"
  ]
}
```

Para obtener más información, consulte [Administrar etiquetas](#) en la Guía del usuario AWS de Resource Groups.

- Para API obtener más información, consulte Eliminar [etiquetas](#) en la referencia de AWS CLI comandos.

## update-group-query

En el siguiente ejemplo de código, se muestra cómo usar `update-group-query`.

AWS CLI

Ejemplo 1: Para actualizar la consulta de un grupo de recursos basado en etiquetas

En el siguiente `update-group-query` ejemplo, se actualiza la consulta adjunta al grupo de recursos basado en etiquetas especificado.

```
aws resource-groups update-group-query \
  --group-name tbq-WebServer \
  --resource-query '{"Type":"TAG_FILTERS_1_0", "Query":{"ResourceTypeFilters\":[\
  "AWS::EC2::Instance"], "TagFilters":[{"Key":"Name", "Values":["WebServers"]}]}'
```

Salida:

```
{
  "Group": {
    "GroupArn": "arn:aws:resource-groups:us-east-2:123456789012:group/tbq-WebServer",
    "Name": "tbq-WebServer"
  },
  "ResourceQuery": {
```

```

    "Type": "TAG_FILTERS_1_0",
    "Query": "{\"ResourceTypeFilters\":[\"AWS::EC2::Instance\"],\"TagFilters\":
[{\\"Key\\":\\"Name\\", \\"Values\\":[\"WebServers\"]}]}"
  }
}

```

Para obtener más información, consulte [Actualizar grupos](#) en la Guía del usuario AWS de Resource Groups.

Ejemplo 2: Para actualizar la consulta de un grupo de recursos CloudFormation basado en pilas

En el siguiente `update-group-query` ejemplo, se actualiza la consulta adjunta al grupo de recursos basado en AWS CloudFormation pilas especificado.

```

aws resource-groups update-group-query \
  --group-name cbq-CFNstackgroup \
  --resource-query '{"Type": "CLOUDFORMATION_STACK_1_0", "Query":
  "\{"ResourceTypeFilters\":[\"AWS::AllSupported\"],\"StackIdentifier\":
  \\"arn:aws:cloudformation:us-west-2:123456789012:stack/MyCFNStack/1415z9z0-
  z39z-11z8-97z5-500z212zz6fz\"}"}'

```

Salida:

```

{
  "Group": {
    "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/cbq-
CFNstackgroup",
    "Name": "cbq-CFNstackgroup"
  },
  "ResourceQuery": {
    "Type": "CLOUDFORMATION_STACK_1_0",
    "Query": "\{"ResourceTypeFilters\":[\"AWS::AllSupported\"],\"StackIdentifier
\":\\"arn:aws:cloudformation:us-west-2:123456789012:stack/MyCFNStack/1415z9z0-
z39z-11z8-97z5-500z212zz6fz\"}"
  }
}

```

Para obtener más información, consulte [Actualizar grupos](#) en la Guía del usuario AWS de Resource Groups.

- Para API obtener más información, consulte [UpdateGroupQuery](#) la Referencia de AWS CLI comandos.

## update-group

En el siguiente ejemplo de código, se muestra cómo usar `update-group`.

### AWS CLI

Para actualizar la descripción de un grupo de recursos

En el siguiente `update-group` ejemplo, se actualiza la descripción del grupo de recursos especificado.

```
aws resource-groups update-group \  
  --group-name tbq-WebServer \  
  --description "Resource group for all web server resources."
```

Salida:

```
{  
  "Group": {  
    "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-  
WebServer",  
    "Name": "tbq-WebServer"  
    "Description": "Resource group for all web server resources."  
  }  
}
```

Para obtener más información, consulte [Actualizar grupos](#) en la Guía del usuario AWS de Resource Groups.

- Para API obtener más información, consulte [UpdateGroup](#) la Referencia de AWS CLI comandos.

## Resource Groups: API ejemplos de etiquetado mediante AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface with Resource Groups TaggingAPI.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### get-resources

En el siguiente ejemplo de código, se muestra cómo usar `get-resources`.

#### AWS CLI

Para obtener una lista de los recursos etiquetados

En el siguiente `get-resources` ejemplo, se muestra una lista de los recursos de la cuenta que están etiquetados con el nombre y el valor clave especificados.

```
aws resourcegroupstaggingapi get-resources \  
  --tag-filters Key=Environment,Values=Production \  
  --tags-per-page 100
```

Salida:

```
{  
  "ResourceTagMappingList": [  
    {  
      "ResourceARN": " arn:aws:inspector:us-west-2:123456789012:target/0-  
nvgVhaxX/template/0-7sbz2Kz0",  
      "Tags": [  
        {  
          "Key": "Environment",  
          "Value": "Production"  
        }  
      ]  
    }  
  ]  
}
```



Para obtener más información, consulte la APIReferencia [GetResources](#)de etiquetado de Resource Groups.

- Para API obtener más información, consulte [GetResources](#)la Referencia de AWS CLI comandos.

## get-tag-keys

En el siguiente ejemplo de código, se muestra cómo usar `get-tag-keys`.

### AWS CLI

Para obtener una lista de todas las claves de etiquetas

En el siguiente `get-tag-keys` ejemplo, se recupera la lista de todos los nombres de las claves de etiqueta que utilizan los recursos de la cuenta.

```
aws resourcegroupstaggingapi get-tag-keys
```

Salida:

```
{
  "TagKeys": [
    "Environment",
    "CostCenter",
    "Department"
  ]
}
```

Para obtener más información, consulte la APIReferencia [GetTagKeys](#)de etiquetado de Resource Groups.

- Para API obtener más información, consulte [GetTagKeys](#)la Referencia de AWS CLI comandos.

## get-tag-values

En el siguiente ejemplo de código, se muestra cómo usar `get-tag-values`.

### AWS CLI

Para obtener una lista de todos los valores de las etiquetas

En el `get-tag-values` ejemplo siguiente se muestran todos los valores utilizados para la clave especificada para todos los recursos del

```
aws resourcegroupstaggingapi get-tag-values \  
  --key=Environment
```

Salida:

```
{  
  "TagValues": [  
    "Alpha",  
    "Gamma",  
    "Production"  
  ]  
}
```

Para obtener más información, consulte la API Referencia [GetTagValues](#) de etiquetado de Resource Groups.

- Para API obtener más información, consulte [GetTagValues](#) la Referencia de AWS CLI comandos.

## tag-resources

En el siguiente ejemplo de código, se muestra cómo usar `tag-resources`.

AWS CLI

Para adjuntar una etiqueta a un recurso

El siguiente `tag-resources` ejemplo etiqueta el recurso especificado con un nombre y un valor clave.

```
aws resourcegroupstaggingapi tag-resources \  
  --resource-arn-list arn:aws:s3:::MyProductionBucket \  
  --tags Environment=Production, CostCenter=1234
```

Salida:

```
{
```

```
"FailedResourcesMap": {}  
}
```

Para obtener más información, consulte la APIReferencia [TagResources](#)de etiquetado de Resource Groups.

- Para API obtener más información, consulte [TagResources](#)la Referencia de AWS CLI comandos.

## untag-resources

En el siguiente ejemplo de código, se muestra cómo usar untag-resources.

### AWS CLI

Para eliminar una etiqueta de un recurso

En el siguiente untag-resources ejemplo, se quitan las claves de etiqueta especificadas y todos los valores asociados del recurso especificado.

```
aws resourcegroupstaggingapi untag-resources \  
  --resource-arn-list arn:aws:s3:::awsexamplebucket \  
  --tag-keys Environment CostCenter
```

Salida:

```
{  
  "FailedResourcesMap": {}  
}
```

Para obtener más información, consulte la APIReferencia [UntagResources](#)de etiquetado de Resource Groups.

- Para API obtener más información, consulte [UntagResources](#)la Referencia de AWS CLI comandos.

## AWS RoboMaker ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS RoboMaker.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **batch-describe-simulation-job**

En el siguiente ejemplo de código, se muestra cómo usar `batch-describe-simulation-job`.

#### AWS CLI

Para describir por lotes los trabajos de simulación

En el siguiente `batch-describe-simulation-job` ejemplo, se recuperan los detalles de los tres trabajos de simulación especificados.

Comando:

```
aws robomaker batch-describe-simulation-job \  
--job arn:aws:robomaker:us-west-2:111111111111:simulation-job/  
sim-66bbb3gpxm8x arn:aws:robomaker:us-west-2:111111111111:simulation-job/  
sim-p0cpdrrwng2n arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-  
g8h6tg1mblgw
```

Salida:

```
{  
  "jobs": [  
    {  
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/  
sim-66bbb3gpxm8x",  
      "status": "Completed",  
      "lastUpdatedAt": 1548959178.0,  
      "failureBehavior": "Continue",  
      "clientRequestToken": "6020408e-b05c-4310-9f13-4ed71c5221ed",
```

```
    "outputLocation": {
      "s3Bucket": "awsrobomakerobjecttracker-111111111-
bundlesbucket-2lk584kiq1oa",
      "s3Prefix": "output"
    },
    "maxJobDurationInSeconds": 3600,
    "simulationTimeMillis": 0,
    "iamRole": "arn:aws:iam::111111111111:role/
AWSRoboMakerObjectTracker-154895-SimulationJobRole-14D5ASA7PQE3A",
    "simulationApplications": [
      {
        "application": "arn:aws:robomaker:us-
west-2:111111111111:simulation-application/
AWSRoboMakerObjectTracker-1548959046124_NPvyfcatq/1548959170096",
        "applicationVersion": "$LATEST",
        "launchConfig": {
          "packageName": "object_tracker_simulation",
          "launchFile": "local_training.launch",
          "environmentVariables": {
            "MARKOV_PRESET_FILE": "object_tracker.py",
            "MODEL_S3_BUCKET": "awsrobomakerobjecttracker-111111111-
bundlesbucket-2lk584kiq1oa",
            "MODEL_S3_PREFIX": "model-store",
            "ROS_AWS_REGION": "us-west-2"
          }
        }
      }
    ],
    "tags": {},
    "vpcConfig": {
      "subnets": [
        "subnet-716dd52a",
        "subnet-43c22325",
        "subnet-3f526976"
      ],
      "securityGroups": [
        "sg-3fb40545"
      ],
      "vpcId": "vpc-99895eff",
      "assignPublicIp": true
    }
  },
  {
```

```
    "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-
p0cpdrrwng2n",
    "status": "Completed",
    "lastUpdatedAt": 1548168817.0,
    "failureBehavior": "Continue",
    "clientRequestToken": "e4a23e75-f9a7-411d-835f-21881c82c58b",
    "outputLocation": {
      "s3Bucket": "awsrobomakercloudwatch-111111111111-
bundlesbucket-14e5s9jvwtmv7",
      "s3Prefix": "output"
    },
    "maxJobDurationInSeconds": 3600,
    "simulationTimeMillis": 0,
    "iamRole": "arn:aws:iam::111111111111:role/
AWSRoboMakerCloudWatch-154766341-SimulationJobRole-G00BWTQ8YBG6",
    "robotApplications": [
      {
        "application": "arn:aws:robomaker:us-west-2:111111111111:robot-
application/AWSRoboMakerCloudWatch-1547663411642_NZbpqEJ3T/1547663517377",
        "applicationVersion": "$LATEST",
        "launchConfig": {
          "packageName": "cloudwatch_robot",
          "launchFile": "await_commands.launch",
          "environmentVariables": {
            "LAUNCH_ID": "1548168752173",
            "ROS_AWS_REGION": "us-west-2"
          }
        }
      }
    ],
    "simulationApplications": [
      {
        "application": "arn:aws:robomaker:us-
west-2:111111111111:simulation-application/
AWSRoboMakerCloudWatch-1547663411642_0LI6D1h6/1547663521470",
        "applicationVersion": "$LATEST",
        "launchConfig": {
          "packageName": "cloudwatch_simulation",
          "launchFile": "bookstore_turtlebot_navigation.launch",
          "environmentVariables": {
            "LAUNCH_ID": "1548168752173",
            "ROS_AWS_REGION": "us-west-2",
            "TURTLEBOT3_MODEL": "waffle_pi"
          }
        }
      }
    ]
  }
}
```

```

    }
  }
],
"tags": {},
"vpcConfig": {
  "subnets": [
    "subnet-716dd52a",
    "subnet-43c22325",
    "subnet-3f526976"
  ],
  "securityGroups": [
    "sg-3fb40545"
  ],
  "vpcId": "vpc-99895eff",
  "assignPublicIp": true
}
},
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-
g8h6tglmblgw",
  "status": "Canceled",
  "lastUpdatedAt": 1546543442.0,
  "failureBehavior": "Fail",
  "clientRequestToken": "d796bbb4-2a2c-1abc-f2a9-0d9e547d853f",
  "outputLocation": {
    "s3Bucket": "sample-bucket",
    "s3Prefix": "SimulationLog_115490482698"
  },
  "maxJobDurationInSeconds": 28800,
  "simulationTimeMillis": 0,
  "iamRole": "arn:aws:iam::111111111111:role/RoboMakerSampleTheFirst",
  "robotApplications": [
    {
      "application": "arn:aws:robomaker:us-west-2:111111111111:robot-
application/RoboMakerHelloWorldRobot/1546541208251",
      "applicationVersion": "$LATEST",
      "launchConfig": {
        "packageName": "hello_world_robot",
        "launchFile": "rotate.launch"
      }
    }
  ],
  "simulationApplications": [
    {

```

```

        "application": "arn:aws:robomaker:us-
west-2:111111111111:simulation-application/
RoboMakerHelloWorldSimulation/1546541198985",
        "applicationVersion": "$LATEST",
        "launchConfig": {
            "packageName": "hello_world_simulation",
            "launchFile": "empty_world.launch"
        }
    ],
    "tags": {}
}
],
"unprocessedJobs": []
}

```

- Para API obtener más información, consulte [BatchDescribeSimulationJob](#) la Referencia de AWS CLI comandos.

## cancel-simulation-job

En el siguiente ejemplo de código, se muestra cómo usar `cancel-simulation-job`.

### AWS CLI

Para cancelar un trabajo de simulación

El siguiente `cancel-simulation-job` ejemplo cancela el trabajo de simulación especificado.

```

aws robomaker cancel-simulation-job \
  --job arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-66bbb3gpxm8x

```

- Para API obtener más información, consulte [CancelSimulationJob](#) la Referencia de AWS CLI comandos.

## create-deployment-job

En el siguiente ejemplo de código, se muestra cómo usar `create-deployment-job`.

### AWS CLI

Para crear un trabajo de despliegue



En este ejemplo, se crea un trabajo de despliegue para la flota MyFleet. Incluye una variable de entorno denominada "ENVIRONMENT». También adjunta una etiqueta denominada «Región».

Comando:

```
aws robomaker create-deployment-job --deployment-
config concurrentDeploymentPercentage=20, failureThresholdPercentage=25
--fleet arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/
Trek/1539894765711 --tags Region=West --deployment-application-
configs application=arn:aws:robomaker:us-west-2:111111111111:robot-application/
RoboMakerVoiceInteractionRobot/1546537110575, applicationVersion=1, launchConfig={environmentV
```

Salida:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/sim-0974h36s4v0t",
  "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/
MyFleet/1539894765711",
  "status": "Pending",
  "deploymentApplicationConfigs": [
    {
      "application": "arn:aws:robomaker:us-west-2:111111111111:robot-
application/RoboMakerVoiceInteractionRobot/1546537110575",
      "applicationVersion": "1",
      "launchConfig": {
        "packageName": "voice_interaction_robot",
        "launchFile": "await_commands.launch",
        "environmentVariables": {
          "ENVIRONMENT": "Beta"
        }
      }
    }
  ],
  "createdAt": 1550770236.0,
  "deploymentConfig": {
    "concurrentDeploymentPercentage": 20,
    "failureThresholdPercentage": 25
  },
  "tags": {
    "Region": "West"
  }
}
```

- Para API obtener más información, consulte [CreateDeploymentJob](#) la Referencia de AWS CLI comandos.

## create-fleet

En el siguiente ejemplo de código, se muestra cómo usar `create-fleet`.

### AWS CLI

Para crear una flota

En este ejemplo se crea una flota. Adjunta una etiqueta llamada Región.

Comando:

```
aws robomaker create-fleet --name MyFleet --tags Region=East
```

Salida:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/
MyOtherFleet/1550771394395",
  "name": "MyFleet",
  "createdAt": 1550771394.0,
  "tags": {
    "Region": "East"
  }
}
```

- Para API obtener más información, consulte [CreateFleet](#) la Referencia de AWS CLI comandos.

## create-robot-application-version

En el siguiente ejemplo de código, se muestra cómo usar `create-robot-application-version`.

### AWS CLI

Para crear una versión de una aplicación robótica

En este ejemplo se crea una versión de la aplicación robótica.

Comando:

```
aws robomaker create-robot-application-version --application arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobotApplication/1551201873931
```

Salida:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobotApplication/1551201873931",
  "name": "MyRobotApplication",
  "version": "1",
  "sources": [
    {
      "s3Bucket": "my-bucket",
      "s3Key": "my-robot-application.tar.gz",
      "etag": "f8cf5526f1c6e7b3a72c3ed3f79c5493-70",
      "architecture": "ARMHF"
    }
  ],
  "robotSoftwareSuite": {
    "name": "ROS",
    "version": "Kinetic"
  },
  "lastUpdatedAt": 1551201873.0,
  "revisionId": "9986bb8d-a695-4ab4-8810-9f4a74d1aa00"
  "tags": {}
}
```

- Para API obtener más información, consulte [CreateRobotApplicationVersion](#) la Referencia de AWS CLI comandos.

## **create-robot-application**

En el siguiente ejemplo de código, se muestra cómo usar `create-robot-application`.

AWS CLI

Para crear una aplicación de robot

En este ejemplo se crea una aplicación de robot.

Comando:

```
aws robomaker create-robot-application --name MyRobotApplication --  
sources s3Bucket=my-bucket,s3Key=my-robot-application.tar.gz,architecture=X86_64 --  
robot-software-suite name=ROS,version=Kinetic
```

Salida:

```
{  
  "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/  
MyRobotApplication/1551201873931",  
  "name": "MyRobotApplication",  
  "version": "$LATEST",  
  "sources": [  
    {  
      "s3Bucket": "my-bucket",  
      "s3Key": "my-robot-application.tar.gz",  
      "architecture": "ARMHF"  
    }  
  ],  
  "robotSoftwareSuite": {  
    "name": "ROS",  
    "version": "Kinetic"  
  },  
  "lastUpdatedAt": 1551201873.0,  
  "revisionId": "1f3cb539-9239-4841-a656-d3efcffa07e1",  
  "tags": {}  
}
```

- Para API obtener más información, consulte [CreateRobotApplication](#) la Referencia de AWS CLI comandos.

## create-robot

En el siguiente ejemplo de código, se muestra cómo usar `create-robot`.

### AWS CLI

Para crear un robot

En este ejemplo se crea un robot. Utiliza la ARMHF arquitectura. También adjunta una etiqueta llamada Región.

Comando:

```
aws robomaker create-robot --name MyRobot --architecture ARMHF --greengrass-group-id 0f728a3c-7dbf-4a3e-976d-d16a8360caba --tags Region=East
```

Salida:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398",
  "name": "MyRobot",
  "createdAt": 1550772325.0,
  "greengrassGroupId": "0f728a3c-7dbf-4a3e-976d-d16a8360caba",
  "architecture": "ARMHF",
  "tags": {
    "Region": "East"
  }
}
```

- Para API obtener más información, consulte [CreateRobot](#) la Referencia de AWS CLI comandos.

## **create-simulation-application-version**

En el siguiente ejemplo de código, se muestra cómo usar `create-simulation-application-version`.

AWS CLI

Para crear una versión de la aplicación de simulación

En este ejemplo se crea una versión de la aplicación robótica.

Comando:

```
aws robomaker create-simulation-application-version --
application arn:aws:robomaker:us-west-2:111111111111:robot-application/
MySimulationApplication/1551203427605
```

Salida:

```
{
```

```
"arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/MyRobotApplication/1551203427605",
"name": "MyRobotApplication",
"version": "1",
"sources": [
  {
    "s3Bucket": "my-bucket",
    "s3Key": "my-simulation-application.tar.gz",
    "etag": "00d8a94ff113856688c4fce618ae0f45-94",
    "architecture": "X86_64"
  }
],
"simulationSoftwareSuite": {
  "name": "Gazebo",
  "version": "7"
},
"robotSoftwareSuite": {
  "name": "ROS",
  "version": "Kinetic"
},
"renderingEngine": {
  "name": "OGRE",
  "version": "1.x"
},
"lastUpdatedAt": 1551203853.0,
"revisionId": "ee753e53-519c-4d37-895d-65e79bcd1914",
"tags": {}
}
```

- Para API obtener más información, consulte [CreateSimulationApplicationVersion](#) la Referencia de AWS CLI comandos.

## create-simulation-application

En el siguiente ejemplo de código, se muestra cómo usar create-simulation-application.

### AWS CLI

Para crear una aplicación de simulación

En este ejemplo se crea una aplicación de simulación.

Comando:

```
aws robomaker create-simulation-application --name MyRobotApplication --  
sources s3Bucket=my-bucket,s3Key=my-simulation-application.tar.gz,architecture=ARMHF  
--robot-software-suite name=ROS,version=Kinetic --simulation-software-  
suite name=Gazebo,version=7 --rendering-engine name=OGRE,version=1.x
```

Salida:

```
{  
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/  
MyRobotApplication/1551203301792",  
  "name": "MyRobotApplication",  
  "version": "$LATEST",  
  "sources": [  
    {  
      "s3Bucket": "my-bucket",  
      "s3Key": "my-simulation-application.tar.gz",  
      "architecture": "X86_64"  
    }  
  ],  
  "simulationSoftwareSuite": {  
    "name": "Gazebo",  
    "version": "7"  
  },  
  "robotSoftwareSuite": {  
    "name": "ROS",  
    "version": "Kinetic"  
  },  
  "renderingEngine": {  
    "name": "OGRE",  
    "version": "1.x"  
  },  
  "lastUpdatedAt": 1551203301.0,  
  "revisionId": "ee753e53-519c-4d37-895d-65e79bcd1914",  
  "tags": {}  
}
```

- Para API obtener más información, consulte [CreateSimulationApplication](#) la Referencia de AWS CLI comandos.

## **create-simulation-job**

En el siguiente ejemplo de código, se muestra cómo usar `create-simulation-job`.

## AWS CLI

Para crear un trabajo de simulación

En este ejemplo se crea un trabajo de simulación. Utiliza una aplicación de robot y una aplicación de simulación.

Comando:

```
aws robomaker create-simulation-job --max-job-duration-  
in-seconds 3600 --iam-role arn:aws:iam::111111111111:role/  
AWSRoboMakerCloudWatch-154766341-SimulationJobRole-G00BWTQ8YBG6 --robot-  
applications application=arn:aws:robomaker:us-west-2:111111111111:robot-application/  
MyRobotApplication/1551203485821,launchConfig={packageName=hello_world_robot,launchFile=rotate.launch}  
--simulation-applications application=arn:aws:robomaker:us-  
west-2:111111111111:simulation-application/  
MySimulationApplication/1551203427605,launchConfig={packageName=hello_world_simulation,launchFile=rotate.launch}  
--tags Region=North
```

Salida:

```
{  
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-w7m68wpr05h8",  
  "status": "Pending",  
  "lastUpdatedAt": 1551213837.0,  
  "failureBehavior": "Fail",  
  "clientRequestToken": "b283ccce-e468-43ee-8642-be76a9d69f15",  
  "maxJobDurationInSeconds": 3600,  
  "simulationTimeMillis": 0,  
  "iamRole": "arn:aws:iam::111111111111:role/MySimulationRole",  
  "robotApplications": [  
    {  
      "application": "arn:aws:robomaker:us-west-2:111111111111:robot-  
application/MyRobotApplication/1551203485821",  
      "applicationVersion": "$LATEST",  
      "launchConfig": {  
        "packageName": "hello_world_robot",  
        "launchFile": "rotate.launch"  
      }  
    }  
  ],  
  "simulationApplications": [  
    {
```



```
    "application": "arn:aws:robomaker:us-west-2:111111111111:simulation-
application/MySimulationApplication/1551203427605",
    "applicationVersion": "$LATEST",
    "launchConfig": {
        "packageName": "hello_world_simulation",
        "launchFile": "empty_world.launch"
    }
  ],
  "tags": {
    "Region": "North"
  }
}
```

- Para API obtener más información, consulte [CreateSimulationJob](#) la Referencia de AWS CLI comandos.

## delete-fleet

En el siguiente ejemplo de código, se muestra cómo usar `delete-fleet`.

### AWS CLI

Para eliminar una flota

En este ejemplo se elimina una flota.

Comando:

```
aws robomaker delete-fleet --fleet arn:aws:robomaker:us-
west-2:111111111111:deployment-fleet/MyFleet/1550771394395
```

- Para API obtener más información, consulte [DeleteFleet](#) la Referencia de AWS CLI comandos.

## delete-robot-application

En el siguiente ejemplo de código, se muestra cómo usar `delete-robot-application`.

### AWS CLI

Para eliminar una aplicación de robot

En este ejemplo, se elimina una aplicación de robot.

Comando:

```
aws robomaker delete-robot-application --application arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobotApplication/1551203485821
```

- Para API obtener más información, consulte [DeleteRobotApplication](#) la Referencia de AWS CLI comandos.

## **delete-robot**

En el siguiente ejemplo de código, se muestra cómo usar delete-robot.

AWS CLI

Para eliminar un robot

En este ejemplo se elimina un robot.

Comando:

```
aws robomaker delete-robot --robot arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1540829698778
```

- Para API obtener más información, consulte [DeleteRobot](#) la Referencia de AWS CLI comandos.

## **delete-simulation-application**

En el siguiente ejemplo de código, se muestra cómo usar delete-simulation-application.

AWS CLI

Para eliminar una aplicación de simulación

En este ejemplo se elimina una aplicación de simulación.

Comando:

```
aws robomaker delete-simulation-application --application arn:aws:robomaker:us-west-2:111111111111:simulation-application/MySimulationApplication/1551203427605
```

- Para API obtener más información, consulte [DeleteSimulationApplication](#) la Referencia de AWS CLI comandos.

## deregister-robot

En el siguiente ejemplo de código, se muestra cómo usar `deregister-robot`.

### AWS CLI

Para anular el registro de un robot de una flota

En este ejemplo se anula el registro de un robot de una flota.

Comando:

```
aws robomaker deregister-robot --fleet arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1550771358907 --robot arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398
```

Salida:

```
{
  "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1550771358907",
  "robot": "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398"
}
```

- Para API obtener más información, consulte la Referencia [DeregisterRobot](#) de AWS CLI comandos.

## describe-deployment-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-deployment-job`.

### AWS CLI

Para describir un trabajo de despliegue

En el siguiente `describe-deployment-job` ejemplo, se recuperan los detalles sobre el trabajo de despliegue especificado.

```
aws robomaker describe-deployment-job \
  --job arn:aws:robomaker:us-west-2:111111111111:deployment-job/deployment-
xl8qssl6pbcn
```

Salida:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/deployment-
xl8qssl6pbcn",
  "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/
Trek/1539894765711",
  "status": "InProgress",
  "deploymentConfig": {
    "concurrentDeploymentPercentage": 20,
    "failureThresholdPercentage": 25
  },
  "deploymentApplicationConfigs": [
    {
      "application": "arn:aws:robomaker:us-west-2:111111111111:robot-
application/RoboMakerHelloWorldRobot/1546541208251",
      "applicationVersion": "1",
      "launchConfig": {
        "packageName": "hello_world_robot",
        "launchFile": "rotate.launch"
      }
    }
  ],
  "createdAt": 1551218369.0,
  "robotDeploymentSummary": [
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/
MyRobot/1540834232469",
      "deploymentStartTime": 1551218376.0,
      "status": "Deploying",
      "progressDetail": {}
    }
  ],
  "tags": {}
}
```

- Para API obtener más información, consulte [DescribeDeploymentJob](#) la Referencia de AWS CLI comandos.

## describe-fleet

En el siguiente ejemplo de código, se muestra cómo usar `describe-fleet`.

### AWS CLI

Para describir una flota

En el siguiente `describe-fleet` ejemplo, se recuperan los detalles de la flota especificada.

```
aws robomaker describe-fleet \  
  --fleet arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/  
MyFleet/1550771358907
```

Salida:

```
{  
  "name": "MyFleet",  
  "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/  
MyFleet/1539894765711",  
  "robots": [  
    {  
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/  
MyRobot/1540834232469",  
      "createdAt": 1540834232.0  
    },  
    {  
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/  
MyOtherRobot/1540829698778",  
      "createdAt": 1540829698.0  
    }  
  ],  
  "createdAt": 1539894765.0,  
  "lastDeploymentStatus": "Succeeded",  
  "lastDeploymentJob": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/  
deployment-xl8qssl6pbcn",  
  "lastDeploymentTime": 1551218369.0,  
  "tags": {}  
}
```

- Para API obtener más información, consulte [DescribeFleet](#) la Referencia de AWS CLI comandos.

## describe-robot-application

En el siguiente ejemplo de código, se muestra cómo usar `describe-robot-application`.

### AWS CLI

Para describir una aplicación de robot

En este ejemplo se describe una aplicación de robot.

Comando:

```
aws robomaker describe-robot-application --application arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobotApplication/1551203485821
```

Salida:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobotApplication/1551203485821",
  "name": "MyRobotApplication",
  "version": "$LATEST",
  "sources": [
    {
      "s3Bucket": "my-bucket",
      "s3Key": "my-robot-application.tar.gz",
      "architecture": "X86_64"
    }
  ],
  "robotSoftwareSuite": {
    "name": "ROS",
    "version": "Kinetic"
  },
  "revisionId": "e72efe0d-f44f-4333-b604-f6fa5c6bb50b",
  "lastUpdatedAt": 1551203485.0,
  "tags": {}
}
```

- Para API obtener más información, consulte [DescribeRobotApplication](#) la Referencia de AWS CLI comandos.

## describe-robot

En el siguiente ejemplo de código, se muestra cómo usar `describe-robot`.

### AWS CLI

Para describir un robot

En este ejemplo se describe un robot.

Comando:

```
aws robomaker describe-robot --robot arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398
```

Salida:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398",
  "name": "MyRobot",
  "status": "Available",
  "greengrassGroupId": "0f728a3c-7dbf-4a3e-976d-d16a8360caba",
  "createdAt": 1550772325.0,
  "architecture": "ARMHF",
  "tags": {
    "Region": "East"
  }
}
```

- Para API obtener más información, consulte [DescribeRobot](#) la Referencia de AWS CLI comandos.

## describe-simulation-application

En el siguiente ejemplo de código, se muestra cómo usar `describe-simulation-application`.

### AWS CLI

Para describir una aplicación de simulación

En este ejemplo se describe una aplicación de simulación.

**Comando:**

```
aws robomaker describe-simulation-application --application arn:aws:robomaker:us-west-2:111111111111:simulation-application/MySimulationApplication/1551203427605
```

**Salida:**

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/MySimulationApplication/1551203427605",
  "name": "MySimulationApplication",
  "version": "$LATEST",
  "sources": [
    {
      "s3Bucket": "my-bucket",
      "s3Key": "my-simulation-application.tar.gz",
      "architecture": "X86_64"
    }
  ],
  "simulationSoftwareSuite": {
    "name": "Gazebo",
    "version": "7"
  },
  "robotSoftwareSuite": {
    "name": "ROS",
    "version": "Kinetic"
  },
  "renderingEngine": {
    "name": "OGRE",
    "version": "1.x"
  },
  "revisionId": "783674ab-b7b8-42d9-b01f-9373907987e5",
  "lastUpdatedAt": 1551203427.0,
  "tags": {}
}
```

- Para API obtener más información, consulte [DescribeSimulationApplication](#) la Referencia de AWS CLI comandos.

**describe-simulation-job**

En el siguiente ejemplo de código, se muestra cómo usar `describe-simulation-job`.



## AWS CLI

Para describir un trabajo de simulación

En este ejemplo se describe un trabajo de simulación.

Comando:

```
aws robomaker describe-simulation-job --job arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-pql32v7pfjy6
```

Salida:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-pql32v7pfjy6",
  "status": "Running",
  "lastUpdatedAt": 1551219349.0,
  "failureBehavior": "Continue",
  "clientRequestToken": "a19ec4b5-e50d-3591-33da-c2e593c60615",
  "outputLocation": {
    "s3Bucket": "my-output-bucket",
    "s3Prefix": "output"
  },
  "maxJobDurationInSeconds": 3600,
  "simulationTimeMillis": 0,
  "iamRole": "arn:aws:iam::111111111111:role/MySimulationRole",
  "robotApplications": [
    {
      "application": "arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobotApplication/1551206341136",
      "applicationVersion": "$LATEST",
      "launchConfig": {
        "packageName": "hello_world_robot",
        "launchFile": "rotate.launch"
      }
    }
  ],
  "simulationApplications": [
    {
      "application": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/MySimulationApplication/1551206347967",
      "applicationVersion": "$LATEST",
      "launchConfig": {
```

```
        "packageName": "hello_world_simulation",
        "launchFile": "empty_world.launch"
    }
  ],
  "tags": {}
}
```

- Para API obtener más información, consulte [DescribeSimulationJob](#) la Referencia de AWS CLI comandos.

## list-deployment-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-deployment-jobs`.

AWS CLI

Para enumerar los trabajos de despliegue

En el siguiente `list-deployment-jobs` ejemplo, se recupera una lista de trabajos de despliegue.

```
aws robomaker list-deployment-jobs
```

Salida:

```
{
  "deploymentJobs": [
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/sim-6293szzm56rv",
      "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1539894765711",
      "status": "InProgress",
      "deploymentApplicationConfigs": [
        {
          "application": "arn:aws:robomaker:us-west-2:111111111111:robot-application/HelloWorldRobot/1546537110575",
          "applicationVersion": "1",
          "launchConfig": {
            "packageName": "hello_world_robot",
```

```

        "launchFile": "rotate.launch",
        "environmentVariables": {
            "ENVIRONMENT": "Desert"
        }
    }
},
"deploymentConfig": {
    "concurrentDeploymentPercentage": 20,
    "failureThresholdPercentage": 25
},
"createdAt": 1550689373.0
},
{
    "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/
deployment-4w4g69p25zdb",
    "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/
MyFleet/1539894765711",
    "status": "Pending",
    "deploymentApplicationConfigs": [
        {
            "application": "arn:aws:robomaker:us-west-2:111111111111:robot-
application/AWSRoboMakerHelloWorld-1544562726923_YGHM_sh5M/1544562822877",
            "applicationVersion": "1",
            "launchConfig": {
                "packageName": "fail",
                "launchFile": "fail"
            }
        }
    ],
    "deploymentConfig": {
        "concurrentDeploymentPercentage": 20,
        "failureThresholdPercentage": 25
    },
    "failureReason": "",
    "failureCode": "",
    "createdAt": 1544719763.0
}
]
}

```

- Para API obtener más información, consulte [ListDeploymentJobs](#) la Referencia de AWS CLI comandos.

## list-fleets

En el siguiente ejemplo de código, se muestra cómo usar `list-fleets`.

### AWS CLI

Para enumerar las flotas

En este ejemplo se enumeran las flotas. Se devolverá un máximo de 20 flotas.

Comando:

```
aws robomaker list-fleets --max-items 20
```

Salida:

```
{
  "fleetDetails": [
    {
      "name": "Trek",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1539894765711",
      "createdAt": 1539894765.0,
      "lastDeploymentStatus": "Failed",
      "lastDeploymentJob": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/deployment-4w4g69p25zdb",
      "lastDeploymentTime": 1544719763.0
    }
  ]
}
```

- Para API obtener más información, consulte [ListFleets](#) la Referencia de AWS CLI comandos.

## list-robot-applications

En el siguiente ejemplo de código, se muestra cómo usar `list-robot-applications`.

### AWS CLI

Para enumerar las aplicaciones de los robots

En este ejemplo se enumeran las aplicaciones de robots. Los resultados están limitados a 20 aplicaciones robóticas.

Comando:

```
aws robomaker list-robot-applications --max-results 20
```

Salida:

```
{
  "robotApplicationSummaries": [
    {
      "name": "MyRobot",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobot/1546537110575",
      "version": "$LATEST",
      "lastUpdatedAt": 1546540372.0
    },
    {
      "name": "AnotherRobot",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/AnotherRobot/1546541208251",
      "version": "$LATEST",
      "lastUpdatedAt": 1546541208.0
    },
    {
      "name": "MySuperRobot",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/MySuperRobot/1547663517377",
      "version": "$LATEST",
      "lastUpdatedAt": 1547663517.0
    }
  ]
}
```

- Para API obtener más información, consulte [ListRobotApplications](#) la Referencia de AWS CLI comandos.

## list-robots

En el siguiente ejemplo de código, se muestra cómo usar `list-robots`.

AWS CLI

Para enumerar los robots

En este ejemplo se enumeran los robots. Se devolverá un máximo de 20 robots.

Comando:

```
aws robomaker list-robots --max-results 20
```

Salida:

```
{
  "robots": [
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/Robot100/1544035373264",
      "name": "Robot100",
      "status": "Available",
      "createdAt": 1544035373.0,
      "architecture": "X86_64"
    },
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/Robot101/1542146976587",
      "name": "Robot101",
      "status": "Available",
      "createdAt": 1542146976.0,
      "architecture": "X86_64"
    },
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/Robot102/1540834232469",
      "fleetArn": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/Trek/1539894765711",
      "status": "Available",
      "createdAt": 1540834232.0,
      "architecture": "X86_64",
      "lastDeploymentJob": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/deployment-jb007b75gl5f",
      "lastDeploymentTime": 1550689533.0
    },
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1540829698778",
      "name": "MyRobot",

```

```

        "status": "Registered",
        "createdAt": 1540829698.0,
        "architecture": "X86_64"
    }
]
}

```

- Para API obtener más información, consulte [ListRobots](#) la Referencia de AWS CLI comandos.

## list-simulation-applications

En el siguiente ejemplo de código, se muestra cómo usar `list-simulation-applications`.

### AWS CLI

Para enumerar las aplicaciones de simulación

En este ejemplo se enumeran las aplicaciones de simulación. Se devolverá un máximo de 20 aplicaciones de simulación.

Comando:

```
aws robomaker list-simulation-applications --max-results 20
```

Salida:

```

{
  "simulationApplicationSummaries": [
    {
      "name": "AWSRoboMakerObjectTracker-1548959046124_NPvyfcatq",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/AWSRoboMakerObjectTracker-1548959046124_NPvyfcatq/1548959170096",
      "version": "$LATEST",
      "lastUpdatedAt": 1548959170.0
    },
    {
      "name": "RoboMakerHelloWorldSimulation",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/RoboMakerHelloWorldSimulation/1546541198985",
      "version": "$LATEST",
      "lastUpdatedAt": 1546541198.0
    },
  ],
}

```

```

    {
      "name": "RoboMakerObjectTrackerSimulation",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/RoboMakerObjectTrackerSimulation/1545846795615",
      "version": "$LATEST",
      "lastUpdatedAt": 1545847405.0
    },
    {
      "name": "RoboMakerVoiceInteractionSimulation",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/RoboMakerVoiceInteractionSimulation/1546537100507",
      "version": "$LATEST",
      "lastUpdatedAt": 1546540352.0
    },
    {
      "name": "AWSRoboMakerCloudWatch-1547663411642_0LI6D1h6",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/AWSRoboMakerCloudWatch-1547663411642_0LI6D1h6/1547663521470",
      "version": "$LATEST",
      "lastUpdatedAt": 1547663521.0
    },
    {
      "name": "AWSRoboMakerDeepRacer-1545848257672_1YZCaieQ-",
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/AWSRoboMakerDeepRacer-1545848257672_1YZCaieQ-/1545848370525",
      "version": "$LATEST",
      "lastUpdatedAt": 1545848370.0
    }
  ]
}

```

- Para API obtener más información, consulte [ListSimulationApplications](#) la Referencia de AWS CLI comandos.

## list-simulation-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-simulation-jobs`.

### AWS CLI

Para enumerar los trabajos de simulación

En este ejemplo se enumeran los trabajos de simulación.



## Comando:

```
aws robomaker list-simulation-jobs
```

## Salida:

```
{
  "simulationJobSummaries": [
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-66bbb3gpxm8x",
      "lastUpdatedAt": 1548959178.0,
      "status": "Completed",
      "simulationApplicationNames": [
        "AWSRoboMakerObjectTracker-1548959046124_NPvyfcatq"
      ],
      "robotApplicationNames": [
        null
      ]
    },
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-b27c4rkrtzcx",
      "lastUpdatedAt": 1543514088.0,
      "status": "Canceled",
      "simulationApplicationNames": [
        "AWSRoboMakerPersonDetection-1543513948280_T8rHW2_lu"
      ],
      "robotApplicationNames": [
        "AWSRoboMakerPersonDetection-1543513948280_EYaMT0mYb"
      ]
    },
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-51vxjbyz4q8t",
      "lastUpdatedAt": 1543508858.0,
      "status": "Canceled",
      "simulationApplicationNames": [
        "AWSRoboMakerCloudWatch-1543504747391_1FF9ZQyx6"
      ],
      "robotApplicationNames": [
        "AWSRoboMakerCloudWatch-1543504747391_axbYa3S3K"
      ]
    }
  ]
}
```

```

    },
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-
kgf1fqxflqbx",
      "lastUpdatedAt": 1543504862.0,
      "status": "Completed",
      "simulationApplicationNames": [
        "AWSRoboMakerCloudWatch-1543504747391_1FF9ZQyx6"
      ],
      "robotApplicationNames": [
        "AWSRoboMakerCloudWatch-1543504747391_axbYa3S3K"
      ]
    },
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-
vw8lvh061nqt",
      "lastUpdatedAt": 1543441430.0,
      "status": "Completed",
      "simulationApplicationNames": [
        "AWSRoboMakerHelloWorld-1543437372341__yb_Jg961"
      ],
      "robotApplicationNames": [
        "AWSRoboMakerHelloWorld-1543437372341_lNbmKHvs9"
      ]
    },
    {
      "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-job/sim-
txy5ypxmh84",
      "lastUpdatedAt": 1543437488.0,
      "status": "Completed",
      "simulationApplicationNames": [
        "AWSRoboMakerHelloWorld-1543437372341__yb_Jg961"
      ],
      "robotApplicationNames": [
        "AWSRoboMakerHelloWorld-1543437372341_lNbmKHvs9"
      ]
    }
  ]
}

```

- Para API obtener más información, consulte [ListSimulationJobs](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un recurso

En este ejemplo, se muestran las etiquetas de un AWS RoboMaker recurso.

Comando:

```
aws robomaker list-tags-for-resource --resource-arn "arn:aws:robomaker:us-west-2:111111111111:robot/Robby_the_Robot/1544035373264"
```

Salida:

```
{
  "tags": {
    "Region": "North",
    "Stage": "Initial"
  }
}
```

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## register-robot

En el siguiente ejemplo de código, se muestra cómo usar `register-robot`.

### AWS CLI

Para registrar un robot

En este ejemplo, se registra un robot en una flota.

Comando:

```
aws robomaker register-robot --fleet arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/MyFleet/1550771358907 --robot arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398
```

Salida:

```
{
  "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/
MyFleet/1550771358907",
  "robot": "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1550772324398"
}
```

- Para API obtener más información, consulte [RegisterRobot](#) la Referencia de AWS CLI comandos.

## restart-simulation-job

En el siguiente ejemplo de código, se muestra cómo usar `restart-simulation-job`.

AWS CLI

Para reiniciar una simulación

En este ejemplo se reinicia una simulación.

Comando:

```
aws robomaker restart-simulation-job --job arn:aws:robomaker:us-
west-2:111111111111:simulation-job/sim-t6rdgt70mftr
```

- Para API obtener más información, consulte [RestartSimulationJob](#) la Referencia de AWS CLI comandos.

## sync-deployment-job

En el siguiente ejemplo de código, se muestra cómo usar `sync-deployment-job`.

AWS CLI

Para sincronizar un trabajo de despliegue

En este ejemplo se sincroniza un trabajo de despliegue.

Comando:

```
aws robomaker sync-deployment-job --fleet arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/Trek/1539894765711
```

Salida:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:deployment-job/
deployment-09ccxs3t1fms",
  "fleet": "arn:aws:robomaker:us-west-2:111111111111:deployment-fleet/
MyFleet/1539894765711",
  "status": "Pending",
  "deploymentConfig": {
    "concurrentDeploymentPercentage": 20,
    "failureThresholdPercentage": 25
  },
  "deploymentApplicationConfigs": [
    {
      "application": "arn:aws:robomaker:us-west-2:111111111111:robot-
application/MyRobotApplication/1546541208251",
      "applicationVersion": "1",
      "launchConfig": {
        "packageName": "hello_world_simulation",
        "launchFile": "empty_world.launch"
      }
    }
  ],
  "createdAt": 1551286954.0
}
```

- Para API obtener más información, consulte [SyncDeploymentJob](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar tag-resource.

### AWS CLI

Para etiquetar un recurso

En este ejemplo, se etiqueta un recurso. Adjunta dos etiquetas: Región y Etapa.

Comando:

```
aws robomaker tag-resource --resource-arn "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1544035373264" --tags Region=North,Stage=Initial
```

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

AWS CLI

Para quitar la etiqueta de un recurso

En este ejemplo, se elimina una etiqueta de un recurso. Elimina la etiqueta de región.

Comando:

```
aws robomaker untag-resource --resource-arn "arn:aws:robomaker:us-west-2:111111111111:robot/MyRobot/1544035373264" --tag-keys Region
```

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-robot-application

En el siguiente ejemplo de código, se muestra cómo usar `update-robot-application`.

AWS CLI

Para actualizar una aplicación de robot

En este ejemplo se actualiza una aplicación de robot.

Comando:

```
aws robomaker update-robot-application --application arn:aws:robomaker:us-west-2:111111111111:robot-application/MyRobotApplication/1551203485821 --
```

```
sources s3Bucket=my-bucket,s3Key=my-robot-application.tar.gz,architecture=X86_64 --  
robot-software-suite name=ROS,version=Kinetic
```

Salida:

```
{  
  "arn": "arn:aws:robomaker:us-west-2:111111111111:robot-application/  
MyRobotApplication/1551203485821",  
  "name": "MyRobotApplication",  
  "version": "$LATEST",  
  "sources": [  
    {  
      "s3Bucket": "my-bucket",  
      "s3Key": "my-robot-application.tar.gz",  
      "architecture": "X86_64"  
    }  
  ],  
  "robotSoftwareSuite": {  
    "name": "ROS",  
    "version": "Kinetic"  
  },  
  "lastUpdatedAt": 1551287993.0,  
  "revisionId": "20b5e331-24fd-4504-8b8c-531afe5f4c94"  
}
```

- Para API obtener más información, consulte [UpdateRobotApplication](#) la Referencia de AWS CLI comandos.

## update-simulation-application

En el siguiente ejemplo de código, se muestra cómo usar `update-simulation-application`.

AWS CLI

Para actualizar una aplicación de simulación

En este ejemplo se actualiza una aplicación de simulación.

Comando:

```
aws robomaker update-simulation-application --application arn:aws:robomaker:us-  
west-2:111111111111:simulation-application/MySimulationApplication/1551203427605 --
```

```
sources s3Bucket=my-bucket,s3Key=my-simulation-
application.tar.gz,architecture=X86_64 --robot-software-
suite name=ROS,version=Kinetic --simulation-software-suite name=Gazebo,version=7 --
rendering-engine name=OGRE,version=1.x
```

Salida:

```
{
  "arn": "arn:aws:robomaker:us-west-2:111111111111:simulation-application/
MySimulationApplication/1551203427605",
  "name": "MySimulationApplication",
  "version": "$LATEST",
  "sources": [
    {
      "s3Bucket": "my-bucket",
      "s3Key": "my-simulation-application.tar.gz",
      "architecture": "X86_64"
    }
  ],
  "simulationSoftwareSuite": {
    "name": "Gazebo",
    "version": "7"
  },
  "robotSoftwareSuite": {
    "name": "ROS",
    "version": "Kinetic"
  },
  "renderingEngine": {
    "name": "OGRE",
    "version": "1.x"
  },
  "lastUpdatedAt": 1551289361.0,
  "revisionId": "4a22cb5d-93c5-4cef-9311-52bdd119b79e"
}
```

- Para API obtener más información, consulte [UpdateSimulationApplication](#) la Referencia de AWS CLI comandos.

## Ejemplos de Route 53 usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Route 53.



Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

Acciones

### **change-resource-record-sets**

En el siguiente ejemplo de código, se muestra cómo usar `change-resource-record-sets`.

AWS CLI

Para crear, actualizar o eliminar un conjunto de registros de recursos

El siguiente `change-resource-record-sets` comando crea un conjunto de registros de recursos mediante la configuración JSON con formato `hosted-zone-id Z1R8UBAEXAMPLE` y del archivo: `C:\awscli\route53\change-resource-record-sets.json`

```
aws route53 change-resource-record-sets --hosted-zone-id Z1R8UBAEXAMPLE --change-batch file://C:\awscli\route53\change-resource-record-sets.json
```

Para obtener más información, consulte `POST ChangeResourceRecordSets` la APIreferencia de Amazon Route 53.

La configuración del JSON archivo depende del tipo de conjunto de registros de recursos que desee crear:

BasicWeightedAliasWeighted AliasLatencyLatency AliasFailoverFailover Alias

Sintaxis básica:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
```

```

{
  "Action": "CREATE"|"DELETE"|"UPSERT",
  "ResourceRecordSet": {
    "Name": "DNS domain name",
    "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
    "TTL": time to live in seconds,
    "ResourceRecords": [
      {
        "Value": "applicable value for the record type"
      },
      {...}
    ]
  },
  {...}
}

```

#### Sintaxis ponderada:

```

{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Weight": value between 0 and 255,
        "TTL": time to live in seconds,
        "ResourceRecords": [
          {
            "Value": "applicable value for the record type"
          },
          {...}
        ],
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}

```

**Sintaxis de alias:**

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "AliasTarget": {
          "HostedZoneId": "hosted zone ID for your CloudFront distribution, Amazon
          S3 bucket, Elastic Load Balancing load balancer, or Amazon Route 53 hosted zone",
          "DNSName": "DNS domain name for your CloudFront distribution, Amazon S3
          bucket, Elastic Load Balancing load balancer, or another resource record set in
          this hosted zone",
          "EvaluateTargetHealth": true|false
        },
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}
```

**Sintaxis de alias ponderada:**

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Weight": value between 0 and 255,
        "AliasTarget": {
          "HostedZoneId": "hosted zone ID for your CloudFront distribution, Amazon
          S3 bucket, Elastic Load Balancing load balancer, or Amazon Route 53 hosted zone",
          "DNSName": "DNS domain name for your CloudFront distribution, Amazon S3
          bucket, Elastic Load Balancing load balancer, or another resource record set in
          this hosted zone",

```

```

        "EvaluateTargetHealth": true|false
    },
    "HealthCheckId": "optional ID of an Amazon Route 53 health check"
}
},
{...}
]
}

```

### Sintaxis de latencia:

```

{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Region": "Amazon EC2 region name",
        "TTL": time to live in seconds,
        "ResourceRecords": [
          {
            "Value": "applicable value for the record type"
          },
          {...}
        ],
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}

```

### Sintaxis de alias de latencia:

```

{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {

```

```

    "Name": "DNS domain name",
    "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
    "SetIdentifier": "unique description for this resource record set",
    "Region": "Amazon EC2 region name",
    "AliasTarget": {
      "HostedZoneId": "hosted zone ID for your CloudFront distribution, Amazon
S3 bucket, Elastic Load Balancing load balancer, or Amazon Route 53 hosted zone",
      "DNSName": "DNS domain name for your CloudFront distribution, Amazon S3
bucket, Elastic Load Balancing load balancer, or another resource record set in
this hosted zone",
      "EvaluateTargetHealth": true|false
    },
    "HealthCheckId": "optional ID of an Amazon Route 53 health check"
  }
  {...}
]
}

```

### Sintaxis de conmutación por error:

```

{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Failover": "PRIMARY" | "SECONDARY",
        "TTL": time to live in seconds,
        "ResourceRecords": [
          {
            "Value": "applicable value for the record type"
          },
          {...}
        ],
        "HealthCheckId": "ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}

```

```
}

```

Sintaxis del alias de conmutación por error:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Failover": "PRIMARY" | "SECONDARY",
        "AliasTarget": {
          "HostedZoneId": "hosted zone ID for your CloudFront distribution, Amazon
          S3 bucket, Elastic Load Balancing load balancer, or Amazon Route 53 hosted zone",
          "DNSName": "DNS domain name for your CloudFront distribution, Amazon S3
          bucket, Elastic Load Balancing load balancer, or another resource record set in
          this hosted zone",
          "EvaluateTargetHealth": true|false
        },
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}
```

- Para API obtener más información, consulte [ChangeResourceRecordSets](#) la Referencia de AWS CLI comandos.

## change-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `change-tags-for-resource`.

### AWS CLI

El siguiente comando agrega una etiqueta con el nombre `owner` de un recurso de verificación de estado especificado por el ID:

```
aws route53 change-tags-for-resource --resource-type healthcheck --resource-id 6233434j-18c1-34433-ba8e-3443434 --add-tags Key=owner,Value=myboss
```

El siguiente comando elimina una etiqueta nombrada `owner` de un recurso de zona hospedada especificado por el ID:

```
aws route53 change-tags-for-resource --resource-type hostedzone --resource-id Z1523434445 --remove-tag-keys owner
```

- Para API obtener más información, consulte [ChangeTagsForResource](#) la Referencia de AWS CLI comandos.

## create-health-check

En el siguiente ejemplo de código, se muestra cómo usar `create-health-check`.

### AWS CLI

Para crear un chequeo de estado

El siguiente `create-health-check` comando crea una comprobación de estado utilizando la referencia de la persona que llama `2014-04-01-18:47` y la configuración JSON con formato L del archivo: `C:\awscli\route53\create-health-check.json`

```
aws route53 create-health-check --caller-reference 2014-04-01-18:47 --health-check-config file://C:\awscli\route53\create-health-check.json
```

JSONsintaxis:

```
{
  "IPAddress": "IP address of the endpoint to check",
  "Port": port on the endpoint to check--required when Type is "TCP",
  "Type": "HTTP"|"HTTPS"|"HTTP_STR_MATCH"|"HTTPS_STR_MATCH"|"TCP",
  "ResourcePath": "path of the file that you want Amazon Route 53 to request--all Types except TCP",
  "FullyQualifiedDomainName": "domain name of the endpoint to check--all Types except TCP",
  "SearchString": "if Type is HTTP_STR_MATCH or HTTPS_STR_MATCH, the string to search for in the response body from the specified resource",
  "RequestInterval": 10 | 30,
  "FailureThreshold": integer between 1 and 10
```

```
}
```

Para añadir la comprobación de estado a un conjunto de registros de recursos de Route 53, utilice el `change-resource-record-sets` comando.

Para obtener más información, consulte Comprobaciones de estado y DNS conmutación por error de Amazon Route 53 en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [CreateHealthCheck](#) la Referencia de AWS CLI comandos.

## **create-hosted-zone**

En el siguiente ejemplo de código, se muestra cómo usar `create-hosted-zone`.

### AWS CLI

Para crear una zona alojada

El siguiente `create-hosted-zone` comando agrega una zona hospedada denominada `example.com` mediante la referencia `2014-04-01-18:47` de la persona que llama. El comentario opcional incluye un espacio, por lo que debe escribirse entre comillas:

```
aws route53 create-hosted-zone --name example.com --caller-reference 2014-04-01-18:47 --hosted-zone-config Comment="command-line version"
```

Para obtener más información, consulte Trabajar con zonas alojadas en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [CreateHostedZone](#) la Referencia de AWS CLI comandos.

## **delete-health-check**

En el siguiente ejemplo de código, se muestra cómo usar `delete-health-check`.

### AWS CLI

Para eliminar una comprobación de estado

El siguiente `delete-health-check` comando elimina la comprobación de estado con un `health-check-id` `dee75b48d9-547a-4c3d-88a5-ae4002397608`:



```
aws route53 delete-health-check --health-check-id e75b48d9-547a-4c3d-88a5-ae4002397608
```

- Para API obtener más información, consulte [DeleteHealthCheck](#) la Referencia de AWS CLI comandos.

## delete-hosted-zone

En el siguiente ejemplo de código, se muestra cómo usar delete-hosted-zone.

### AWS CLI

Para eliminar una zona alojada

El siguiente delete-hosted-zone comando elimina la zona alojada con un id de Z36KTIQEXAMPLE:

```
aws route53 delete-hosted-zone --id Z36KTIQEXAMPLE
```

- Para API obtener más información, consulte [DeleteHostedZone](#) la Referencia de AWS CLI comandos.

## get-change

En el siguiente ejemplo de código, se muestra cómo usar get-change.

### AWS CLI

Para obtener el estado de un cambio en los conjuntos de registros de recursos

El siguiente get-change comando obtiene el estado y otra información sobre la change-resource-record-sets solicitud que tiene un valor Id de /change/CWPIK4URU2I5S:

```
aws route53 get-change --id /change/CWPIK4URU2I5S
```

- Para API obtener más información, consulte [GetChange](#) la Referencia de AWS CLI comandos.

## get-health-check

En el siguiente ejemplo de código, se muestra cómo usar get-health-check.

## AWS CLI

Para obtener información sobre un chequeo de estado

El siguiente `get-health-check` comando obtiene información sobre la comprobación de estado que tiene una `health-check-id` de las `02ec8401-9879-4259-91fa-04e66d094674` siguientes características:

```
aws route53 get-health-check --health-check-id 02ec8401-9879-4259-91fa-04e66d094674
```

- Para API obtener más información, consulte [GetHealthCheck](#) la Referencia de AWS CLI comandos.

## get-hosted-zone

En el siguiente ejemplo de código, se muestra cómo usar `get-hosted-zone`.

### AWS CLI

Para obtener información sobre una zona alojada

El siguiente `get-hosted-zone` comando obtiene información sobre la zona alojada con una `id` de las siguientes opciones `Z1R8UBAEXAMPLE`:

```
aws route53 get-hosted-zone --id Z1R8UBAEXAMPLE
```

- Para API obtener más información, consulte [GetHostedZone](#) la Referencia de AWS CLI comandos.

## list-health-checks

En el siguiente ejemplo de código, se muestra cómo usar `list-health-checks`.

### AWS CLI

Para enumerar los controles de estado asociados a la AWS cuenta corriente

El siguiente `list-health-checks` comando muestra información detallada sobre las 100 primeras comprobaciones de estado asociadas a la AWS cuenta corriente. :

```
aws route53 list-health-checks
```

Si tiene más de 100 controles de estado o si quiere incluirlos en grupos de menos de 100, incluya el `--max-items` parámetro. Por ejemplo, para enumerar las comprobaciones de estado de una en una, utilice el siguiente comando:

```
aws route53 list-health-checks --max-items 1
```

Para ver la siguiente comprobación de estado, tome el valor `NextToken` de de la respuesta al comando anterior e inclúyalo en el `--starting-token` parámetro, por ejemplo:

```
aws route53 list-health-checks --max-items 1 --starting-token Z3M3LMPEXAMPLE
```

- Para API obtener más información, consulte [ListHealthChecks](#) la Referencia de AWS CLI comandos.

## list-hosted-zones-by-name

En el siguiente ejemplo de código, se muestra cómo usar `list-hosted-zones-by-name`.

### AWS CLI

El siguiente comando muestra hasta 100 zonas alojadas ordenadas por nombre de dominio:

```
aws route53 list-hosted-zones-by-name
```

Salida:

```
{
  "HostedZones": [
    {
      "ResourceRecordSetCount": 2,
      "CallerReference": "test20150527-2",
      "Config": {
        "Comment": "test2",
        "PrivateZone": false
      },
      "Id": "/hostedzone/Z119WBBTVP5WFX",
      "Name": "2.example.com."
    },
  ],
}
```

```
{
  "ResourceRecordSetCount": 2,
  "CallerReference": "test20150527-1",
  "Config": {
    "Comment": "test",
    "PrivateZone": false
  },
  "Id": "/hostedzone/Z3P5QSUBK4POTI",
  "Name": "www.example.com."
}
],
"IsTruncated": false,
"MaxItems": "100"
}
```

El siguiente comando muestra las zonas alojadas ordenadas por nombre, empezando por `www.example.com`:

```
aws route53 list-hosted-zones-by-name --dns-name www.example.com
```

Salida:

```
{
  "HostedZones": [
    {
      "ResourceRecordSetCount": 2,
      "CallerReference": "mwunder120150527-1",
      "Config": {
        "Comment": "test",
        "PrivateZone": false
      },
      "Id": "/hostedzone/Z3P5QSUBK4POTI",
      "Name": "www.example.com."
    }
  ],
  "DNSName": "www.example.com",
  "IsTruncated": false,
  "MaxItems": "100"
}
```

- Para API obtener más información, consulte [ListHostedZonesByName](#) la Referencia de AWS CLI comandos.

## list-hosted-zones

En el siguiente ejemplo de código, se muestra cómo usar `list-hosted-zones`.

### AWS CLI

Para ver una lista de las zonas alojadas asociadas a la AWS cuenta corriente

El siguiente `list-hosted-zones` comando muestra información resumida sobre las primeras 100 zonas alojadas asociadas a la AWS cuenta actual. :

```
aws route53 list-hosted-zones
```

Si tiene más de 100 zonas alojadas o si desea enumerarlas en grupos de menos de 100, incluya el parámetro `--max-items`. Por ejemplo, para enumerar las zonas alojadas, utilice el siguiente comando:

```
aws route53 list-hosted-zones --max-items 1
```

Para ver información sobre la siguiente zona alojada, tome el valor de `NextToken` de la respuesta al comando anterior e inclúyalo en el parámetro `--starting-token`, por ejemplo:

```
aws route53 list-hosted-zones --max-items 1 --starting-token Z3M3LMPEXAMPLE
```

- Para API obtener más información, consulte [ListHostedZones](#) la Referencia de AWS CLI comandos.

## list-query-logging-configs

En el siguiente ejemplo de código, se muestra cómo usar `list-query-logging-configs`.

### AWS CLI

Para ver una lista de las configuraciones de registro de consultas

En el siguiente `list-query-logging-configs` ejemplo, se muestra información sobre las 100 primeras configuraciones de registro de consultas de su AWS cuenta, para la zona alojada `Z10X3WQEXAMPLE`.

```
aws route53 list-query-logging-configs \
```

```
--hosted-zone-id Z10X3WQEXAMPLE
```

Salida:

```
{
  "QueryLoggingConfigs": [
    {
      "Id": "964ff34e-ae03-4f06-80a2-9683cexample",
      "HostedZoneId": "Z10X3WQEXAMPLE",
      "CloudWatchLogsLogGroupArn": "arn:aws:logs:us-east-1:111122223333:log-
group:/aws/route53/example.com:*"
    }
  ]
}
```

Para obtener más información, consulte [Registro de DNS consultas](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [ListQueryLoggingConfigs](#) la Referencia de AWS CLI comandos.

## list-resource-record-sets

En el siguiente ejemplo de código, se muestra cómo usar `list-resource-record-sets`.

AWS CLI

Para enumerar los conjuntos de registros de recursos de una zona alojada

El siguiente `list-resource-record-sets` comando muestra información resumida sobre los primeros 100 conjuntos de registros de recursos de una zona hospedada específica. :

```
aws route53 list-resource-record-sets --hosted-zone-id Z2LD58HEXAMPLE
```

Si la zona alojada contiene más de 100 conjuntos de registros de recursos, o si desea enumerarlos en grupos de menos de 100, incluya el `--max-items` parámetro. Por ejemplo, para enumerar los conjuntos de registros de recursos de uno en uno, utilice el siguiente comando:

```
aws route53 list-resource-record-sets --hosted-zone-id Z2LD58HEXAMPLE --max-items 1
```

Para ver información sobre el siguiente registro de recursos establecido en la zona alojada, tome el valor `NextToken` de de la respuesta al comando anterior e inclúyalo en el `--starting-token` parámetro, por ejemplo:

```
aws route53 list-resource-record-sets --hosted-zone-id Z2LD58HEXAMPLE --max-items 1
--starting-token Z3M3LMPEXAMPLE
```

Para ver todos los conjuntos de registros de recursos con un nombre concreto, utilice el `--query` parámetro para filtrarlos. Por ejemplo:

```
aws route53 list-resource-record-sets --hosted-zone-id Z2LD58HEXAMPLE --
query "ResourceRecordSets[?Name == 'example.domain.']"
```

- Para API obtener más información, consulte [ListResourceRecordSets](#) la Referencia de AWS CLI comandos.

## Ejemplos de registro de dominios de Route 53 utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante el AWS Command Line Interface registro de dominios de Route 53.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

#### **check-domain-availability**

En el siguiente ejemplo de código, se muestra cómo usar `check-domain-availability`.

## AWS CLI

Para determinar si puede registrar un nombre de dominio con Route 53

El siguiente `check-domain-availability` comando devuelve información sobre si el nombre de dominio `example.com` está disponible para registrarse mediante Route 53.

Este comando se ejecuta solo en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains check-domain-availability \  
  --region us-east-1 \  
  --domain-name example.com
```

Salida:

```
{  
  "Availability": "UNAVAILABLE"  
}
```

Route 53 admite una gran cantidad de dominios de nivel superior (TLDs), como `.com` y `.jp`, pero no admitimos todos los disponibles TLDs. Si compruebas la disponibilidad de un dominio y Route 53 no lo admite TLD, `check-domain-availability` devuelve el siguiente mensaje.

```
An error occurred (UnsupportedTLD) when calling the CheckDomainAvailability  
operation: <top-level domain> tld is not supported.
```

Para obtener una lista de los TLDs que puede utilizar al registrar un dominio en Route 53, consulte [Dominios que puede registrar en Amazon Route 53](#) en la Guía para desarrolladores de Amazon Route 53. Para obtener más información sobre el registro de dominios en Amazon Route 53, consulte [Registrar un nuevo dominio](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [CheckDomainAvailability](#) la Referencia de AWS CLI comandos.

### **check-domain-transferability**

En el siguiente ejemplo de código, se muestra cómo usar `check-domain-transferability`.



## AWS CLI

Para determinar si un dominio se puede transferir a Route 53

El siguiente `check-domain-transferability` comando devuelve información sobre si puede transferir el nombre de dominio `example.com` a Route 53.

Este comando se ejecuta solo en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains check-domain-transferability \  
  --region us-east-1 \  
  --domain-name example.com
```

Salida:

```
{  
  "Transferability": {  
    "Transferable": "UNTRANSFERABLE"  
  }  
}
```

Para obtener más información, consulte [Transferir el registro de un dominio a Amazon Route 53](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [CheckDomainTransferability](#) la Referencia de AWS CLI comandos.

## **delete-tags-for-domain**

En el siguiente ejemplo de código, se muestra cómo usar `delete-tags-for-domain`.

### AWS CLI

Para eliminar las etiquetas de un dominio

El siguiente `delete-tags-for-domain` comando elimina tres etiquetas del dominio especificado. Tenga en cuenta que solo especifica la clave de la etiqueta, no el valor de la etiqueta.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains delete-tags-for-domain \  
  --region us-east-1 \  
  --domain-name example.com \  
  --tags-to-delete accounting-key hr-key engineering-key
```

Este comando no genera ninguna salida.

Para confirmar que se eliminaron las etiquetas, puede ejecutar [list-tags-for-domain](#). Para obtener más información, consulte [Etiquetado de los recursos de Amazon Route 53](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [DeleteTagsForDomain](#) la Referencia de AWS CLI comandos.

## **disable-domain-auto-renew**

En el siguiente ejemplo de código, se muestra cómo usar `disable-domain-auto-renew`.

### AWS CLI

Para deshabilitar la renovación automática de un dominio

El siguiente `disable-domain-auto-renew` comando configura Route 53 para que no renueve automáticamente el dominio `example.com` antes de que caduque el registro del dominio.

Este comando se ejecuta solo en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains disable-domain-auto-renew \  
  --region us-east-1 \  
  --domain-name example.com
```

Este comando no genera ninguna salida.

Para confirmar que se ha cambiado la configuración, puede ejecutar [get-domain-detail](#). Si la renovación automática está desactivada, el valor de `AutoRenew` es `False`. Para obtener más información sobre la renovación automática, consulte [Renovación del registro de un dominio < https://docs.aws.amazon.com/Route53/latest/DeveloperGuide /domain-renew.html](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/domain-renew.html) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte la Referencia de [DisableDomainAutoRenew](#) comandos AWS CLI .

## disable-domain-transfer-lock

En el siguiente ejemplo de código, se muestra cómo usar `disable-domain-transfer-lock`.

### AWS CLI

Para deshabilitar el bloqueo de transferencia en un dominio

El siguiente `disable-domain-transfer-lock` comando elimina el bloqueo de transferencia del dominio `example.com` para que el dominio se pueda transferir a otro registrador. Este comando cambia el `clientTransferProhibited` estado.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains disable-domain-transfer-lock \  
  --region us-east-1 \  
  --domain-name example.com
```

Salida:

```
{  
  "OperationId": "3f28e0ac-126a-4113-9048-cc930example"  
}
```

Para confirmar que se ha cambiado el bloqueo de transferencia, puedes ejecutar [get-domain-detail](#). Cuando el bloqueo de transferencia está desactivado, el valor de `StatusList` no se incluye `clientTransferProhibited`.

Para obtener más información sobre el proceso de transferencia, consulte [Transferencia de un dominio de Amazon Route 53 a otro registrador](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [DisableDomainTransferLock](#) la Referencia de AWS CLI comandos.

## enable-domain-auto-renew

En el siguiente ejemplo de código, se muestra cómo usar `enable-domain-auto-renew`.

## AWS CLI

Para habilitar la renovación automática de un dominio

El siguiente `enable-domain-auto-renew` comando configura Route 53 para que renueve automáticamente el dominio `example.com` antes de que caduque el registro del dominio.

Este comando se ejecuta solo en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains enable-domain-auto-renew \  
  --region us-east-1 \  
  --domain-name example.com
```

Este comando no genera ninguna salida. Para confirmar que se ha cambiado la configuración, puede ejecutar [get-domain-detail](#). Si la renovación automática está habilitada, el valor de `AutoRenew` es `True`.

Para obtener más información sobre la renovación automática, consulte [Renovación del registro de un dominio < https://docs.aws.amazon.com/Route53/latest/ DeveloperGuide /domain-renew.html](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/domain-renew.html) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte la Referencia de [EnableDomainAutoRenew](#) comandos AWS CLI .

## **enable-domain-transfer-lock**

En el siguiente ejemplo de código, se muestra cómo usar `enable-domain-transfer-lock`.

### AWS CLI

Para habilitar el bloqueo de transferencia en un dominio

El siguiente `enable-domain-transfer-lock` comando bloquea el dominio especificado para que no se pueda transferir a otro registrador. Este comando cambia el `clientTransferProhibited` estado.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains enable-domain-transfer-lock \  
  --region us-east-1 \  
  --domain-name example.com
```

```
--domain-name example.com
```

Salida:

```
{  
  "OperationId": "3f28e0ac-126a-4113-9048-cc930example"  
}
```

Para confirmar que se ha cambiado el bloqueo de transferencia, puedes ejecutar [get-domain-detail](#). Cuando el bloqueo de transferencia está activado, el valor de StatusList incluye `clientTransferProhibited`.

Para obtener más información sobre el proceso de transferencia, consulte [Transferencia de un dominio de Amazon Route 53 a otro registrador](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [EnableDomainTransferLock](#) la Referencia de AWS CLI comandos.

## get-contact-reachability-status

En el siguiente ejemplo de código, se muestra cómo usar `get-contact-reachability-status`.

AWS CLI

Para determinar si el contacto registrante ha respondido a un correo electrónico de confirmación

El siguiente `get-contact-reachability-status` comando devuelve información sobre si el contacto registrante del dominio especificado ha respondido a un correo electrónico de confirmación.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains get-contact-reachability-status \  
  --region us-east-1 \  
  --domain-name example.com
```

Salida:

```
{
```

```
"domainName": "example.com",  
"status": "DONE"  
}
```

Para obtener más información, consulte [Reenviar correos electrónicos de autorización y confirmación](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [GetContactReachabilityStatus](#) la Referencia de AWS CLI comandos.

## get-domain-detail

En el siguiente ejemplo de código, se muestra cómo usar `get-domain-detail`.

### AWS CLI

Para obtener información detallada sobre un dominio específico

El siguiente `get-domain-detail` comando muestra información detallada sobre el dominio especificado.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains get-domain-detail \  
  --region us-east-1 \  
  --domain-name example.com
```

Salida:

```
{  
  "DomainName": "example.com",  
  "Nameservers": [  
    {  
      "Name": "ns-2048.awsdns-64.com",  
      "GlueIps": []  
    },  
    {  
      "Name": "ns-2049.awsdns-65.net",  
      "GlueIps": []  
    },  
    {
```

```
        "Name": "ns-2050.awsdns-66.org",
        "GlueIps": []
    },
    {
        "Name": "ns-2051.awsdns-67.co.uk",
        "GlueIps": []
    }
],
"AutoRenew": true,
"AdminContact": {
    "FirstName": "Saanvi",
    "LastName": "Sarkar",
    "ContactType": "COMPANY",
    "OrganizationName": "Example",
    "AddressLine1": "123 Main Street",
    "City": "Anytown",
    "State": "WA",
    "CountryCode": "US",
    "ZipCode": "98101",
    "PhoneNumber": "+1.8005551212",
    "Email": "ssarkar@example.com",
    "ExtraParams": []
},
"RegistrantContact": {
    "FirstName": "Alejandro",
    "LastName": "Rosalez",
    "ContactType": "COMPANY",
    "OrganizationName": "Example",
    "AddressLine1": "123 Main Street",
    "City": "Anytown",
    "State": "WA",
    "CountryCode": "US",
    "ZipCode": "98101",
    "PhoneNumber": "+1.8005551212",
    "Email": "arosalez@example.com",
    "ExtraParams": []
},
"TechContact": {
    "FirstName": "Wang",
    "LastName": "Xiulan",
    "ContactType": "COMPANY",
    "OrganizationName": "Example",
    "AddressLine1": "123 Main Street",
    "City": "Anytown",
```

```

    "State": "WA",
    "CountryCode": "US",
    "ZipCode": "98101",
    "PhoneNumber": "+1.8005551212",
    "Email": "wxiulan@example.com",
    "ExtraParams": []
  },
  "AdminPrivacy": true,
  "RegistrantPrivacy": true,
  "TechPrivacy": true,
  "RegistrarName": "Amazon Registrar, Inc.",
  "WhoIsServer": "whois.registrar.amazon.com",
  "RegistrarUrl": "http://registrar.amazon.com",
  "AbuseContactEmail": "abuse@registrar.amazon.com",
  "AbuseContactPhone": "+1.2062661000",
  "CreationDate": 1444934889.601,
  "ExpirationDate": 1602787689.0,
  "StatusList": [
    "clientTransferProhibited"
  ]
}

```

- Para API obtener más información, consulte [GetDomainDetail](#) la Referencia de AWS CLI comandos.

## get-domain-suggestions

En el siguiente ejemplo de código, se muestra cómo usar `get-domain-suggestions`.

### AWS CLI

Para obtener una lista de nombres de dominio sugeridos

El siguiente `get-domain-suggestions` comando muestra una lista de nombres de dominio sugeridos en función del nombre de dominio `example.com`. La respuesta incluye solo los nombres de dominio que están disponibles. Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```

aws route53domains get-domain-suggestions \
  --region us-east-1 \
  --domain-name example.com \

```



```
--suggestion-count 10 \  
--only-available
```

Salida:

```
{  
  "SuggestionsList": [  
    {  
      "DomainName": "egzaampal.com",  
      "Availability": "AVAILABLE"  
    },  
    {  
      "DomainName": "examplelaw.com",  
      "Availability": "AVAILABLE"  
    },  
    {  
      "DomainName": "examplehouse.net",  
      "Availability": "AVAILABLE"  
    },  
    {  
      "DomainName": "homeexample.net",  
      "Availability": "AVAILABLE"  
    },  
    {  
      "DomainName": "examplelist.com",  
      "Availability": "AVAILABLE"  
    },  
    {  
      "DomainName": "examplenews.net",  
      "Availability": "AVAILABLE"  
    },  
    {  
      "DomainName": "officeexample.com",  
      "Availability": "AVAILABLE"  
    },  
    {  
      "DomainName": "exampleworld.com",  
      "Availability": "AVAILABLE"  
    },  
    {  
      "DomainName": "exampleart.com",  
      "Availability": "AVAILABLE"  
    }  
  ]  
}
```

```
]
}
```

- Para API obtener más información, consulte [GetDomainSuggestions](#) la Referencia de AWS CLI comandos.

## get-operation-detail

En el siguiente ejemplo de código, se muestra cómo usar `get-operation-detail`.

### AWS CLI

Para obtener el estado actual de una operación

Algunas operaciones de registro de dominios funcionan de forma asíncrona y devuelven una respuesta antes de que finalicen. Estas operaciones devuelven un identificador de operación que puede usar para obtener el estado actual. El siguiente `get-operation-detail` comando devuelve el estado de la operación especificada.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains get-operation-detail \
  --region us-east-1 \
  --operation-id edbd8d63-7fe7-4343-9bc5-54033example
```

Salida:

```
{
  "OperationId": "edbd8d63-7fe7-4343-9bc5-54033example",
  "Status": "SUCCESSFUL",
  "DomainName": "example.com",
  "Type": "DOMAIN_LOCK",
  "SubmittedDate": 1573749367.864
}
```

- Para API obtener más información, consulte [GetOperationDetail](#) la Referencia de AWS CLI comandos.

## list-domains

En el siguiente ejemplo de código, se muestra cómo usar `list-domains`.

### AWS CLI

Para ver una lista de los dominios que están registrados en la AWS cuenta corriente

El siguiente `list-domains` comando muestra información resumida sobre los dominios que están registrados en la AWS cuenta actual.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains list-domains
  --region us-east-1
```

Salida:

```
{
  "Domains": [
    {
      "DomainName": "example.com",
      "AutoRenew": true,
      "TransferLock": true,
      "Expiry": 1602712345.0
    },
    {
      "DomainName": "example.net",
      "AutoRenew": true,
      "TransferLock": true,
      "Expiry": 1602723456.0
    },
    {
      "DomainName": "example.org",
      "AutoRenew": true,
      "TransferLock": true,
      "Expiry": 1602734567.0
    }
  ]
}
```

- Para API obtener más información, consulte [ListDomains](#) la Referencia de AWS CLI comandos.

## list-operations

En el siguiente ejemplo de código, se muestra cómo usar `list-operations`.

### AWS CLI

Para ver el estado de las operaciones que devuelven un identificador de operación

Algunas operaciones de registro de dominios se ejecutan de forma asíncrona y devuelven una respuesta antes de que finalicen. Estas operaciones devuelven un identificador de operación que puede usar para obtener el estado actual. El siguiente `list-operations` comando muestra información resumida, incluido el estado, sobre las operaciones actuales de registro de dominios.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains list-operations
  --region us-east-1
```

Salida:

```
{
  "Operations": [
    {
      "OperationId": "aab9822f-1da0-4bf3-8a15-fd4e0example",
      "Status": "SUCCESSFUL",
      "Type": "DOMAIN_LOCK",
      "SubmittedDate": 1455321739.986
    },
    {
      "OperationId": "c24379ed-76be-42f8-bdad-9379bexample",
      "Status": "SUCCESSFUL",
      "Type": "UPDATE_NAMESERVER",
      "SubmittedDate": 1468960475.109
    },
    {
      "OperationId": "f47e1297-ef9e-4c2b-ae1e-a5fcbexample",
      "Status": "SUCCESSFUL",
      "Type": "RENEW_DOMAIN",
      "SubmittedDate": 1473561835.943
    },
    {
      "OperationId": "75584f23-b15f-459e-aed7-dc6f5example",
```

```

        "Status": "SUCCESSFUL",
        "Type": "UPDATE_DOMAIN_CONTACT",
        "SubmittedDate": 1547501003.41
    }
]
}

```

El resultado incluye todas las operaciones que devuelven un identificador de operación y que ha realizado en todos los dominios que haya registrado alguna vez con la AWS cuenta corriente. Si desea obtener solo las operaciones que envió después de una fecha específica, puede incluir el `submitted-since` parámetro y especificar una fecha en formato Unix y en horario universal coordinado (UTC). El siguiente comando obtiene el estado de todas las operaciones que se enviaron después de las 12:00 horas del 1 de UTC enero de 2020.

```

aws route53domains list-operations \
  --submitted-since 1577836800

```

- Para API obtener más información, consulte [ListOperations](#) la Referencia de AWS CLI comandos.

## list-tags-for-domain

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-domain`.

### AWS CLI

Para enumerar las etiquetas de un dominio

El siguiente `list-tags-for-domain` comando muestra las etiquetas que están asociadas actualmente al dominio especificado.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```

aws route53domains list-tags-for-domain \
  --region us-east-1 \
  --domain-name example.com

```

Salida:

```
{
```

```
"TagList": [  
  {  
    "Key": "key1",  
    "Value": "value1"  
  },  
  {  
    "Key": "key2",  
    "Value": "value2"  
  }  
]  
}
```

Para obtener más información, consulte [Etiquetado de los recursos de Amazon Route 53](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [ListTagsForDomain](#) la Referencia de AWS CLI comandos.

## register-domain

En el siguiente ejemplo de código, se muestra cómo usar `register-domain`.

### AWS CLI

Para registrar un dominio

El siguiente `register-domain` comando registra un dominio y recupera todos los valores de los parámetros de un archivo JSON con formato.

Este comando solo se ejecuta en la región. `us-east-1` Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains register-domain \  
  --region us-east-1 \  
  --cli-input-json file://register-domain.json
```

Contenidos de `register-domain.json`:

```
{  
  "DomainName": "example.com",  
  "DurationInYears": 1,  
  "AutoRenew": true,
```

```
"AdminContact": {
  "FirstName": "Martha",
  "LastName": "Rivera",
  "ContactType": "PERSON",
  "OrganizationName": "Example",
  "AddressLine1": "1 Main Street",
  "City": "Anytown",
  "State": "WA",
  "CountryCode": "US",
  "ZipCode": "98101",
  "PhoneNumber": "+1.8005551212",
  "Email": "mrivera@example.com"
},
"RegistrantContact": {
  "FirstName": "Li",
  "LastName": "Juan",
  "ContactType": "PERSON",
  "OrganizationName": "Example",
  "AddressLine1": "1 Main Street",
  "City": "Anytown",
  "State": "WA",
  "CountryCode": "US",
  "ZipCode": "98101",
  "PhoneNumber": "+1.8005551212",
  "Email": "ljuan@example.com"
},
"TechContact": {
  "FirstName": "Mateo",
  "LastName": "Jackson",
  "ContactType": "PERSON",
  "OrganizationName": "Example",
  "AddressLine1": "1 Main Street",
  "City": "Anytown",
  "State": "WA",
  "CountryCode": "US",
  "ZipCode": "98101",
  "PhoneNumber": "+1.8005551212",
  "Email": "mjackson@example.com"
},
"PrivacyProtectAdminContact": true,
"PrivacyProtectRegistrantContact": true,
"PrivacyProtectTechContact": true
}
```

Salida:

```
{
  "OperationId": "b114c44a-9330-47d1-a6e8-a0b11example"
}
```

Para confirmar que la operación se realizó correctamente, puede ejecutar `target-operation-detail`. Para obtener más información, consulte [get-operation-detail](#).

Para obtener más información, consulte [Renovación de un nuevo dominio](#) en la guía para desarrolladores de Amazon Route 53.

Para obtener información sobre los dominios de nivel superior (TLDs) que requieren valores `ExtraParams` y cuáles son los valores válidos, consulte [ExtraParam](#) la API referencia de Amazon Route 53.

- Para API obtener más información, consulte [RegisterDomain](#) la Referencia de AWS CLI comandos.

## **renew-domain**

En el siguiente ejemplo de código, se muestra cómo usar `renew-domain`.

### AWS CLI

Para renovar un dominio

El siguiente `renew-domain` comando renueva el dominio especificado durante cinco años. Para obtener el valor `current-expiry-year`, utilice el `get-domain-detail` comando y convierta el valor de `ExpirationDate` desde el formato Unix.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains renew-domain \
  --region us-east-1 \
  --domain-name example.com \
  --duration-in-years 5 \
  --current-expiry-year 2020
```

Salida:



```
{
  "OperationId": "3f28e0ac-126a-4113-9048-cc930example"
}
```

Para confirmar que la operación se realizó correctamente, puede ejecutar `target-operation-detail`. Para obtener más información, consulte [get-operation-detail](#).

El registro de cada dominio de nivel superior (TLD), como `.com` o `.org`, controla el número máximo de años durante los que se puede renovar un dominio. Para obtener el período máximo de renovación de su dominio, consulte la sección «Periodo de registro y renovación» correspondiente a [los dominios que puede registrar TLD en Amazon Route 53](#) en la Guía para desarrolladores de Amazon Route 53.

Para obtener más información, consulte [Renovación del registro de un dominio en la Guía para desarrolladores de Amazon Route 53](#).

- Para API obtener más información, consulte [RenewDomain](#) la Referencia de AWS CLI comandos.

## **resend-contact-reachability-email**

En el siguiente ejemplo de código, se muestra cómo usar `resend-contact-reachability-email`.

### AWS CLI

Para volver a enviar el correo electrónico de confirmación a la dirección de correo electrónico actual del registrante, póngase en contacto con

El siguiente `resend-contact-reachability-email` comando reenvía el correo electrónico de confirmación a la dirección de correo electrónico actual del contacto del registrante del dominio `example.com`.

Este comando solo se ejecuta en la región `us-east-1`. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains resend-contact-reachability-email \
  --region us-east-1 \
  --domain-name example.com
```

Salida:

```
{
  "domainName": "example.com",
  "emailAddress": "moliveira@example.com",
  "isAlreadyVerified": true
}
```

Si el valor de `isAlreadyVerified` es `true`, como en este ejemplo, el contacto registrante ya ha confirmado que se puede acceder a la dirección de correo electrónico especificada.

Para obtener más información, consulte [Reenviar correos electrónicos de autorización y confirmación](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [ResendContactReachabilityEmail](#) la Referencia de AWS CLI comandos.

## retrieve-domain-auth-code

En el siguiente ejemplo de código, se muestra cómo usar `retrieve-domain-auth-code`.

### AWS CLI

Para obtener el código de autorización de un dominio para poder transferirlo a otro registrador

El siguiente `retrieve-domain-auth-code` comando obtiene el código de autorización actual del dominio `example.com`. Le das este valor a otro registrador de dominios cuando deseas transferir el dominio a ese registrador.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains retrieve-domain-auth-code \
  --region us-east-1 \
  --domain-name example.com
```

Salida:

```
{
  "AuthCode": ")o!v3dJeXampLe"
}
```

Para obtener más información, consulte [Transferencia de un dominio de Amazon Route 53 a otro registrador](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [RetrieveDomainAuthCode](#) la Referencia de AWS CLI comandos.

## transfer-domain

En el siguiente ejemplo de código, se muestra cómo usar `transfer-domain`.

### AWS CLI

Para transferir un dominio a Amazon Route 53

El siguiente `transfer-domain` comando transfiere un dominio a Route 53, con los parámetros proporcionados por el archivo con JSON formato. `C:\temp\transfer-domain.json`

Este comando se ejecuta solo en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains transfer-domain \  
  --region us-east-1 \  
  --cli-input-json file://C:\temp\transfer-domain.json
```

Contenidos de `transfer-domain.json`:

```
{  
  "DomainName": "example.com",  
  "DurationInYears": 1,  
  "Nameservers": [  
    {  
      "Name": "ns-2048.awsdns-64.com"  
    },  
    {  
      "Name": "ns-2049.awsdns-65.net"  
    },  
    {  
      "Name": "ns-2050.awsdns-66.org"  
    },  
    {  
      "Name": "ns-2051.awsdns-67.co.uk"  
    }  
  ],  
}
```

```
"AuthCode": ")o!v3dJeXampLe",
"AutoRenew": true,
"AdminContact": {
  "FirstName": "Martha",
  "LastName": "Rivera",
  "ContactType": "PERSON",
  "OrganizationName": "Example",
  "AddressLine1": "1 Main Street",
  "City": "Anytown",
  "State": "WA",
  "CountryCode": "US",
  "ZipCode": "98101",
  "PhoneNumber": "+1.8005551212",
  "Email": "mrivera@example.com"
},
"RegistrantContact": {
  "FirstName": "Li",
  "LastName": "Juan",
  "ContactType": "PERSON",
  "OrganizationName": "Example",
  "AddressLine1": "1 Main Street",
  "City": "Anytown",
  "State": "WA",
  "CountryCode": "US",
  "ZipCode": "98101",
  "PhoneNumber": "+1.8005551212",
  "Email": "ljuan@example.com"
},
"TechContact": {
  "FirstName": "Mateo",
  "LastName": "Jackson",
  "ContactType": "PERSON",
  "OrganizationName": "Example",
  "AddressLine1": "1 Main Street",
  "City": "Anytown",
  "State": "WA",
  "CountryCode": "US",
  "ZipCode": "98101",
  "PhoneNumber": "+1.8005551212",
  "Email": "mjackson@example.com"
},
"PrivacyProtectAdminContact": true,
"PrivacyProtectRegistrantContact": true,
"PrivacyProtectTechContact": true
```

```
}
```

Salida:

```
{  
  "OperationId": "b114c44a-9330-47d1-a6e8-a0b11example"  
}
```

Para confirmar que la operación se realizó correctamente, puede ejecutar `target-operation-detail`. Para obtener más información, consulte [get-operation-detail](#).

Para obtener más información, consulte [Transferir el registro de un dominio a Amazon Route 53](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [TransferDomain](#) la Referencia de AWS CLI comandos.

## update-domain-contact-privacy

En el siguiente ejemplo de código, se muestra cómo usar `update-domain-contact-privacy`.

AWS CLI

Para actualizar la configuración de privacidad de los contactos de un dominio

El siguiente `update-domain-contact-privacy` comando desactiva la protección de la privacidad del contacto administrativo del dominio `example.com`. Este comando solo se ejecuta en la `us-east-1` región.

Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains update-domain-contact-privacy \  
  --region us-east-1 \  
  --domain-name example.com \  
  --no-admin-privacy
```

Salida:

```
{  
  "OperationId": "b3a219e9-d801-4244-b533-b7256example"  
}
```

Para confirmar que la operación se realizó correctamente, puede ejecutar `target-operation-detail`. Para obtener más información, consulte [get-operation-detail](#).

Para obtener más información, consulte [Habilitar o deshabilitar la protección de la privacidad de la información de contacto de un dominio en la Guía para desarrolladores de Amazon Route 53](#).

- Para API obtener más información, consulte [UpdateDomainContactPrivacy](#) la Referencia de AWS CLI comandos.

## update-domain-contact

En el siguiente ejemplo de código, se muestra cómo usar `update-domain-contact`.

### AWS CLI

Para actualizar la información de contacto de un dominio

El siguiente `update-domain-contact` comando actualiza la información de contacto de un dominio y obtiene los parámetros del archivo JSON con formato. `C:\temp\update-domain-contact.json`

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains update-domain-contact \  
  --region us-east-1 \  
  --cli-input-json file://C:\temp\update-domain-contact.json
```

Contenidos de `update-domain-contact.json`:

```
{  
  "AdminContact": {  
    "AddressLine1": "101 Main Street",  
    "AddressLine2": "Suite 1a",  
    "City": "Seattle",  
    "ContactType": "COMPANY",  
    "CountryCode": "US",  
    "Email": "w.xiulan@example.com",  
    "FirstName": "Wang",  
    "LastName": "Xiulan",  
    "OrganizationName": "Example",  
    "PhoneNumber": "+1.8005551212",
```

```
    "State": "WA",
    "ZipCode": "98101"
  },
  "DomainName": "example.com",
  "RegistrantContact": {
    "AddressLine1": "101 Main Street",
    "AddressLine2": "Suite 1a",
    "City": "Seattle",
    "ContactType": "COMPANY",
    "CountryCode": "US",
    "Email": "w.xiulan@example.com",
    "FirstName": "Wang",
    "LastName": "Xiulan",
    "OrganizationName": "Example",
    "PhoneNumber": "+1.8005551212",
    "State": "WA",
    "ZipCode": "98101"
  },
  "TechContact": {
    "AddressLine1": "101 Main Street",
    "AddressLine2": "Suite 1a",
    "City": "Seattle",
    "ContactType": "COMPANY",
    "CountryCode": "US",
    "Email": "w.xiulan@example.com",
    "FirstName": "Wang",
    "LastName": "Xiulan",
    "OrganizationName": "Example",
    "PhoneNumber": "+1.8005551212",
    "State": "WA",
    "ZipCode": "98101"
  }
}
```

Salida:

```
{
  "OperationId": "b3a219e9-d801-4244-b533-b7256example"
}
```

Para confirmar que la operación se realizó correctamente, puede ejecutar [get-domain-detail](#). Para obtener más información, consulte [Actualización de la información de contacto de un dominio en la Guía para desarrolladores de Amazon Route 53](#).

- Para API obtener más información, consulte [UpdateDomainContact](#) la Referencia de AWS CLI comandos.

## update-domain-nameservers

En el siguiente ejemplo de código, se muestra cómo usar `update-domain-nameservers`.

### AWS CLI

Para actualizar los servidores de nombres de un dominio

El siguiente `update-domain-nameservers` comando actualiza los servidores de nombres de un dominio.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains update-domain-nameservers \
  --region us-east-1 \
  --domain-name example.com \
  --
nameservers Name=ns-1.awsdns-01.org Name=ns-2.awsdns-02.co.uk Name=ns-3.awsdns-03.net Name=ns-4.awsdns-04.com
```

Salida:

```
{
  "OperationId": "f1691ec4-0e7a-489e-82e0-b19d3example"
}
```

Para confirmar que la operación se realizó correctamente, puede ejecutar [get-domain-detail](#).

Para obtener más información, consulte [Añadir o cambiar servidores de nombres y registros de Glue para un dominio](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [UpdateDomainNameservers](#) la Referencia de AWS CLI comandos.

## update-tags-for-domain

En el siguiente ejemplo de código, se muestra cómo usar `update-tags-for-domain`.



## AWS CLI

Para añadir o actualizar etiquetas para un dominio

El siguiente `update-tags-for-domain` comando agrega o actualiza dos claves y los valores correspondientes para el dominio `example.com`. Para actualizar el valor de una clave, simplemente incluye la clave y el nuevo valor. Solo puedes añadir o actualizar etiquetas en un dominio a la vez.

Este comando solo se ejecuta en la `us-east-1` región. Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains update-tags-for-domain \  
  --region us-east-1 \  
  --domain-name example.com \  
  --tags-to-update "Key=key1,Value=value1" "Key=key2,Value=value2"
```

Este comando no genera ninguna salida. Para confirmar que las etiquetas se han añadido o actualizado, puede ejecutar [list-tags-for-domain](#).

Para obtener más información, consulte [Etiquetado de los recursos de Amazon Route 53](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [UpdateTagsForDomain](#) la Referencia de AWS CLI comandos.

## view-billing

En el siguiente ejemplo de código, se muestra cómo usar `view-billing`.

## AWS CLI

Para obtener la información de facturación de los cargos de registro de dominio de la AWS cuenta corriente

El siguiente `view-billing` comando devuelve todos los registros de facturación relacionados con el dominio de la cuenta corriente durante el período comprendido entre el 1 de enero de 2018 (1514764800 en horario Unix) y la medianoche del 31 de diciembre de 2019 (1577836800 en horario Unix).

Este comando solo se ejecuta en la región. `us-east-1` Si su región predeterminada está establecida en `us-east-1`, puede omitir el `region` parámetro.

```
aws route53domains view-billing \  
  --region us-east-1 \  
  --start-time 1514764800 \  
  --end-time 1577836800
```

Salida:

```
{  
  "BillingRecords": [  
    {  
      "DomainName": "example.com",  
      "Operation": "RENEW_DOMAIN",  
      "InvoiceId": "149962827",  
      "BillDate": 1536618063.181,  
      "Price": 12.0  
    },  
    {  
      "DomainName": "example.com",  
      "Operation": "RENEW_DOMAIN",  
      "InvoiceId": "290913289",  
      "BillDate": 1568162630.884,  
      "Price": 12.0  
    }  
  ]  
}
```

Para obtener más información, consulte [ViewBilling](#) la API referencia de Amazon Route 53.

- Para API obtener más información, consulte [ViewBilling](#) la Referencia de AWS CLI comandos.

## Ejemplos de Route 53 Resolver usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Route 53 Resolver.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **associate-firewall-rule-group**

En el siguiente ejemplo de código, se muestra cómo usar `associate-firewall-rule-group`.

#### AWS CLI

Para asociar un grupo de reglas de firewall a un VPC

En el siguiente `associate-firewall-rule-group` ejemplo, se asocia un grupo de reglas de DNS Firewall a un AmazonVPC.

```
aws route53resolver associate-firewall-rule-group \  
  --name test-association \  
  --firewall-rule-group-id rslvr-frg-47f93271fexample \  
  --vpc-id vpc-31e92222 \  
  --priority 101
```

Salida:

```
{  
  "FirewallRuleGroupAssociation": {  
    "Id": "rslvr-frgassoc-57e8873d7example",  
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-  
association/rslvr-frgassoc-57e8873d7example",  
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",  
    "VpcId": "vpc-31e92222",  
    "Name": "test-association",  
    "Priority": 101,  
    "MutationProtection": "DISABLED",  
    "Status": "UPDATING",  
    "StatusMessage": "Creating Firewall Rule Group Association",  
    "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",  
    "CreationTime": "2021-05-25T21:47:48.755768Z",  
    "ModificationTime": "2021-05-25T21:47:48.755768Z"  
  }  
}
```

Para obtener más información, consulte [Administrar las asociaciones entre sus grupos de reglas de Route 53 Resolver DNS Firewall VPC](#) y los de Route 53 Resolver en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [AssociateFirewallRuleGroup](#) la Referencia de AWS CLI comandos.

## **associate-resolver-endpoint-ip-address**

En el siguiente ejemplo de código, se muestra cómo usar `associate-resolver-endpoint-ip-address`.

### AWS CLI

Para asociar otra dirección IP a un punto final del Resolver

El siguiente `associate-resolver-endpoint-ip-address` ejemplo asocia otra dirección IP a un punto final de Resolver entrante. Si especifica solo un identificador de subred y omite la dirección IP del `--ip-address` parámetro, Resolver elegirá una dirección IP de entre las direcciones IP disponibles en la subred especificada.

```
aws route53resolver associate-resolver-endpoint-ip-address \  
  --resolver-endpoint-id rslvr-in-497098ad5example \  
  --ip-address="SubnetId=subnet-12d8exam,Ip=192.0.2.118"
```

Salida:

```
{  
  "ResolverEndpoint": {  
    "Id": "rslvr-in-497098ad5example",  
    "CreatorRequestId": "AWSConsole.25.0123456789",  
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/  
rslvr-in-497098ad5example",  
    "Name": "my-inbound-endpoint",  
    "SecurityGroupIds": [  
      "sg-05cd7b25d6example"  
    ],  
    "Direction": "INBOUND",  
    "IpAddressCount": 3,  
    "HostVPCId": "vpc-304bexam",  
    "Status": "UPDATING",
```

```

    "StatusMessage": "Updating the Resolver Endpoint",
    "CreationTime": "2020-01-02T23:25:45.538Z",
    "ModificationTime": "2020-01-02T23:25:45.538Z"
  }
}

```

Para obtener más información, consulte [los valores que se especifican al crear o editar puntos de enlace entrantes](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte la Referencia [AssociateResolverEndpointIpAddress](#) de AWS CLI comandos.

## associate-resolver-rule

En el siguiente ejemplo de código, se muestra cómo usar `associate-resolver-rule`.

### AWS CLI

Para asociar una regla de resolución a un VPC

En el siguiente `associate-resolver-rule` ejemplo, se asocia una regla de resolución a una de AmazonVPC. Tras ejecutar el comando, Resolver comienza a reenviar DNS las consultas a la red en función de la configuración de la regla, como el nombre de dominio de las consultas que se reenvían.

```

aws route53resolver associate-resolver-rule \
  --name my-resolver-rule-association \
  --resolver-rule-id rslvr-rr-42b60677c0example \
  --vpc-id vpc-304bexam

```

Salida:

```

{
  "ResolverRuleAssociation": {
    "Id": "rslvr-rrassoc-d61cbb2c8bexample",
    "ResolverRuleId": "rslvr-rr-42b60677c0example",
    "Name": "my-resolver-rule-association",
    "VPCId": "vpc-304bexam",
    "Status": "CREATING",
    "StatusMessage": "[Trace id: 1-5dc5a8fa-ec2cc480d2ef07617example] Creating
the association."
  }
}

```

```
}
```

Para obtener más información, consulte [Reenviar DNS consultas salientes a su red](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte la Referencia [AssociateResolverRule](#) de AWS CLI comandos.

## **create-firewall-domain-list**

En el siguiente ejemplo de código, se muestra cómo usar `create-firewall-domain-list`.

### AWS CLI

Para crear una lista de dominios de Route 53 Resolver DNS Firewall

En el siguiente `create-firewall-domain-list` ejemplo, se crea una lista de dominios de Route 53 Resolver DNS Firewall, denominada `test`, en su AWS cuenta.

```
aws route53resolver create-firewall-domain-list \  
  --creator-request-id my-request-id \  
  --name test
```

Salida:

```
{  
  "FirewallDomainList": {  
    "Id": "rslvr-fdl-d61cbb2cbexample",  
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-domain-list/  
rslvr-fdl-d61cbb2cbexample",  
    "Name": "test",  
    "DomainCount": 0,  
    "Status": "COMPLETE",  
    "StatusMessage": "Created Firewall Domain List",  
    "CreatorRequestId": "my-request-id",  
    "CreationTime": "2021-05-25T15:55:51.115365Z",  
    "ModificationTime": "2021-05-25T15:55:51.115365Z"  
  }  
}
```

Para obtener más información, consulte [Administrar sus propias listas de dominios](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [CreateFirewallDomainList](#) la Referencia de AWS CLI comandos.

## create-firewall-rule-group

En el siguiente ejemplo de código, se muestra cómo usar `create-firewall-rule-group`.

### AWS CLI

Para crear un grupo de reglas de firewall

En el siguiente `create-firewall-rule-group` ejemplo, se crea un grupo de reglas de DNS firewall.

```
aws route53resolver create-firewall-rule-group \  
  --creator-request-id my-request-id \  
  --name test
```

Salida:

```
{  
  "FirewallRuleGroup": {  
    "Id": "rslvr-frg-47f93271fexample",  
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group/  
rslvr-frg-47f93271fexample",  
    "Name": "test",  
    "RuleCount": 0,  
    "Status": "COMPLETE",  
    "StatusMessage": "Created Firewall Rule Group",  
    "OwnerId": "123456789012",  
    "CreatorRequestId": "my-request-id",  
    "ShareStatus": "NOT_SHARED",  
    "CreationTime": "2021-05-25T18:59:26.490017Z",  
    "ModificationTime": "2021-05-25T18:59:26.490017Z"  
  }  
}
```

Para obtener más información, consulte [Administrar grupos de reglas y reglas en DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [CreateFirewallRuleGroup](#) la Referencia de AWS CLI comandos.

## create-firewall-rule

En el siguiente ejemplo de código, se muestra cómo usar `create-firewall-rule`.

### AWS CLI

Para crear una regla de firewall

El siguiente `create-firewall-rule` ejemplo crea una regla de firewall en una regla de DNS firewall para los dominios incluidos en una lista de dominios de DNS firewall.

```
aws route53resolver create-firewall-rule \  
  --name allow-rule \  
  --firewall-rule-group-id rslvr-frg-47f93271fexample \  
  --firewall-domain-list-id rslvr-fdl-9e956e9ffexample \  
  --priority 101 \  
  --action ALLOW
```

Salida:

```
{  
  "FirewallRule": {  
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",  
    "FirewallDomainListId": "rslvr-fdl-9e956e9ffexample",  
    "Name": "allow-rule",  
    "Priority": 101,  
    "Action": "ALLOW",  
    "CreatorRequestId": "d81e3fb7-020b-415e-939f-EXAMPLE11111",  
    "CreationTime": "2021-05-25T21:44:00.346093Z",  
    "ModificationTime": "2021-05-25T21:44:00.346093Z"  
  }  
}
```

Para obtener más información, consulte [Administrar grupos de reglas y reglas en DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [CreateFirewallRule](#) la Referencia de AWS CLI comandos.

## create-resolver-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `create-resolver-endpoint`.



## AWS CLI

Para crear un punto final de Resolver entrante

En el siguiente `create-resolver-endpoint` ejemplo, se crea un punto final de Resolver entrante. Puede usar el mismo comando para crear puntos finales entrantes y salientes.

```
aws route53resolver create-resolver-endpoint --name my-inbound-endpoint -- creator-request-id 2020-01-01-18:47 -- «sg-f62bexam» --direction --ip-addresses =subnet-ba47 exam, Ip=192.0.2.255 =subnet-12d8exam, security-group-ids Ip=192.0.2.254 INBOUND SubnetId SubnetId
```

Salida:

```
{
  "ResolverEndpoint": {
    "Id": "rslvr-in-f9ab8a03f1example",
    "CreatorRequestId": "2020-01-01-18:47",
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/rslvr-in-f9ab8a03f1example",
    "Name": "my-inbound-endpoint",
    "SecurityGroupIds": [
      "sg-f62bexam"
    ],
    "Direction": "INBOUND",
    "IpAddressCount": 2,
    "HostVPCId": "vpc-304examp",
    "Status": "CREATING",
    "StatusMessage": "[Trace id: 1-5dc1ff84-f3477826e4a190025example] Creating the Resolver Endpoint",
    "CreationTime": "2020-01-01T23:02:29.583Z",
    "ModificationTime": "2020-01-01T23:02:29.583Z"
  }
}
```

Para crear un punto final de Resolver saliente

En el siguiente `create-resolver-endpoint` ejemplo, se crea un punto final de resolución saliente utilizando los valores del documento con formato JSON. `create-outbound-resolver-endpoint.json`

```
aws route53resolver create-resolver-endpoint \
```

```
--cli-input-json file://c:\temp\create-outbound-resolver-endpoint.json
```

Contenidos de `create-outbound-resolver-endpoint.json`:

```
{
  "CreatorRequestId": "2020-01-01-18:47",
  "Direction": "OUTBOUND",
  "IpAddresses": [
    {
      "Ip": "192.0.2.255",
      "SubnetId": "subnet-ba47exam"
    },
    {
      "Ip": "192.0.2.254",
      "SubnetId": "subnet-12d8exam"
    }
  ],
  "Name": "my-outbound-endpoint",
  "SecurityGroupIds": [ "sg-05cd7b25d6example" ],
  "Tags": [
    {
      "Key": "my-key-name",
      "Value": "my-key-value"
    }
  ]
}
```

Para obtener más información, consulte [Resolución de DNS consultas entre VPCs y su red](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [CreateResolverEndpoint](#) la Referencia de AWS CLI comandos.

## **create-resolver-rule**

En el siguiente ejemplo de código, se muestra cómo usar `create-resolver-rule`.

AWS CLI

Para crear una regla de resolución

En el siguiente `create-resolver-rule` ejemplo, se crea una regla de reenvío de Resolver. La regla utiliza el `rslvr-out-d` ejemplo del extremo saliente `5e5920e37` para reenviar las DNS consultas a las direcciones IP `10.24.8.75` y `10.24.8.156`. `example.com`

```
aws route53resolver create-resolver-rule \  
  --creator-request-id 2020-01-02-18:47 \  
  --domain-name example.com \  
  --name my-rule \  
  --resolver-endpoint-id rslvr-out-d5e5920e37example \  
  --rule-type FORWARD \  
  --target-ips "Ip=10.24.8.75" "Ip=10.24.8.156"
```

Salida:

```
{  
  "ResolverRule": {  
    "Status": "COMPLETE",  
    "RuleType": "FORWARD",  
    "ResolverEndpointId": "rslvr-out-d5e5920e37example",  
    "Name": "my-rule",  
    "DomainName": "example.com.",  
    "CreationTime": "2022-05-10T21:35:30.923187Z",  
    "TargetIps": [  
      {  
        "Ip": "10.24.8.75",  
        "Port": 53  
      },  
      {  
        "Ip": "10.24.8.156",  
        "Port": 53  
      }  
    ],  
    "CreatorRequestId": "2022-05-10-16:33",  
    "ModificationTime": "2022-05-10T21:35:30.923187Z",  
    "ShareStatus": "NOT_SHARED",  
    "Arn": "arn:aws:route53resolver:us-east-1:111117012054:resolver-rule/rslvr-  
rr-b1e0b905e93611111",  
    "OwnerId": "111111111111",  
    "Id": "rslvr-rr-rslvr-rr-b1e0b905e93611111",  
    "StatusMessage": "[Trace id: 1-22222222-3e56afcc71a3724664f22e24]  
Successfully created Resolver Rule."  
  }  
}
```

```
}
```

- API Para [CreateResolverRule](#) obtener más AWS CLI información, consulte la Referencia de comandos.

## delete-firewall-domain-list

En el siguiente ejemplo de código, se muestra cómo usar `delete-firewall-domain-list`.

### AWS CLI

Para eliminar una lista de dominios de Route 53 Resolver DNS Firewall

En el siguiente `delete-firewall-domain-list` ejemplo, se elimina una lista de dominios de Route 53 Resolver DNS Firewall, denominada `test`, de su AWS cuenta.

```
aws route53resolver delete-firewall-domain-list \  
--firewall-domain-list-id rslvr-fdl-9e956e9ffexample
```

Salida:

```
{  
  "FirewallDomainList": {  
    "Id": "rslvr-fdl-9e956e9ffexample",  
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-domain-list/  
rslvr-fdl-9e956e9ffexample",  
    "Name": "test",  
    "DomainCount": 6,  
    "Status": "DELETING",  
    "StatusMessage": "Deleting the Firewall Domain List",  
    "CreatorRequestId": "my-request-id",  
    "CreationTime": "2021-05-25T15:55:51.115365Z",  
    "ModificationTime": "2021-05-25T18:58:05.588024Z"  
  }  
}
```

Para obtener más información, consulte [Administrar sus propias listas de dominios](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [DeleteFirewallDomainList](#) la Referencia de AWS CLI comandos.

## delete-firewall-rule-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-firewall-rule-group`.

### AWS CLI

Para eliminar un grupo de reglas de firewall

En el siguiente `delete-firewall-rule-group` ejemplo, se elimina un grupo de reglas de firewall.

```
aws route53resolver delete-firewall-rule-group \  
--firewall-rule-group-id rslvr-frg-47f93271fexample
```

Salida:

```
{  
  "FirewallRuleGroup": {  
    "Id": "rslvr-frg-47f93271fexample",  
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group/  
rslvr-frg-47f93271fexample",  
    "Name": "test",  
    "RuleCount": 0,  
    "Status": "UPDATING",  
    "StatusMessage": "Updating Firewall Rule Group",  
    "OwnerId": "123456789012",  
    "CreatorRequestId": "my-request-id",  
    "ShareStatus": "NOT_SHARED",  
    "CreationTime": "2021-05-25T18:59:26.490017Z",  
    "ModificationTime": "2021-05-25T21:51:53.028688Z"  
  }  
}
```

Para obtener más información, consulte [Administrar grupos de reglas y reglas en DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [DeleteFirewallRuleGroup](#) la Referencia de AWS CLI comandos.

## delete-firewall-rule

En el siguiente ejemplo de código, se muestra cómo usar `delete-firewall-rule`.

## AWS CLI

Para eliminar una regla de firewall

En el siguiente `delete-firewall-rule` ejemplo, se elimina una regla de firewall especificada.

```
aws route53resolver delete-firewall-rule \  
  --firewall-rule-group-id rslvr-frg-47f93271fexample \  
  --firewall-domain-list-id rslvr-fdl-9e956e9ffexample
```

Salida:

```
{  
  "FirewallRule": {  
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",  
    "FirewallDomainListId": "rslvr-fdl-9e956e9ffexample",  
    "Name": "allow-rule",  
    "Priority": 102,  
    "Action": "ALLOW",  
    "CreatorRequestId": "d81e3fb7-020b-415e-939f-EXAMPLE11111",  
    "CreationTime": "2021-05-25T21:44:00.346093Z",  
    "ModificationTime": "2021-05-25T21:45:59.611600Z"  
  }  
}
```

Para obtener más información, consulte [Administrar grupos de reglas y reglas en DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [DeleteFirewallRule](#) la Referencia de AWS CLI comandos.

## `delete-resolver-endpoint`

En el siguiente ejemplo de código, se muestra cómo usar `delete-resolver-endpoint`.

## AWS CLI

Para eliminar un punto final de Resolver

En el siguiente `delete-resolver-endpoint` ejemplo, se elimina el punto final especificado.

Importante: si elimina un punto final entrante, DNS las consultas de su red ya no se reenvían a Resolver en la dirección VPC que especificó en el punto final. Si elimina un punto final de salida,

Resolver deja de reenviar DNS las consultas de la suya VPC a la red en busca de reglas que especifiquen el punto final de salida eliminado.

```
aws route53resolver delete-resolver-endpoint \  
--resolver-endpoint-id rslvr-in-497098ad59example
```

Salida:

```
{  
  "ResolverEndpoint": {  
    "Id": "rslvr-in-497098ad59example",  
    "CreatorRequestId": "AWSConsole.25.157290example",  
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/  
rslvr-in-497098ad59example",  
    "Name": "my-inbound-endpoint",  
    "SecurityGroupIds": [  
      "sg-05cd7b25d6example"  
    ],  
    "Direction": "INBOUND",  
    "IpAddressCount": 5,  
    "HostVPCId": "vpc-304bexam",  
    "Status": "DELETING",  
    "StatusMessage": "[Trace id: 1-5dc5b658-811b5be0922bbc382example] Deleting  
ResolverEndpoint.",  
    "CreationTime": "2020-01-01T23:25:45.538Z",  
    "ModificationTime": "2020-01-02T23:25:45.538Z"  
  }  
}
```

- Para API obtener más información, consulte la Referencia de [DeleteResolverEndpoint](#) comandos AWS CLI .

## delete-resolver-rule

En el siguiente ejemplo de código, se muestra cómo usar `delete-resolver-rule`.

AWS CLI

Para eliminar una regla de resolución

En el siguiente `delete-resolver-rule` ejemplo, se elimina la regla especificada.

Nota: Si una regla está asociada a alguna VPCs, primero debe desasociarla de la regla para VPCs poder eliminarla.

```
aws route53resolver delete-resolver-rule \  
--resolver-rule-id rslvr-rr-5b3809426bexample
```

Salida:

```
{  
  "ResolverRule": {  
    "Id": "rslvr-rr-5b3809426bexample",  
    "CreatorRequestId": "2020-01-03-18:47",  
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/rslvr-  
rr-5b3809426bexample",  
    "DomainName": "zenith.example.com.",  
    "Status": "DELETING",  
    "StatusMessage": "[Trace id: 1-5dc5e05b-602e67b052cb74f05example] Deleting  
Resolver Rule.",  
    "RuleType": "FORWARD",  
    "Name": "my-resolver-rule",  
    "TargetIps": [  
      {  
        "Ip": "192.0.2.50",  
        "Port": 53  
      }  
    ],  
    "ResolverEndpointId": "rslvr-out-d5e5920e3example",  
    "OwnerId": "111122223333",  
    "ShareStatus": "NOT_SHARED"  
  }  
}
```

- Para API obtener más información, consulte [DeleteResolverRule](#) la Referencia de AWS CLI comandos.

## **disassociate-firewall-rule-group**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-firewall-rule-group`.



## AWS CLI

Para desasociar un grupo de reglas de firewall de un VPC

El siguiente `disassociate-firewall-rule-group` ejemplo desasocia un grupo de reglas de DNS Firewall de AmazonVPC.

```
aws route53resolver disassociate-firewall-rule-group \  
  --firewall-rule-group-association-id rslvr-frgassoc-57e8873d7example
```

Salida:

```
{  
  "FirewallRuleGroupAssociation": {  
    "Id": "rslvr-frgassoc-57e8873d7example",  
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-  
association/rslvr-frgassoc-57e8873d7example",  
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",  
    "VpcId": "vpc-31e92222",  
    "Name": "test-association",  
    "Priority": 103,  
    "MutationProtection": "DISABLED",  
    "Status": "DELETING",  
    "StatusMessage": "Deleting the Firewall Rule Group Association",  
    "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",  
    "CreationTime": "2021-05-25T21:47:48.755768Z",  
    "ModificationTime": "2021-05-25T21:51:02.377887Z"  
  }  
}
```

Para obtener más información, consulte [Administrar las asociaciones entre sus grupos de reglas de Route 53 Resolver DNS Firewall VPC](#) y los de Route 53 Resolver en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [DisassociateFirewallRuleGroup](#) la Referencia de AWS CLI comandos.

### **disassociate-resolver-endpoint-ip-address**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-resolver-endpoint-ip-address`.

## AWS CLI

Para desasociar una dirección IP de un punto final de Resolver

El siguiente `disassociate-resolver-endpoint-ip-address` ejemplo elimina una dirección IP de un punto final entrante o saliente de Resolver especificado.

Nota: Un punto final debe tener al menos dos direcciones IP. Si un punto final tiene actualmente solo dos direcciones IP y desea reemplazar una dirección por otra, primero debe usar [associate-resolver-endpoint-ip-address](#) para asociar la nueva dirección IP. A continuación, puede desasociar una de las direcciones IP originales del punto final.

```
aws route53resolver disassociate-resolver-endpoint-ip-address \  
  --resolver-endpoint-id rslvr-in-f9ab8a03f1example \  
  --ip-address="SubnetId=subnet-12d8a459,Ip=172.31.40.121"
```

Salida:

```
{  
  "ResolverEndpoint": {  
    "Id": "rslvr-in-f9ab8a03f1example",  
    "CreatorRequestId": "2020-01-01-18:47",  
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/  
rslvr-in-f9ab8a03f1example",  
    "Name": "my-inbound-endpoint",  
    "SecurityGroupIds": [  
      "sg-f62bexam"  
    ],  
    "Direction": "INBOUND",  
    "IpAddressCount": 3,  
    "HostVPCId": "vpc-304bexam",  
    "Status": "UPDATING",  
    "StatusMessage": "Updating the Resolver Endpoint",  
    "CreationTime": "2020-01-01T23:02:29.583Z",  
    "ModificationTime": "2020-01-05T23:02:29.583Z"  
  }  
}
```

- Para API obtener más información, consulte [DisassociateResolverEndpointIpAddress](#) la Referencia de AWS CLI comandos.

## disassociate-resolver-rule

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-resolver-rule`.

### AWS CLI

Para desasociar una regla de Resolver de una de Amazon VPC

El siguiente `disassociate-resolver-rule` ejemplo elimina la asociación entre la regla de resolución especificada y la especificadaVPC. Puede desasociar una regla de a VPC en las siguientes circunstancias:

Para DNS las consultas que se originan en esta opciónVPC, desea que Resolver deje de reenviar las consultas a la red para el nombre de dominio especificado en la regla. Desea eliminar la regla de reenvío. Si una regla está actualmente asociada a una o variasVPCs, debe desasociarla de todas VPCs antes de poder eliminarla.

```
aws route53resolver disassociate-resolver-rule \
  --resolver-rule-id rslvr-rr-4955cb98ceexample \
  --vpc-id vpc-304bexam
```

Salida:

```
{
  "ResolverRuleAssociation": {
    "Id": "rslvr-rrassoc-322f4e8b9cexample",
    "ResolverRuleId": "rslvr-rr-4955cb98ceexample",
    "Name": "my-resolver-rule-association",
    "VPCId": "vpc-304bexam",
    "Status": "DELETING",
    "StatusMessage": "[Trace id: 1-5dc5ffa2-a26c38004c1f94006example] Deleting Association"
  }
}
```

- Para API obtener más información, consulte [DisassociateResolverRule](#) la Referencia de AWS CLI comandos.

## get-firewall-config

En el siguiente ejemplo de código, se muestra cómo usar `get-firewall-config`.

## AWS CLI

Para obtener una configuración de firewall para un VPC

En el siguiente `get-firewall-config` ejemplo, se recupera el comportamiento del DNS firewall del elemento especificado VPC.

```
aws route53resolver get-firewall-config \  
  --resource-id vpc-31e92222
```

Salida:

```
{  
  "FirewallConfig": {  
    "Id": "rslvr-fc-86016850cexample",  
    "ResourceId": "vpc-31e92222",  
    "OwnerId": "123456789012",  
    "FirewallFailOpen": "DISABLED"  
  }  
}
```

Para obtener más información, consulte [VPCConfiguración del DNS firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [GetFirewallConfig](#) la Referencia de AWS CLI comandos.

## `get-firewall-domain-list`

En el siguiente ejemplo de código, se muestra cómo usar `get-firewall-domain-list`.

## AWS CLI

Para obtener una lista de dominios de Route 53 Resolver DNS Firewall

En el siguiente `get-firewall-domain-list` ejemplo, se recupera la lista de dominios con el ID que especifique.

```
aws route53resolver get-firewall-domain-list \  
  --firewall-domain-list-id rslvr-fdl-42b60677cexample
```

Salida:

```
{
  "FirewallDomainList": {
    "Id": "rslvr-fdl-9e956e9ffexample",
    "Arn": "arn:aws:route53resolver:us-west-2:123457689012:firewall-domain-list/rslvr-fdl-42b60677cexample",
    "Name": "test",
    "DomainCount": 0,
    "Status": "COMPLETE",
    "StatusMessage": "Created Firewall Domain List",
    "CreatorRequestId": "my-request-id",
    "CreationTime": "2021-05-25T15:55:51.115365Z",
    "ModificationTime": "2021-05-25T15:55:51.115365Z"
  }
}
```

Para obtener más información, consulte [Administrar sus propias listas de dominios](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [GetFirewallDomainList](#) la Referencia de AWS CLI comandos.

## get-firewall-rule-group-association

En el siguiente ejemplo de código, se muestra cómo usar `get-firewall-rule-group-association`.

AWS CLI

Para obtener una asociación de grupos de reglas de firewall

En el siguiente `get-firewall-rule-group-association` ejemplo, se recupera una asociación de grupos de reglas de firewall.

```
aws route53resolver get-firewall-rule-group-association \
  --firewall-rule-group-association-id rslvr-frgassoc-57e8873d7example
```

Salida:

```
{
```

```

    "FirewallRuleGroupAssociation": {
      "Id": "rslvr-frgassoc-57e8873d7example",
      "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-association/rslvr-frgassoc-57e8873d7example",
      "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
      "VpcId": "vpc-31e92222",
      "Name": "test-association",
      "Priority": 101,
      "MutationProtection": "DISABLED",
      "Status": "COMPLETE",
      "StatusMessage": "Finished rule group association update",
      "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
      "CreationTime": "2021-05-25T21:47:48.755768Z",
      "ModificationTime": "2021-05-25T21:47:48.755768Z"
    }
  }
}

```

Para obtener más información, consulte [Administrar las asociaciones entre sus grupos de reglas de Route 53 Resolver DNS Firewall VPC](#) y los de Route 53 Resolver en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [GetFirewallRuleGroupAssociation](#) la Referencia de AWS CLI comandos.

## get-firewall-rule-group-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-firewall-rule-group-policy`.

### AWS CLI

Para obtener una AWS IAM política

En el siguiente `get-firewall-rule-group-policy` ejemplo, se obtiene la política AWS Identity and Access Management (AWS IAM) para compartir el grupo de reglas especificado.

```

aws route53resolver get-firewall-rule-group-policy \
  --arn arn:aws:route53resolver:us-west-2:AWS_ACCOUNT_ID:firewall-rule-group/
rslvr-frg-47f93271fexample

```

Salida:

```
{
```

```
"FirewallRuleGroupPolicy": "{ \"Version\": \"2012-10-17\",
  \"Statement\": [{ \"Sid\": \"test\", \"Effect\": \"Allow\", \"Principal\": { \"AWS\": \"arn:aws:iam::AWS_ACCOUNT_ID:root\" }, \"Action\": [ \"route53resolver:GetFirewallRuleGroup\", \"route53resolver:ListFirewallRuleGroups\" ], \"Resource\": \"arn:aws:route53resolver:us-east-1:AWS_ACCOUNT_ID:firewall-rule-group/rslvr-frg-47f93271fexample\" } ] }"
```

Para obtener más información, consulte [Administrar grupos de reglas y reglas en DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [GetFirewallRuleGroupPolicy](#) la Referencia de AWS CLI comandos.

## get-firewall-rule-group

En el siguiente ejemplo de código, se muestra cómo usar `get-firewall-rule-group`.

### AWS CLI

Para obtener un grupo de reglas de firewall

En el siguiente `get-firewall-rule-group` ejemplo, se recupera información sobre un grupo de reglas de DNS firewall con el identificador que usted proporciona.

```
aws route53resolver get-firewall-rule-group \
  --firewall-rule-group-id rslvr-frg-47f93271fexample
```

Salida:

```
{
  "FirewallRuleGroup": {
    "Id": "rslvr-frg-47f93271fexample",
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group/rslvr-frg-47f93271fexample",
    "Name": "test",
    "RuleCount": 0,
    "Status": "COMPLETE",
    "StatusMessage": "Created Firewall Rule Group",
    "OwnerId": "123456789012",
    "CreatorRequestId": "my-request-id",
```

```

    "ShareStatus": "NOT_SHARED",
    "CreationTime": "2021-05-25T18:59:26.490017Z",
    "ModificationTime": "2021-05-25T18:59:26.490017Z"
  }
}

```

Para obtener más información, consulte [Administrar grupos de reglas y reglas en DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [GetFirewallRuleGroup](#) la Referencia de AWS CLI comandos.

## get-resolver-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `get-resolver-endpoint`.

### AWS CLI

Para obtener información sobre un punto final de Resolver

En el siguiente `get-resolver-endpoint` ejemplo, se muestran los detalles del punto final especificado de salida. Se puede utilizar tanto `get-resolver-endpoint` para los puntos finales de entrada como de salida especificando el ID de punto final correspondiente.

```

aws route53resolver get-resolver-endpoint \
  --resolver-endpoint-id rslvr-out-d5e5920e37example

```

Salida:

```

{
  "ResolverEndpoint": {
    "Id": "rslvr-out-d5e5920e37example",
    "CreatorRequestId": "2020-01-01-18:47",
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/rslvr-out-d5e5920e37example",
    "Name": "my-outbound-endpoint",
    "SecurityGroupIds": [
      "sg-05cd7b25d6example"
    ],
    "Direction": "OUTBOUND",
    "IpAddressCount": 2,
  }
}

```



```

    "HostVPCId": "vpc-304bexam",
    "Status": "OPERATIONAL",
    "StatusMessage": "This Resolver Endpoint is operational.",
    "CreationTime": "2020-01-01T23:50:50.979Z",
    "ModificationTime": "2020-01-02T23:50:50.979Z"
  }
}

```

Para obtener más información, consulte [los valores que se especifican al crear o editar puntos de enlace entrantes](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte la Referencia [GetResolverEndpoint](#) de AWS CLI comandos.

## get-resolver-rule-association

En el siguiente ejemplo de código, se muestra cómo usar `get-resolver-rule-association`.

### AWS CLI

Para obtener información sobre la asociación entre una regla de Resolver y una VPC

En el siguiente `get-resolver-rule-association` ejemplo, se muestran detalles sobre la asociación entre una regla de resolución especificada y una VPC. Se asocia una regla de resolución y un VPC uso [associate-resolver-rule](#).

```

aws route53resolver get-resolver-rule-association \
  --resolver-rule-association-id rslvr-rrassoc-d61cbb2c8bexample

```

Salida:

```

{
  "ResolverRuleAssociation": {
    "Id": "rslvr-rrassoc-d61cbb2c8bexample",
    "ResolverRuleId": "rslvr-rr-42b60677c0example",
    "Name": "my-resolver-rule-association",
    "VPCId": "vpc-304bexam",
    "Status": "COMPLETE",
    "StatusMessage": ""
  }
}

```

- Para API obtener más información, consulte [GetResolverRuleAssociation](#) la Referencia de AWS CLI comandos.

## get-resolver-rule

En el siguiente ejemplo de código, se muestra cómo usar `get-resolver-rule`.

### AWS CLI

Para obtener información sobre una regla de resolución

En el siguiente `get-resolver-rule` ejemplo, se muestran detalles sobre la regla de resolución especificada, como el nombre de dominio al que la regla reenvía DNS las consultas y el ID del punto final de resolución saliente al que está asociada la regla.

```
aws route53resolver get-resolver-rule \  
--resolver-rule-id rslvr-rr-42b60677c0example
```

Salida:

```
{  
  "ResolverRule": {  
    "Id": "rslvr-rr-42b60677c0example",  
    "CreatorRequestId": "2020-01-01-18:47",  
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/rslvr-  
rr-42b60677c0example",  
    "DomainName": "example.com.",  
    "Status": "COMPLETE",  
    "StatusMessage": "[Trace id: 1-5dc4b177-ff1d9d001a0f80005example]  
Successfully created Resolver Rule.",  
    "RuleType": "FORWARD",  
    "Name": "my-rule",  
    "TargetIps": [  
      {  
        "Ip": "192.0.2.45",  
        "Port": 53  
      }  
    ],  
    "ResolverEndpointId": "rslvr-out-d5e5920e37example",  
    "OwnerId": "111122223333",  
    "ShareStatus": "NOT_SHARED"  
  }  
}
```

```
}
```

Para obtener más información, consulte [Valores que se especifican al crear o editar reglas](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [GetResolverRule](#) la Referencia de AWS CLI comandos.

## **import-firewall-domains**

En el siguiente ejemplo de código, se muestra cómo usar `import-firewall-domains`.

### AWS CLI

Para importar dominios a una lista de dominios

En el siguiente `import-firewall-domains` ejemplo, se importa un conjunto de dominios de un archivo a una lista de dominios de DNS Firewall que especifique.

```
aws route53resolver import-firewall-domains \  
  --firewall-domain-list-id rslvr-fdl-d61cbb2cbexample \  
  --operation REPLACE \  
  --domain-file-url s3://PATH/TO/YOUR/FILE
```

Salida:

```
{  
  "Id": "rslvr-fdl-d61cbb2cbexample",  
  "Name": "test",  
  "Status": "IMPORTING",  
  "StatusMessage": "Importing domains from provided file."  
}
```

Para obtener más información, consulte [Administrar sus propias listas de dominios](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [ImportFirewallDomains](#) la Referencia de AWS CLI comandos.

## **list-firewall-configs**

En el siguiente ejemplo de código, se muestra cómo usar `list-firewall-configs`.

## AWS CLI

Para enumerar las configuraciones de firewall

En el siguiente `list-firewall-configs` ejemplo, se enumeran las configuraciones DNS del firewall.

```
aws route53resolver list-firewall-configs
```

Salida:

```
{
  "FirewallConfigs": [
    {
      "Id": "rslvr-fc-86016850cexample",
      "ResourceId": "vpc-31e92222",
      "OwnerId": "123456789012",
      "FirewallFailOpen": "DISABLED"
    }
  ]
}
```

Para obtener más información, consulte [VPCConfiguración del DNS firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [ListFirewallConfigs](#) la Referencia de AWS CLI comandos.

## `list-firewall-domain-lists`

En el siguiente ejemplo de código, se muestra cómo usar `list-firewall-domain-lists`.

## AWS CLI

Para enumerar todas las listas de dominios de Route 53 Resolver DNS Firewall

En el siguiente `list-firewall-domain-lists` ejemplo, se enumeran todas las listas de dominios.

```
aws route53resolver list-firewall-domain-lists
```

Salida:

```
{
  "FirewallDomainLists": [
    {
      "Id": "rslvr-fdl-2c46f2ecfexample",
      "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-domain-
list/rslvr-fdl-2c46f2ecfexample",
      "Name": "AWSManagedDomainsMalwareDomainList",
      "CreatorRequestId": "AWSManagedDomainsMalwareDomainList",
      "ManagedOwnerName": "Route 53 Resolver DNS Firewall"
    },
    {
      "Id": "rslvr-fdl-aa970e9e1example",
      "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-domain-
list/rslvr-fdl-aa970e9e1example",
      "Name": "AWSManagedDomainsBotnetCommandandControl",
      "CreatorRequestId": "AWSManagedDomainsBotnetCommandandControl",
      "ManagedOwnerName": "Route 53 Resolver DNS Firewall"
    },
    {
      "Id": "rslvr-fdl-42b60677cexample",
      "Arn": "arn:aws:route53resolver:us-west-2:123456789111:firewall-domain-
list/rslvr-fdl-42b60677cexample",
      "Name": "test",
      "CreatorRequestId": "my-request-id"
    }
  ]
}
```

Para obtener más información, consulte las [listas de dominios de Route 53 Resolver DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [ListFirewallDomainLists](#) la Referencia de AWS CLI comandos.

## **list-firewall-domains**

En el siguiente ejemplo de código, se muestra cómo usar `list-firewall-domains`.

### AWS CLI

Para incluir dominios en una lista de dominios

En el siguiente `list-firewall-domains` ejemplo, se enumeran los dominios de una lista de dominios de DNS Firewall que especifique.

```
aws route53resolver list-firewall-domains \  
--firewall-domain-list-id rs1vr-fdl-d61cbb2cbexample
```

Salida:

```
{  
  "Domains": [  
    "test1.com.",  
    "test2.com.",  
    "test3.com."  
  ]  
}
```

Para obtener más información, consulte [Administrar sus propias listas de dominios](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [ListFirewallDomains](#) la Referencia de AWS CLI comandos.

## **list-firewall-rule-group-associations**

En el siguiente ejemplo de código, se muestra cómo usar `list-firewall-rule-group-associations`.

AWS CLI

Para enumerar las asociaciones de grupos de reglas de DNS Firewall

En el siguiente `list-firewall-rule-group-associations` ejemplo, se enumeran las asociaciones de grupos de reglas de DNS Firewall con AmazonVPCs.

```
aws route53resolver list-firewall-rule-group-associations
```

Salida:

```
{  
  "FirewallRuleGroupAssociations": [  

```

```
{
  "Id": "rslvr-frgassoc-57e8873d7example",
  "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-
group-association/rslvr-frgassoc-57e8873d7example",
  "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
  "VpcId": "vpc-31e92222",
  "Name": "test-association",
  "Priority": 101,
  "MutationProtection": "DISABLED",
  "Status": "UPDATING",
  "StatusMessage": "Creating Firewall Rule Group Association",
  "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
  "CreationTime": "2021-05-25T21:47:48.755768Z",
  "ModificationTime": "2021-05-25T21:47:48.755768Z"
}
]
```

Para obtener más información, consulte [Administrar las asociaciones entre su grupo de reglas VPC y el de Route 53 Resolver DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [ListFirewallRuleGroupAssociations](#) la Referencia de AWS CLI comandos.

## list-firewall-rule-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-firewall-rule-groups`.

AWS CLI

Para obtener una lista de los grupos de reglas de su firewall

En el siguiente `list-firewall-rule-groups` ejemplo, se enumeran los grupos de reglas del DNS firewall.

```
aws route53resolver list-firewall-rule-groups
```

Salida:

```
{
```

```

    "FirewallRuleGroups": [
      {
        "Id": "rslvr-frg-47f93271fexample",
        "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-
group/rslvr-frg-47f93271fexample",
        "Name": "test",
        "OwnerId": "123456789012",
        "CreatorRequestId": "my-request-id",
        "ShareStatus": "NOT_SHARED"
      }
    ]
  }
}

```

Para obtener más información, consulte [Administrar grupos de reglas y reglas en DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [ListFirewallRuleGroups](#) la Referencia de AWS CLI comandos.

## list-firewall-rules

En el siguiente ejemplo de código, se muestra cómo usar `list-firewall-rules`.

### AWS CLI

Para ver una lista de las reglas de firewall

En el siguiente `list-firewall-rules` ejemplo, se enumeran todas las reglas de DNS firewall de un grupo de reglas de firewall.

```

aws route53resolver list-firewall-rules \
  --firewall-rule-group-id rslvr-frg-47f93271fexample

```

Salida:

```

{
  "FirewallRules": [
    {
      "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
      "FirewallDomainListId": "rslvr-fdl-9e956e9ffexample",
      "Name": "allow-rule",
      "Priority": 101,

```



```

        "Action": "ALLOW",
        "CreatorRequestId": "d81e3fb7-020b-415e-939f-EXAMPLE11111",
        "CreationTime": "2021-05-25T21:44:00.346093Z",
        "ModificationTime": "2021-05-25T21:44:00.346093Z"
    }
]
}

```

Para obtener más información, consulte [Administrar grupos de reglas y reglas en DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [ListFirewallRules](#) la Referencia de AWS CLI comandos.

## list-resolver-endpoint-ip-addresses

En el siguiente ejemplo de código, se muestra cómo usar `list-resolver-endpoint-ip-addresses`.

### AWS CLI

Para enumerar las direcciones IP de un punto final entrante o saliente específico

En el siguiente `list-resolver-endpoint-ip-addresses` ejemplo, se muestra información sobre las direcciones IP asociadas al punto final de entrada. `rslvr-in-f9ab8a03f1example` También se puede utilizar `list-resolver-endpoint-ip-addresses` para los puntos de enlace de salida especificando el ID de punto final correspondiente.

```

aws route53resolver list-resolver-endpoint-ip-addresses \
  --resolver-endpoint-id rslvr-in-f9ab8a03f1example

```

Salida:

```

{
  "MaxResults": 10,
  "IpAddresses": [
    {
      "IpId": "rni-1de60cdbfeexample",
      "SubnetId": "subnet-ba47exam",
      "Ip": "192.0.2.44",
      "Status": "ATTACHED",
    }
  ]
}

```

```

        "StatusMessage": "This IP address is operational.",
        "CreationTime": "2020-01-03T23:02:29.587Z",
        "ModificationTime": "2020-01-03T23:03:05.555Z"
    },
    {
        "IpId": "rni-aac7085e38example",
        "SubnetId": "subnet-12d8exam",
        "Ip": "192.0.2.45",
        "Status": "ATTACHED",
        "StatusMessage": "This IP address is operational.",
        "CreationTime": "2020-01-03T23:02:29.593Z",
        "ModificationTime": "2020-01-03T23:02:55.060Z"
    }
]
}

```

Para obtener más información sobre los valores de la salida, consulte [Valores que se especifican al crear o editar puntos de enlace de entrada](#) y [Valores que se especifican al crear o editar puntos de enlace de salida, ambos en la Guía](#) para desarrolladores de Amazon Route 53.

- Para obtener API más información, consulte la Referencia de comandos [ListResolverEndpointIpAddresses](#).AWS CLI

## list-resolver-endpoints

En el siguiente ejemplo de código, se muestra cómo usar `list-resolver-endpoints`.

### AWS CLI

Para enumerar los puntos finales del Resolver en una región AWS

En el siguiente `list-resolver-endpoints` ejemplo, se enumeran los puntos finales de Resolver entrantes y salientes que existen en la cuenta actual.

```
aws route53resolver list-resolver-endpoints
```

Salida:

```

{
  "MaxResults": 10,
  "ResolverEndpoints": [

```

```

    {
      "Id": "rslvr-in-497098ad59example",
      "CreatorRequestId": "2020-01-01-18:47",
      "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-
endpoint/rslvr-in-497098ad59example",
      "Name": "my-inbound-endpoint",
      "SecurityGroupIds": [
        "sg-05cd7b25d6example"
      ],
      "Direction": "INBOUND",
      "IpAddressCount": 2,
      "HostVPCId": "vpc-304bexam",
      "Status": "OPERATIONAL",
      "StatusMessage": "This Resolver Endpoint is operational.",
      "CreationTime": "2020-01-01T23:25:45.538Z",
      "ModificationTime": "2020-01-01T23:25:45.538Z"
    },
    {
      "Id": "rslvr-out-d5e5920e37example",
      "CreatorRequestId": "2020-01-01-18:48",
      "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-
endpoint/rslvr-out-d5e5920e37example",
      "Name": "my-outbound-endpoint",
      "SecurityGroupIds": [
        "sg-05cd7b25d6example"
      ],
      "Direction": "OUTBOUND",
      "IpAddressCount": 2,
      "HostVPCId": "vpc-304bexam",
      "Status": "OPERATIONAL",
      "StatusMessage": "This Resolver Endpoint is operational.",
      "CreationTime": "2020-01-01T23:50:50.979Z",
      "ModificationTime": "2020-01-01T23:50:50.979Z"
    }
  ]
}

```

- Para API obtener más información, consulte la Referencia de [ListResolverEndpoints](#) comandos AWS CLI .

## list-resolver-rule-associations

En el siguiente ejemplo de código, se muestra cómo usar `list-resolver-rule-associations`.

## AWS CLI

Para enumerar las asociaciones entre las reglas de Resolver y VPCs

En el siguiente `list-resolver-rule-associations` ejemplo, se enumeran las asociaciones entre las reglas de resolución y VPCs las de la AWS cuenta corriente.

```
aws route53resolver list-resolver-rule-associations
```

Salida:

```
{
  "MaxResults": 30,
  "ResolverRuleAssociations": [
    {
      "Id": "rslvr-autodefined-assoc-vpc-304bexam-internet-resolver",
      "ResolverRuleId": "rslvr-autodefined-rr-internet-resolver",
      "Name": "System Rule Association",
      "VPCId": "vpc-304bexam",
      "Status": "COMPLETE",
      "StatusMessage": ""
    },
    {
      "Id": "rslvr-rrassoc-d61cbb2c8bexample",
      "ResolverRuleId": "rslvr-rr-42b60677c0example",
      "Name": "my-resolver-rule-association",
      "VPCId": "vpc-304bexam",
      "Status": "COMPLETE",
      "StatusMessage": ""
    }
  ]
}
```

Para obtener más información, consulte [Cómo Route 53 Resolver reenvía DNS las consultas de su red VPCs a su red](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [ListResolverRuleAssociations](#) la Referencia de AWS CLI comandos.

## **list-resolver-rules**

En el siguiente ejemplo de código, se muestra cómo usar `list-resolver-rules`.

## AWS CLI

Para enumerar las reglas de Resolver

En el siguiente `list-resolver-rules` ejemplo, se enumeran todas las reglas de Resolver de la AWS cuenta corriente.

```
aws route53resolver list-resolver-rules
```

Salida:

```
{
  "MaxResults": 30,
  "ResolverRules": [
    {
      "Id": "rslvr-autodefined-rr-internet-resolver",
      "CreatorRequestId": "",
      "Arn": "arn:aws:route53resolver:us-west-2::autodefined-rule/rslvr-
autodefined-rr-internet-resolver",
      "DomainName": ".",
      "Status": "COMPLETE",
      "RuleType": "RECURSIVE",
      "Name": "Internet Resolver",
      "OwnerId": "Route 53 Resolver",
      "ShareStatus": "NOT_SHARED"
    },
    {
      "Id": "rslvr-rr-42b60677c0example",
      "CreatorRequestId": "2020-01-01-18:47",
      "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/
rslvr-rr-42b60677c0bc4e299",
      "DomainName": "example.com.",
      "Status": "COMPLETE",
      "StatusMessage": "[Trace id: 1-5dc4b177-ff1d9d001a0f80005example]
Successfully created Resolver Rule.",
      "RuleType": "FORWARD",
      "Name": "my-rule",
      "TargetIps": [
        {
          "Ip": "192.0.2.45",
          "Port": 53
        }
      ]
    }
  ],
}
```

```

        "ResolverEndpointId": "rslvr-out-d5e5920e37example",
        "OwnerId": "111122223333",
        "ShareStatus": "NOT_SHARED"
    }
]
}

```

Para obtener más información, consulte [Cómo Route 53 Resolver reenvía DNS las consultas de su red VPCs a su red](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [ListResolverRules](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para enumerar las etiquetas de un recurso de Resolver

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas que se asignan a la regla de resolución especificada.

```

aws route53resolver list-tags-for-resource \
  --resource-arn "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/
  rslvr-rr-42b60677c0example"

```

Salida:

```

{
  "Tags": [
    {
      "Key": "my-key-1",
      "Value": "my-value-1"
    },
    {
      "Key": "my-key-2",
      "Value": "my-value-2"
    }
  ]
}

```

Para obtener información sobre el uso de etiquetas para la asignación de costes, consulte [Uso de etiquetas de asignación de costes](#) en la Guía del usuario de AWS Billing and Cost Management.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## **put-firewall-rule-group-policy**

En el siguiente ejemplo de código, se muestra cómo usar `put-firewall-rule-group-policy`.

### AWS CLI

Para adjuntar una AWS IAM política para compartir una política de grupo de reglas de firewall

En el siguiente `put-firewall-rule-group-policy` ejemplo, se adjunta una política de AWS Identity and Access Management (AWS IAM) para compartir el grupo de reglas.

```
aws route53resolver put-firewall-rule-group-policy \
  --firewall-rule-group-policy "{\"Version\":\"2012-10-17\",
  \"Statement\": [{\"Sid\":\"test\",\"Effect\":\"Allow\",\"Principal
  \": {\"AWS\":\"arn:aws:iam::AWS_ACCOUNT_ID:root\"}, \"Action\":
  [\"route53resolver:GetFirewallRuleGroup\",\"route53resolver:ListFirewallRuleGroups
  \"], \"Resource\":\"arn:aws:route53resolver:us-east-1:AWS_ACCOUNT_ID:firewall-rule-
  group/rs1vr-frg-47f93271fexample\"}]}"
```

Salida:

```
{
  "ReturnValue": true
}
```

Para obtener más información, consulte [Administrar grupos de reglas y reglas en DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [PutFirewallRuleGroupPolicy](#) la Referencia de AWS CLI comandos.

## **put-resolver-rule-policy**

En el siguiente ejemplo de código, se muestra cómo usar `put-resolver-rule-policy`.

## AWS CLI

Para compartir una regla de Resolver con otra AWS cuenta

El siguiente `put-resolver-rule-policy` ejemplo especifica una regla de Resolver que desea compartir con otra AWS cuenta, la cuenta con la que desea compartir la regla y las operaciones relacionadas con la regla que desea que la cuenta pueda realizar en las reglas.

Nota: Debe ejecutar este comando con las credenciales de la misma cuenta que creó la regla.

```
aws route53resolver put-resolver-rule-policy \
  --region us-east-1 \
  --arn "arn:aws:route53resolver:us-east-1:111122223333:resolver-rule/rslvr-rr-42b60677c0example" \
  --resolver-rule-policy "{\"Version\": \"2012-10-17\", \
    \"Statement\": [ { \
      \"Effect\" : \"Allow\", \
      \"Principal\" : {\"AWS\" : \"444455556666\" }, \
      \"Action\" : [ \
        \"route53resolver:GetResolverRule\", \
        \"route53resolver:AssociateResolverRule\", \
        \"route53resolver:DisassociateResolverRule\", \
        \"route53resolver:ListResolverRules\", \
        \"route53resolver:ListResolverRuleAssociations\" ], \
      \"Resource\" : [ \"arn:aws:route53resolver:us-east-1:111122223333:resolver-rule/rslvr-rr-42b60677c0example\" ] } ] }"
```

Salida:

```
{
  "ReturnValue": true
}
```

Después de ejecutar `put-resolver-rule-policy`, puede ejecutar los dos comandos siguientes de Resource Access Manager (RAM). Debe usar la cuenta con la que quiere compartir la regla:

`get-resource-share-invitations` devuelve el valor `resourceShareInvitationArn`. Necesita este valor para aceptar la invitación a usar la regla compartida. `accept-resource-share-invitation` acepta la invitación a usar la regla compartida.

Para obtener más información, consulte la siguiente documentación sobre :



[get-resource-share-invitationsaccept-resource-share-invitationsCompartir reglas de reenvío con otras AWS cuentas y usar reglas compartidas](#) en la guía para desarrolladores de Amazon Route 53

- Para API obtener más información, consulte [PutResolverRulePolicy](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para asociar etiquetas a un recurso de Resolver

El siguiente `tag-resource` ejemplo asocia dos pares clave/valor de etiqueta a la regla Resolver especificada.

```
aws route53resolver tag-resource \  
  --resource-arn "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/  
  rslvr-rr-42b60677c0example" \  
  --tags "Key=my-key-1,Value=my-value-1" "Key=my-key-2,Value=my-value-2"
```

Este comando no genera ninguna salida.

Para obtener información sobre el uso de etiquetas para la asignación de costes, consulte [Uso de etiquetas de asignación de costes](#) en la Guía del usuario de AWS Billing and Cost Management.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar etiquetas de un recurso de Resolver

En el siguiente `untag-resource` ejemplo, se quitan dos etiquetas de la regla de resolución especificada.

```
aws route53resolver untag-resource \  
  --resource-arn "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/  
rslvr-rr-42b60677c0example" \  
  --tag-keys my-key-1 my-key-2
```

Este comando no genera ninguna salida. Para confirmar que se han eliminado las etiquetas, puede utilizar [list-tags-for-resource](#).

Para obtener información sobre el uso de etiquetas para la asignación de costes, consulte [Uso de etiquetas de asignación de costes](#) en la Guía del usuario de AWS Billing and Cost Management.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-firewall-config

En el siguiente ejemplo de código, se muestra cómo usar `update-firewall-config`.

### AWS CLI

Para actualizar la configuración de un firewall

El siguiente `update-firewall-config` ejemplo actualiza la configuración del DNS firewall.

```
aws route53resolver update-firewall-config \  
  --resource-id vpc-31e92222 \  
  --firewall-fail-open DISABLED
```

Salida:

```
{  
  "FirewallConfig": {  
    "Id": "rslvr-fc-86016850cexample",  
    "ResourceId": "vpc-31e92222",  
    "OwnerId": "123456789012",  
    "FirewallFailOpen": "DISABLED"  
  }  
}
```

Para obtener más información, consulte [VPCConfiguración del DNS firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [UpdateFirewallConfig](#) la Referencia de AWS CLI comandos.

## update-firewall-domains

En el siguiente ejemplo de código, se muestra cómo usar `update-firewall-domains`.

### AWS CLI

Para actualizar una lista de dominios

En el siguiente `update-firewall-domains` ejemplo, se agregan los dominios a una lista de dominios con el identificador que usted proporcione.

```
aws route53resolver update-firewall-domains \  
  --firewall-domain-list-id rslvr-fdl-42b60677cexampleb \  
  --operation ADD \  
  --domains test1.com test2.com test3.com
```

Salida:

```
{  
  "Id": "rslvr-fdl-42b60677cexample",  
  "Name": "test",  
  "Status": "UPDATING",  
  "StatusMessage": "Updating the Firewall Domain List"  
}
```

Para obtener más información, consulte [Administrar sus propias listas de dominios](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [UpdateFirewallDomains](#) la Referencia de AWS CLI comandos.

## update-firewall-rule-group-association

En el siguiente ejemplo de código, se muestra cómo usar `update-firewall-rule-group-association`.

## AWS CLI

Para actualizar la asociación de un grupo de reglas de firewall

En el siguiente `update-firewall-rule-group-association` ejemplo, se actualiza una asociación de grupos de reglas de firewall.

```
aws route53resolver update-firewall-rule-group-association \  
  --firewall-rule-group-association-id rslvr-frgassoc-57e8873d7example \  
  --priority 103
```

Salida:

```
{  
  "FirewallRuleGroupAssociation": {  
    "Id": "rslvr-frgassoc-57e8873d7example",  
    "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-  
association/rslvr-frgassoc-57e8873d7example",  
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",  
    "VpcId": "vpc-31e92222",  
    "Name": "test-association",  
    "Priority": 103,  
    "MutationProtection": "DISABLED",  
    "Status": "UPDATING",  
    "StatusMessage": "Updating the Firewall Rule Group Association Attributes",  
    "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",  
    "CreationTime": "2021-05-25T21:47:48.755768Z",  
    "ModificationTime": "2021-05-25T21:50:09.272569Z"  
  }  
}
```

Para obtener más información, consulte [Administrar las asociaciones entre su grupo de reglas VPC y el de Route 53 Resolver DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [UpdateFirewallRuleGroupAssociation](#) la Referencia de AWS CLI comandos.

## update-firewall-rule

En el siguiente ejemplo de código, se muestra cómo usar `update-firewall-rule`.

## AWS CLI

Para actualizar una regla de firewall

En el siguiente `update-firewall-rule` ejemplo, se actualiza una regla de firewall con los parámetros que especifique.

```
aws route53resolver update-firewall-rule \  
  --firewall-rule-group-id rslvr-frg-47f93271fexample \  
  --firewall-domain-list-id rslvr-fdl-9e956e9ffexample \  
  --priority 102
```

Salida:

```
{  
  "FirewallRule": {  
    "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",  
    "FirewallDomainListId": "rslvr-fdl-9e956e9ffexample",  
    "Name": "allow-rule",  
    "Priority": 102,  
    "Action": "ALLOW",  
    "CreatorRequestId": "d81e3fb7-020b-415e-939f-EXAMPLE11111",  
    "CreationTime": "2021-05-25T21:44:00.346093Z",  
    "ModificationTime": "2021-05-25T21:45:59.611600Z"  
  }  
}
```

Para obtener más información, consulte [Administrar grupos de reglas y reglas en DNS Firewall](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [UpdateFirewallRule](#) la Referencia de AWS CLI comandos.

## update-resolver-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `update-resolver-endpoint`.

## AWS CLI

Para actualizar el nombre de un punto final de Resolver

En el siguiente `update-resolver-endpoint` ejemplo, se actualiza el nombre de un punto final del Resolver. No se admite la actualización de otros valores.

```
aws route53resolver update-resolver-endpoint \  
  --resolver-endpoint-id rslvr-in-b5d45e32bdc445f09 \  
  --name my-renamed-inbound-endpoint
```

Salida:

```
{  
  "ResolverEndpoint": {  
    "Id": "rslvr-in-b5d45e32bdexample",  
    "CreatorRequestId": "2020-01-02-18:48",  
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/  
rslvr-in-b5d45e32bdexample",  
    "Name": "my-renamed-inbound-endpoint",  
    "SecurityGroupIds": [  
      "sg-f62bexam"  
    ],  
    "Direction": "INBOUND",  
    "IpAddressCount": 2,  
    "HostVPCId": "vpc-304bexam",  
    "Status": "OPERATIONAL",  
    "StatusMessage": "This Resolver Endpoint is operational.",  
    "CreationTime": "2020-01-01T18:33:59.265Z",  
    "ModificationTime": "2020-01-08T18:33:59.265Z"  
  }  
}
```

- Para API obtener más información, consulte [UpdateResolverEndpoint](#) la Referencia de AWS CLI comandos.

## update-resolver-rule

En el siguiente ejemplo de código, se muestra cómo usar `update-resolver-rule`.

AWS CLI

Ejemplo 1: Para actualizar la configuración del punto final de Resolver

En el siguiente `update-resolver-rule` ejemplo, se actualiza el nombre de la regla, las direcciones IP de la red local a las que se reenvían DNS las consultas y el ID del punto final de Resolver saliente que se utiliza para reenviar las consultas a la red.

Nota Los valores existentes `TargetIps` se sobrescriben, por lo que debe especificar todas las direcciones IP que desee que tenga la regla después de la actualización.

```
aws route53resolver update-resolver-rule \  
  --resolver-rule-id rslvr-rr-1247fa64f3example \  
  --config Name="my-2nd-rule",TargetIps=[{Ip=192.0.2.45,Port=53},  
{Ip=192.0.2.46,Port=53}],ResolverEndpointId=rslvr-out-7b89ed0d25example
```

Salida:

```
{  
  "ResolverRule": {  
    "Id": "rslvr-rr-1247fa64f3example",  
    "CreatorRequestId": "2020-01-02-18:47",  
    "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/rslvr-  
rr-1247fa64f3example",  
    "DomainName": "www.example.com.",  
    "Status": "COMPLETE",  
    "StatusMessage": "[Trace id: 1-5dcc90b9-8a8ee860aba1ebd89example]  
Successfully updated Resolver Rule.",  
    "RuleType": "FORWARD",  
    "Name": "my-2nd-rule",  
    "TargetIps": [  
      {  
        "Ip": "192.0.2.45",  
        "Port": 53  
      },  
      {  
        "Ip": "192.0.2.46",  
        "Port": 53  
      }  
    ],  
    "ResolverEndpointId": "rslvr-out-7b89ed0d25example",  
    "OwnerId": "111122223333",  
    "ShareStatus": "NOT_SHARED"  
  }  
}
```

Ejemplo 2: Para actualizar la configuración, Resolver Endpoint utiliza un archivo para la configuración de ``config``

También puede incluir la config configuración en un JSON archivo y, a continuación, especificar ese archivo cuando lo llame. `update-resolver-rule`

```
aws route53resolver update-resolver-rule \  
  --resolver-rule-id rslvr-rr-1247fa64f3example \  
  --config file://c:\temp\update-resolver-rule.json
```

Contenido de `update-resolver-rule.json`.

```
{  
  "Name": "my-2nd-rule",  
  "TargetIps": [  
    {  
      "Ip": "192.0.2.45",  
      "Port": 53  
    },  
    {  
      "Ip": "192.0.2.46",  
      "Port": 53  
    }  
  ],  
  "ResolverEndpointId": "rslvr-out-7b89ed0d25example"  
}
```

Para obtener más información, consulte [Valores que se especifican al crear o editar reglas](#) en la Guía para desarrolladores de Amazon Route 53.

- Para API obtener más información, consulte [UpdateResolverRule](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon S3 que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface mediante Amazon S3.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.



Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **abort-multipart-upload**

En el siguiente ejemplo de código, se muestra cómo usar `abort-multipart-upload`.

#### AWS CLI

Anular la carga multiparte especificada

El siguiente comando `abort-multipart-upload` anula una carga multiparte de la clave `multipart/01` en el bucket `my-bucket`.

```
aws s3api abort-multipart-upload \  
  --bucket my-bucket \  
  --key multipart/01 \  
  --upload-  
id dfRtDYU0WWCCcH43C3WFbkR0NycyCpTJJvxu2i5GYkZLjF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3UR
```

El ID de carga requerido por este comando se genera mediante `create-multipart-upload` y también se puede recuperar con `list-multipart-uploads`.

- Para API obtener más información, consulte [AbortMultipartUpload](#) la Referencia de AWS CLI comandos.

### **complete-multipart-upload**

En el siguiente ejemplo de código, se muestra cómo usar `complete-multipart-upload`.

#### AWS CLI

El siguiente comando completa una carga multiparte de la clave `multipart/01` en el bucket `my-bucket`:

```
aws s3api complete-multipart-upload --multipart-upload file://
mpustruct --bucket my-bucket --key 'multipart/01' --upload-
id dfRtDYU0WMCcH43C3WfbkR0NycyCpTJJvxu2i5GYkZljF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3UR
```

El ID de carga requerido por este comando se genera mediante `create-multipart-upload` y también se puede recuperar con `list-multipart-uploads`.

La opción de carga multiparte del comando anterior adopta una JSON estructura que describe las partes de la carga multiparte que se deben volver a ensamblar en el archivo completo. En este ejemplo, el `file://` prefijo se usa para cargar la JSON estructura desde un archivo de la carpeta local denominada `mpustruct`

`mpustruct`:

```
{
  "Parts": [
    {
      "ETag": "e868e0f4719e394144ef36531ee6824c",
      "PartNumber": 1
    },
    {
      "ETag": "6bb2b12753d66fe86da4998aa33fffb0",
      "PartNumber": 2
    },
    {
      "ETag": "d0a0112e841abec9c9ec83406f0159c8",
      "PartNumber": 3
    }
  ]
}
```

El ETag valor de cada parte que se carga se muestra cada vez que se carga una parte mediante el `upload-part` comando y también se puede recuperar mediante una llamada `list-parts` o calcularse tomando la MD5 suma de comprobación de cada parte.

Salida:

```
{
  "ETag": "\"3944a9f7a4faab7f78788ff6210f63f0-3\"",
  "Bucket": "my-bucket",
  "Location": "https://my-bucket.s3.amazonaws.com/multipart%2F01",
```

```
"Key": "multipart/01"
}
```

- Para API obtener más información, consulte [CompleteMultipartUpload](#) la Referencia de AWS CLI comandos.

## copy-object

En el siguiente ejemplo de código, se muestra cómo usar `copy-object`.

### AWS CLI

El siguiente comando copia un objeto de bucket -1 a bucket -2:

```
aws s3api copy-object --copy-source bucket-1/test.txt --key test.txt --
bucket bucket-2
```

Salida:

```
{
  "CopyObjectResult": {
    "LastModified": "2015-11-10T01:07:25.000Z",
    "ETag": "\"589c8b79c230a6ecd5a7e1d040a9a030\""
  },
  "VersionId": "YdnYvTCVDqRRFA.NFJjy36p0hxifM1kA"
}
```

- Para API obtener más información, consulte [CopyObject](#) la Referencia de AWS CLI comandos.

## cp

En el siguiente ejemplo de código, se muestra cómo usar `cp`.

### AWS CLI

Ejemplo 1: copiar un archivo local a S3

El siguiente `cp` comando copia un único archivo en un depósito y una clave específicos:

```
aws s3 cp test.txt s3://mybucket/test2.txt
```

Salida:

```
upload: test.txt to s3://mybucket/test2.txt
```

Ejemplo 2: copiar un archivo local a S3 con una fecha de caducidad

El siguiente cp comando copia un único archivo en un depósito y una clave específicos que caducan en la fecha ISO 8601 especificada:

```
aws s3 cp test.txt s3://mybucket/test2.txt \  
--expires 2014-10-01T20:30:00Z
```

Salida:

```
upload: test.txt to s3://mybucket/test2.txt
```

Ejemplo 3: copiar un archivo de S3 a S3

El siguiente cp comando copia un único objeto s3 en un bucket y una clave específicos:

```
aws s3 cp s3://mybucket/test.txt s3://mybucket/test2.txt
```

Salida:

```
copy: s3://mybucket/test.txt to s3://mybucket/test2.txt
```

Ejemplo 4: copiar un objeto S3 a un archivo local

El siguiente cp comando copia un único objeto en un archivo especificado de forma local:

```
aws s3 cp s3://mybucket/test.txt test2.txt
```

Salida:

```
download: s3://mybucket/test.txt to test2.txt
```

Ejemplo 5: copiar un objeto S3 de un bucket a otro

El siguiente cp comando copia un único objeto en un depósito específico sin perder su nombre original:

```
aws s3 cp s3://mybucket/test.txt s3://mybucket2/
```

Salida:

```
copy: s3://mybucket/test.txt to s3://mybucket2/test.txt
```

Ejemplo 6: copiar objetos de S3 de forma recursiva a un directorio local

Cuando se pasa el parámetro `--recursive`, el siguiente `cp` comando copia de forma recursiva todos los objetos con un prefijo y un bucket especificados en un directorio específico. En este ejemplo, el depósito `mybucket` contiene los objetos `test1.txt` y `test2.txt`

```
aws s3 cp s3://mybucket . \  
--recursive
```

Salida:

```
download: s3://mybucket/test1.txt to test1.txt  
download: s3://mybucket/test2.txt to test2.txt
```

Ejemplo 7: copiar archivos locales a S3 de forma recursiva

Cuando se le pasa el parámetro `--recursive`, el siguiente `cp` comando copia de forma recursiva todos los archivos de un directorio específico en un depósito y un prefijo específicos y, al mismo tiempo, excluye algunos archivos mediante un parámetro. `--exclude` En este ejemplo, el directorio `myDir` contiene los archivos `test1.txt` y `test2.jpg`

```
aws s3 cp myDir s3://mybucket/ \  
--recursive \  
--exclude "*.jpg"
```

Salida:

```
upload: myDir/test1.txt to s3://mybucket/test1.txt
```

Ejemplo 8: Copiar objetos S3 de forma recursiva a otro depósito

Cuando se pasa el parámetro `--recursive`, el siguiente `cp` comando copia de forma recursiva todos los objetos de un depósito específico en otro depósito y, al mismo tiempo, excluye algunos

objetos mediante un `--exclude` parámetro. En este ejemplo, el depósito `mybucket` contiene los objetos `test1.txt` y `another/test1.txt`:

```
aws s3 cp s3://mybucket/ s3://mybucket2/ \  
  --recursive \  
  --exclude "another/*"
```

Salida:

```
copy: s3://mybucket/test1.txt to s3://mybucket2/test1.txt
```

Puede combinar `--exclude` `--include` opciones para copiar solo los objetos que coincidan con un patrón, excluyendo todos los demás:

```
aws s3 cp s3://mybucket/logs/ s3://mybucket2/logs/ \  
  --recursive \  
  --exclude "*" \  
  --include "*.log"
```

Salida:

```
copy: s3://mybucket/logs/test/test.log to s3://mybucket2/logs/test/test.log  
copy: s3://mybucket/logs/test3.log to s3://mybucket2/logs/test3.log
```

Ejemplo 9: Configurar la lista de control de acceso (ACL) al copiar un objeto S3

El siguiente `cp` comando copia un único objeto en un depósito y una clave específicos y, al mismo tiempo, establece el ACL `public-read-write`:

```
aws s3 cp s3://mybucket/test.txt s3://mybucket/test2.txt \  
  --acl public-read-write
```

Salida:

```
copy: s3://mybucket/test.txt to s3://mybucket/test2.txt
```

Tenga en cuenta que si utiliza la `--acl` opción, asegúrese de que todas IAM las políticas asociadas incluyan la `s3:PutObjectAcl` acción:

```
aws iam get-user-policy \  
  --user-name my-user-name \  
  --policy-name my-policy-name
```

```
--user-name myuser \  
--policy-name mypolicy
```

Salida:

```
{  
  "UserName": "myuser",  
  "PolicyName": "mypolicy",  
  "PolicyDocument": {  
    "Version": "2012-10-17",  
    "Statement": [  
      {  
        "Action": [  
          "s3:PutObject",  
          "s3:PutObjectAcl"  
        ],  
        "Resource": [  
          "arn:aws:s3:::mybucket/*"  
        ],  
        "Effect": "Allow",  
        "Sid": "Stmt1234567891234"  
      }  
    ]  
  }  
}
```

### Ejemplo 10: Concesión de permisos para un objeto de S3

El siguiente `cp` comando ilustra el uso de la `--grants` opción para conceder acceso de lectura a todos los usuarios identificados por un usuario específico identificado por su ID canónico URI y el control total a un usuario específico identificado por su ID canónico:

```
aws s3 cp file.txt s3://mybucket/ --grants read=uri=http://acs.amazonaws.com/groups/global/AllUsers full=id=79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
```

Salida:

```
upload: file.txt to s3://mybucket/file.txt
```

### Ejemplo 11: Carga de un flujo de archivos local a S3

PowerShell puede alterar la codificación o añadir una entrada CRLF canalizada.

El siguiente cp comando carga un flujo de archivos local desde la entrada estándar a un depósito y una clave específicos:

```
aws s3 cp - s3://mybucket/stream.txt
```

Ejemplo 12: Subir un flujo de archivos local de más de 50 GB a S3

El siguiente cp comando carga un flujo de archivos local de 51 GB desde la entrada estándar a un depósito y una clave específicos. Se debe proporcionar la `--expected-size` opción o la carga podría fallar si se alcanza el límite predeterminado de 10 000 piezas:

```
aws s3 cp - s3://mybucket/stream.txt --expected-size 54760833024
```

Ejemplo 13: Descargar un objeto S3 como flujo de archivos local

PowerShell puede alterar la codificación o añadir una CRLF salida canalizada o redirigida.

El siguiente cp comando descarga un objeto S3 localmente como un flujo a la salida estándar. La descarga como transmisión no es compatible actualmente con el `--recursive` parámetro:

```
aws s3 cp s3://mybucket/stream.txt -
```

Ejemplo 14: Carga a un punto de acceso S3

El siguiente cp comando carga un único archivo (`mydoc.txt`) en el punto de acceso (`myaccesspoint`) situado en la tecla (`mykey`):

```
aws s3 cp mydoc.txt s3://arn:aws:s3:us-west-2:123456789012:accesspoint/  
myaccesspoint/mykey
```

Salida:

```
upload: mydoc.txt to s3://arn:aws:s3:us-west-2:123456789012:accesspoint/  
myaccesspoint/mykey
```

Ejemplo 15: Descarga desde un punto de acceso S3

El siguiente cp comando descarga un único objeto (`mykey`) del punto de acceso (`myaccesspoint`) al archivo local (`mydoc.txt`):



```
aws s3 cp s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/mykey mydoc.txt
```

Salida:

```
download: s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/mykey to mydoc.txt
```

- Para API obtener más información, consulte [Cp](#) en la referencia de AWS CLI comandos.

## create-bucket

En el siguiente ejemplo de código, se muestra cómo usar create-bucket.

### AWS CLI

Ejemplo 1: Creación de un bucket

En los siguientes ejemplos de create-bucket se crea un bucket denominado my-bucket:

```
aws s3api create-bucket \  
  --bucket my-bucket \  
  --region us-east-1
```

Salida:

```
{  
  "Location": "/my-bucket"  
}
```

Para obtener más información, consulte [Crear un bucket](#) en la Guía del usuario de Amazon S3.

Ejemplo 2: Creación de un bucket con propietario obligatorio

En el siguiente ejemplo de create-bucket se crea un bucket denominado my-bucket que utiliza la configuración Aplicada al propietario del bucket de S3 Object Ownership.

```
aws s3api create-bucket \  
  --bucket my-bucket \  
  --region us-east-1 \  
  --object-ownership BucketOwnerEnforced
```

Salida:

```
{
  "Location": "/my-bucket"
}
```

Para obtener más información, consulte [Control de la propiedad de objetos y desactivación ACLs](#) en la Guía del usuario de Amazon S3.

Ejemplo 3: Creación de un bucket fuera de la región ``us-east-1``

En el siguiente ejemplo `create-bucket`, se crea un bucket denominado `my-bucket` en la región `eu-west-1`. Las regiones situadas fuera de `us-east-1` requieren que se especifique el `LocationConstraint` correspondiente para poder crear el bucket en la región deseada.

```
aws s3api create-bucket \
  --bucket my-bucket \
  --region eu-west-1 \
  --create-bucket-configuration LocationConstraint=eu-west-1
```

Salida:

```
{
  "Location": "http://my-bucket.s3.amazonaws.com/"
}
```

Para obtener más información, consulte [Crear un bucket](#) en la Guía del usuario de Amazon S3.

- Para API obtener más información, consulte [CreateBucket](#) la Referencia de AWS CLI comandos.

## **create-multipart-upload**

En el siguiente ejemplo de código, se muestra cómo usar `create-multipart-upload`.

AWS CLI

El siguiente comando crea una carga multiparte en el bucket `my-bucket` con la clave `multipart/01`:

```
aws s3api create-multipart-upload --bucket my-bucket --key 'multipart/01'
```

**Salida:**

```
{
  "Bucket": "my-bucket",
  "UploadId":
  "dfRtDYU0WWCCcH43C3WFbkR0NycyCpTJJvxu2i5GYkZljF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3URC
  "Key": "multipart/01"
}
```

El archivo completo se denominará 01 en una carpeta llamada `multipart` en el bucket `my-bucket`. Guarde el ID de carga, la clave y el nombre del bucket para usarlos con el comando `upload-part`.

- Para API obtener más información, consulte [CreateMultipartUpload](#) la Referencia de AWS CLI comandos.

**delete-bucket-analytics-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-analytics-configuration`.

## AWS CLI

Eliminar una configuración de análisis de un bucket

En el siguiente ejemplo de `delete-bucket-analytics-configuration`, se elimina la configuración de análisis para el bucket e ID especificados.

```
aws s3api delete-bucket-analytics-configuration \
  --bucket my-bucket \
  --id 1
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteBucketAnalyticsConfiguration](#) la Referencia de AWS CLI comandos.

**delete-bucket-cors**

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-cors`.

## AWS CLI

El siguiente comando elimina la configuración de uso compartido de recursos entre orígenes desde un bucket denominado `my-bucket`:

```
aws s3api delete-bucket-cors --bucket my-bucket
```

- Para API obtener más información, consulte [DeleteBucketCors](#) la Referencia de AWS CLI comandos.

## **delete-bucket-encryption**

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-encryption`.

### AWS CLI

Eliminar la configuración de cifrado del servidor de un bucket

En el siguiente ejemplo de `delete-bucket-encryption`, se elimina la configuración de cifrado del servidor del bucket especificado.

```
aws s3api delete-bucket-encryption \  
  --bucket my-bucket
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteBucketEncryption](#) la Referencia de AWS CLI comandos.

## **delete-bucket-intelligent-tiering-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-intelligent-tiering-configuration`.

### AWS CLI

Para eliminar una configuración de S3 Intelligent-Tiering de un bucket

En el siguiente `delete-bucket-intelligent-tiering-configuration` ejemplo, se elimina una configuración de S3 Intelligent-Tiering, denominada, de un bucket. ExampleConfig

```
aws s3api delete-bucket-intelligent-tiering-configuration \  
  --bucket my-bucket --configuration ExampleConfig
```

```
--bucket DOC-EXAMPLE-BUCKET \  
--id ExampleConfig
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de S3 Intelligent-Tiering](#) en la Guía del usuario de Amazon S3.

- Para obtener API más información, consulte la Referencia de comandos [DeleteBucketIntelligentTieringConfiguration](#).AWS CLI

## delete-bucket-inventory-configuration

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-inventory-configuration`.

### AWS CLI

Eliminar la configuración de inventario de un bucket

En el siguiente ejemplo de `delete-bucket-inventory-configuration`, se elimina la configuración de inventario con el ID 1 del bucket especificado.

```
aws s3api delete-bucket-inventory-configuration \  
--bucket my-bucket \  
--id 1
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteBucketInventoryConfiguration](#) la Referencia de AWS CLI comandos.

## delete-bucket-lifecycle

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-lifecycle`.

### AWS CLI

El comando siguiente elimina una configuración del ciclo de vida de un bucket denominado `my-bucket`:

```
aws s3api delete-bucket-lifecycle --bucket my-bucket
```

- Para API obtener más información, consulte [DeleteBucketLifecycle](#) la Referencia de AWS CLI comandos.

## **delete-bucket-metrics-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-metrics-configuration`.

### AWS CLI

Eliminar una configuración de métricas de un bucket

En el siguiente ejemplo de `delete-bucket-metrics-configuration`, se elimina la configuración de métricas para el bucket e ID especificados.

```
aws s3api delete-bucket-metrics-configuration \  
  --bucket my-bucket \  
  --id 123
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteBucketMetricsConfiguration](#) la Referencia de AWS CLI comandos.

## **delete-bucket-ownership-controls**

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-ownership-controls`.

### AWS CLI

Para eliminar la configuración de propiedad de un bucket

El siguiente `delete-bucket-ownership-controls` ejemplo elimina la configuración de propiedad de un bucket.

```
aws s3api delete-bucket-ownership-controls \  
  --bucket DOC-EXAMPLE-BUCKET
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Establecer la propiedad de un objeto en un bucket existente](#) en la Guía del usuario de Amazon S3.

- Para API obtener más información, consulte [DeleteBucketOwnershipControls](#) la Referencia de AWS CLI comandos.

## **delete-bucket-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-policy`.

### AWS CLI

El comando siguiente elimina una política de bucket de un bucket denominado `my-bucket`:

```
aws s3api delete-bucket-policy --bucket my-bucket
```

- Para API obtener más información, consulte [DeleteBucketPolicy](#) la Referencia de AWS CLI comandos.

## **delete-bucket-replication**

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-replication`.

### AWS CLI

El siguiente comando elimina la configuración de replicación de un bucket denominado `my-bucket`:

```
aws s3api delete-bucket-replication --bucket my-bucket
```

- Para API obtener más información, consulte [DeleteBucketReplication](#) la Referencia de AWS CLI comandos.

## **delete-bucket-tagging**

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-tagging`.

### AWS CLI

El siguiente comando elimina la configuración de etiquetado de un bucket denominado `my-bucket`:

```
aws s3api delete-bucket-tagging --bucket my-bucket
```

- Para API obtener más información, consulte [DeleteBucketTagging](#) la Referencia de AWS CLI comandos.

## **delete-bucket-website**

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket-website`.

### AWS CLI

El siguiente comando elimina la configuración de un sitio web de un bucket denominado `my-bucket`:

```
aws s3api delete-bucket-website --bucket my-bucket
```

- Para API obtener más información, consulte [DeleteBucketWebsite](#) la Referencia de AWS CLI comandos.

## **delete-bucket**

En el siguiente ejemplo de código, se muestra cómo usar `delete-bucket`.

### AWS CLI

El comando siguiente elimina un bucket denominado `my-bucket`:

```
aws s3api delete-bucket --bucket my-bucket --region us-east-1
```

- Para API obtener más información, consulte [DeleteBucket](#) la Referencia de AWS CLI comandos.

## **delete-object-tagging**

En el siguiente ejemplo de código, se muestra cómo usar `delete-object-tagging`.

### AWS CLI

Eliminar los conjuntos de etiquetas de un objeto



En el siguiente ejemplo de `delete-object-tagging`, se elimina del objeto `doc1.rtf` la etiqueta con la clave especificada.

```
aws s3api delete-object-tagging \  
  --bucket my-bucket \  
  --key doc1.rtf
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteObjectTagging](#) la Referencia de AWS CLI comandos.

## delete-object

En el siguiente ejemplo de código, se muestra cómo usar `delete-object`.

### AWS CLI

El comando siguiente elimina un objeto denominado `test.txt` de un bucket denominado `my-bucket`:

```
aws s3api delete-object --bucket my-bucket --key test.txt
```

Si el control de versiones del bucket está activado, el resultado contendrá el ID de versión del marcador de eliminación:

```
{  
  "VersionId": "9_gKg5vG56F.TTEUdwkxGpJ3tND1W1Gq",  
  "DeleteMarker": true  
}
```

Para obtener más información acerca de la eliminación de objetos, consulte [Eliminación de objetos](#) en la Guía para desarrolladores de Amazon S3.

- Para API obtener más información, consulte [DeleteObject](#) la Referencia de AWS CLI comandos.

## delete-objects

En el siguiente ejemplo de código, se muestra cómo usar `delete-objects`.

## AWS CLI

El comando siguiente elimina un objeto denominado de un bucket denominado my-bucket:

```
aws s3api delete-objects --bucket my-bucket --delete file://delete.json
```

`delete.json` es un JSON documento del directorio actual que especifica el objeto que se va a eliminar:

```
{
  "Objects": [
    {
      "Key": "test1.txt"
    }
  ],
  "Quiet": false
}
```

Salida:

```
{
  "Deleted": [
    {
      "DeleteMarkerVersionId": "mYAT5Mc6F7aeUL8SS7FAAqUP01koHwzU",
      "Key": "test1.txt",
      "DeleteMarker": true
    }
  ]
}
```

- Para API obtener más información, consulte [DeleteObjects](#) la Referencia de AWS CLI comandos.

## **delete-public-access-block**

En el siguiente ejemplo de código, se muestra cómo usar `delete-public-access-block`.

## AWS CLI

Eliminar la configuración de bloqueo de acceso público de un bucket

En el siguiente ejemplo de `delete-public-access-block`, se elimina la configuración de bloqueo de acceso público en el bucket especificado.

```
aws s3api delete-public-access-block \  
  --bucket my-bucket
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeletePublicAccessBlock](#) la Referencia de AWS CLI comandos.

## **get-bucket-accelerate-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-accelerate-configuration`.

### AWS CLI

Recuperar la configuración acelerada de un bucket

En el siguiente ejemplo de `get-bucket-accelerate-configuration`, se recupera la configuración acelerada para el bucket especificado.

```
aws s3api get-bucket-accelerate-configuration \  
  --bucket my-bucket
```

Salida:

```
{  
  "Status": "Enabled"  
}
```

- Para API obtener más información, consulte [GetBucketAccelerateConfiguration](#) la Referencia de AWS CLI comandos.

## **get-bucket-acl**

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-acl`.

## AWS CLI

El siguiente comando recupera la lista de control de acceso de un bucket denominado my-bucket:

```
aws s3api get-bucket-acl --bucket my-bucket
```

Salida:

```
{
  "Owner": {
    "DisplayName": "my-username",
    "ID": "7009a8971cd538e11f6b6606438875e7c86c5b672f46db45460ddcd087d36c32"
  },
  "Grants": [
    {
      "Grantee": {
        "DisplayName": "my-username",
        "ID": "7009a8971cd538e11f6b6606438875e7c86c5b672f46db45460ddcd087d36c32"
      },
      "Permission": "FULL_CONTROL"
    }
  ]
}
```

- Para API obtener más información, consulte [GetBucketAcl](#) la Referencia de AWS CLI comandos.

## get-bucket-analytics-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-analytics-configuration`.

## AWS CLI

Recuperar la configuración de análisis de un bucket con un ID específico

En el siguiente ejemplo de `get-bucket-analytics-configuration`, se muestra la configuración de análisis para el bucket e ID especificados.

```
aws s3api get-bucket-analytics-configuration \
```

```
--bucket my-bucket \  
--id 1
```

Salida:

```
{  
  "AnalyticsConfiguration": {  
    "StorageClassAnalysis": {},  
    "Id": "1"  
  }  
}
```

- Para API obtener más información, consulte [GetBucketAnalyticsConfiguration](#) la Referencia de AWS CLI comandos.

## get-bucket-cors

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-cors`.

AWS CLI

El siguiente comando recupera la configuración de uso compartido de recursos entre orígenes para un bucket denominado `my-bucket`:

```
aws s3api get-bucket-cors --bucket my-bucket
```

Salida:

```
{  
  "CORSRules": [  
    {  
      "AllowedHeaders": [  
        "*"   
      ],  
      "ExposeHeaders": [  
        "x-amz-server-side-encryption"  
      ],  
      "AllowedMethods": [  
        "PUT",  
        "POST",  
        "DELETE"  
      ],  
    }  
  ]  
}
```

```

        "MaxAgeSeconds": 3000,
        "AllowedOrigins": [
            "http://www.example.com"
        ]
    },
    {
        "AllowedHeaders": [
            "Authorization"
        ],
        "MaxAgeSeconds": 3000,
        "AllowedMethods": [
            "GET"
        ],
        "AllowedOrigins": [
            "*"
        ]
    }
]
}

```

- Para API obtener más información, consulte [GetBucketCors](#) la Referencia de AWS CLI comandos.

## get-bucket-encryption

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-encryption`.

### AWS CLI

Recuperar la configuración de cifrado del servidor de un bucket

En el siguiente ejemplo de `get-bucket-encryption`, se recupera la configuración de cifrado del lado del servidor del bucket `my-bucket`.

```
aws s3api get-bucket-encryption \
  --bucket my-bucket
```

Salida:

```
{
  "ServerSideEncryptionConfiguration": {
    "Rules": [
```

```

        {
            "ApplyServerSideEncryptionByDefault": {
                "SSEAlgorithm": "AES256"
            }
        }
    ]
}

```

- Para API obtener más información, consulte [GetBucketEncryption](#) la Referencia de AWS CLI comandos.

## get-bucket-intelligent-tiering-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-intelligent-tiering-configuration`.

### AWS CLI

Para recuperar una configuración de S3 Intelligent-Tiering en un bucket

El siguiente `get-bucket-intelligent-tiering-configuration` ejemplo recupera una configuración de S3 Intelligent-Tiering, denominada, en un bucket. `ExampleConfig`

```

aws s3api get-bucket-intelligent-tiering-configuration \
  --bucket DOC-EXAMPLE-BUCKET \
  --id ExampleConfig

```

Salida:

```

{
  "IntelligentTieringConfiguration": {
    "Id": "ExampleConfig2",
    "Filter": {
      "Prefix": "images"
    },
    "Status": "Enabled",
    "Tierings": [
      {
        "Days": 90,
        "AccessTier": "ARCHIVE_ACCESS"
      },
    ],
  },
}

```

```
    {
      "Days": 180,
      "AccessTier": "DEEP_ARCHIVE_ACCESS"
    }
  ]
}
```

Para obtener más información, consulte [Uso de S3 Intelligent-Tiering](#) en la Guía del usuario de Amazon S3.

- Para obtener API más información, consulte la Referencia de comandos [GetBucketIntelligentTieringConfiguration](#).AWS CLI

## get-bucket-inventory-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-inventory-configuration`.

### AWS CLI

Recuperar la configuración de inventario de un bucket

En el siguiente ejemplo de `get-bucket-inventory-configuration`, se recupera la configuración de inventario del bucket especificado con el ID 1.

```
aws s3api get-bucket-inventory-configuration \
  --bucket my-bucket \
  --id 1
```

Salida:

```
{
  "InventoryConfiguration": {
    "IsEnabled": true,
    "Destination": {
      "S3BucketDestination": {
        "Format": "ORC",
        "Bucket": "arn:aws:s3:::my-bucket",
        "AccountId": "123456789012"
      }
    }
  },
}
```



```

    "IncludedObjectVersions": "Current",
    "Id": "1",
    "Schedule": {
      "Frequency": "Weekly"
    }
  }
}

```

- Para API obtener más información, consulte [GetBucketInventoryConfiguration](#) la Referencia de AWS CLI comandos.

## get-bucket-lifecycle-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-lifecycle-configuration`.

### AWS CLI

El siguiente comando recupera la configuración del ciclo de vida de un bucket denominado `my-bucket`:

```
aws s3api get-bucket-lifecycle-configuration --bucket my-bucket
```

Salida:

```

{
  "Rules": [
    {
      "ID": "Move rotated logs to Glacier",
      "Prefix": "rotated/",
      "Status": "Enabled",
      "Transitions": [
        {
          "Date": "2015-11-10T00:00:00.000Z",
          "StorageClass": "GLACIER"
        }
      ]
    },
    {
      "Status": "Enabled",
      "Prefix": "",
      "NoncurrentVersionTransitions": [

```

```
        {
            "NoncurrentDays": 0,
            "StorageClass": "GLACIER"
        }
    ],
    "ID": "Move old versions to Glacier"
}
]
```

- Para API obtener más información, consulte [GetBucketLifecycleConfiguration](#) la Referencia de AWS CLI comandos.

## get-bucket-lifecycle

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-lifecycle`.

### AWS CLI

El siguiente comando recupera la configuración del ciclo de vida de un bucket denominado `my-bucket`:

```
aws s3api get-bucket-lifecycle --bucket my-bucket
```

Salida:

```
{
  "Rules": [
    {
      "ID": "Move to Glacier after sixty days (objects in logs/2015/)",
      "Prefix": "logs/2015/",
      "Status": "Enabled",
      "Transition": {
        "Days": 60,
        "StorageClass": "GLACIER"
      }
    },
    {
      "Expiration": {
        "Date": "2016-01-01T00:00:00.000Z"
      },
      "ID": "Delete 2014 logs in 2016.",
    }
  ]
}
```

```
    "Prefix": "logs/2014/",
    "Status": "Enabled"
  }
]
```

- Para API obtener más información, consulte [GetBucketLifecycle](#) la Referencia de AWS CLI comandos.

## get-bucket-location

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-location`.

### AWS CLI

El siguiente comando recupera la restricción de ubicación de un bucket denominado `my-bucket`, si existe una restricción:

```
aws s3api get-bucket-location --bucket my-bucket
```

Salida:

```
{
  "LocationConstraint": "us-west-2"
}
```

- Para API obtener más información, consulte [GetBucketLocation](#) la Referencia de AWS CLI comandos.

## get-bucket-logging

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-logging`.

### AWS CLI

Recuperar el estado de registros de un bucket

En el siguiente ejemplo de `get-bucket-logging`, se recupera el estado de registros del bucket especificado.

```
aws s3api get-bucket-logging \
```

```
--bucket my-bucket
```

Salida:

```
{
  "LoggingEnabled": {
    "TargetPrefix": "",
    "TargetBucket": "my-bucket-logs"
  }
}
```

- Para API obtener más información, consulte [GetBucketLogging](#) la Referencia de AWS CLI comandos.

## get-bucket-metrics-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-metrics-configuration`.

AWS CLI

Recuperar la configuración de métricas de un bucket con un ID específico

En el siguiente ejemplo de `get-bucket-metrics-configuration`, se muestra la configuración de métricas para el bucket e ID especificados.

```
aws s3api get-bucket-metrics-configuration \
  --bucket my-bucket \
  --id 123
```

Salida:

```
{
  "MetricsConfiguration": {
    "Filter": {
      "Prefix": "logs"
    },
    "Id": "123"
  }
}
```

- Para API obtener más información, consulte [GetBucketMetricsConfiguration](#) la Referencia de AWS CLI comandos.

## get-bucket-notification-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-notification-configuration`.

### AWS CLI

El siguiente comando recupera la configuración de notificaciones de un bucket denominado `my-bucket`:

```
aws s3api get-bucket-notification-configuration --bucket my-bucket
```

Salida:

```
{
  "TopicConfigurations": [
    {
      "Id": "YmQzMmEwM2EjZWVlI0NGItNzVtZjI1MC00ZjgyLWZDBiZWw1",
      "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-notification-topic",
      "Events": [
        "s3:ObjectCreated:*"
      ]
    }
  ]
}
```

- Para API obtener más información, consulte [GetBucketNotificationConfiguration](#) la Referencia de AWS CLI comandos.

## get-bucket-notification

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-notification`.

### AWS CLI

El siguiente comando recupera la configuración de notificaciones de un bucket denominado `my-bucket`:

```
aws s3api get-bucket-notification --bucket my-bucket
```

Salida:

```
{
  "TopicConfiguration": {
    "Topic": "arn:aws:sns:us-west-2:123456789012:my-notification-topic",
    "Id": "YmQzMmEwM2EjZWVlI0NGItNzVtZjI1MC00ZjgyLWZDBiZWw1",
    "Event": "s3:ObjectCreated:*",
    "Events": [
      "s3:ObjectCreated:*"
    ]
  }
}
```

- Para API obtener más información, consulte [GetBucketNotification](#) la Referencia de AWS CLI comandos.

## get-bucket-ownership-controls

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-ownership-controls`.

AWS CLI

Para recuperar la configuración de propiedad de un bucket

En el siguiente `get-bucket-ownership-controls` ejemplo, se recupera la configuración de propiedad de un bucket.

```
aws s3api get-bucket-ownership-controls \
  --bucket DOC-EXAMPLE-BUCKET
```

Salida:

```
{
  "OwnershipControls": {
    "Rules": [
      {
        "ObjectOwnership": "BucketOwnerEnforced"
      }
    ]
  }
}
```

```
}  
}
```

Para obtener más información, consulte [Visualización de la configuración de propiedad del objeto para un bucket de S3](#) en la Guía del usuario de Amazon S3.

- Para API obtener más información, consulte [GetBucketOwnershipControls](#) la Referencia de AWS CLI comandos.

## get-bucket-policy-status

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-policy-status`.

### AWS CLI

Recuperar el estado de política de un bucket que indica si el bucket es público

En el siguiente ejemplo de `get-bucket-policy-status`, se recupera el estado de política del bucket `my-bucket`.

```
aws s3api get-bucket-policy-status \  
  --bucket my-bucket
```

Salida:

```
{  
  "PolicyStatus": {  
    "IsPublic": false  
  }  
}
```

- Para API obtener más información, consulte [GetBucketPolicyStatus](#) la Referencia de AWS CLI comandos.

## get-bucket-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-policy`.

### AWS CLI

El siguiente comando recupera la política de bucket de un bucket denominado `my-bucket`:

```
aws s3api get-bucket-policy --bucket my-bucket
```

Salida:

```
{
  "Policy": "{\"Version\":\"2008-10-17\",\"Statement\": [{\"Sid\":\"\", \"Effect\": \"Allow\", \"Principal\": \"*\", \"Action\": \"s3:GetObject\", \"Resource\": \"arn:aws:s3:::my-bucket/*\"}, {\"Sid\":\"\", \"Effect\": \"Deny\", \"Principal\": \"*\", \"Action\": \"s3:GetObject\", \"Resource\": \"arn:aws:s3:::my-bucket/secret/*\"}]}"
}
```

El policyThe siguiente ejemplo de Get and put a bucket muestra cómo puede descargar una política de bucket de Amazon S3, realizar modificaciones en el archivo y, después, utilizarla put-bucket-policy para aplicar la política de bucket modificada. Para descargar la política de bucket a un archivo, puede ejecutar:

```
aws s3api get-bucket-policy --bucket mybucket --query Policy --output text >
policy.json
```

A continuación, puede modificar el archivo policy.json según sea necesario. Por último, puede volver a aplicar esta política modificada al bucket de S3 ejecutando:

archivo policy.json según sea necesario. Por último, puede volver a aplicar esta política modificada al bucket de S3 ejecutando:

archivo según sea necesario. Por último, puede volver a aplicar esta política modificada al bucket de S3 ejecutando:

```
aws s3api put-bucket-policy --bucket mybucket --policy file://policy.json
```

- Para API obtener más información, consulte [GetBucketPolicy](#) la Referencia de AWS CLI comandos.

## get-bucket-replication

En el siguiente ejemplo de código, se muestra cómo usar get-bucket-replication.



## AWS CLI

El siguiente comando recupera la configuración de replicación de un bucket denominado my-bucket:

```
aws s3api get-bucket-replication --bucket my-bucket
```

Salida:

```
{
  "ReplicationConfiguration": {
    "Rules": [
      {
        "Status": "Enabled",
        "Prefix": "",
        "Destination": {
          "Bucket": "arn:aws:s3:::my-bucket-backup",
          "StorageClass": "STANDARD"
        },
        "ID": "ZmUwNzE4ZmQ4tMjVhOS00MTlkLOGI4NDkzZTIWJjNTUtYTA1"
      }
    ],
    "Role": "arn:aws:iam::123456789012:role/s3-replication-role"
  }
}
```

- Para API obtener más información, consulte [GetBucketReplication](#) la Referencia de AWS CLI comandos.

## get-bucket-request-payment

En el siguiente ejemplo de código, se muestra cómo usar get-bucket-request-payment.

### AWS CLI

Recuperar la configuración de pagos de solicitudes de un bucket

En el siguiente ejemplo de get-bucket-request-payment, se recupera la configuración de pagos por el solicitante para el bucket especificado.

```
aws s3api get-bucket-request-payment \
```

```
--bucket my-bucket
```

Salida:

```
{
  "Payer": "BucketOwner"
}
```

- Para API obtener más información, consulte [GetBucketRequestPayment](#) la Referencia de AWS CLI comandos.

## get-bucket-tagging

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-tagging`.

AWS CLI

El siguiente comando recupera la configuración de etiquetado de un bucket denominado `my-bucket`:

```
aws s3api get-bucket-tagging --bucket my-bucket
```

Salida:

```
{
  "TagSet": [
    {
      "Value": "marketing",
      "Key": "organization"
    }
  ]
}
```

- Para API obtener más información, consulte [GetBucketTagging](#) la Referencia de AWS CLI comandos.

## get-bucket-versioning

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-versioning`.

## AWS CLI

El siguiente comando recupera la configuración del control de versiones de un bucket denominado `my-bucket`:

```
aws s3api get-bucket-versioning --bucket my-bucket
```

Salida:

```
{
  "Status": "Enabled"
}
```

- Para API obtener más información, consulte [GetBucketVersioning](#) la Referencia de AWS CLI comandos.

## get-bucket-website

En el siguiente ejemplo de código, se muestra cómo usar `get-bucket-website`.

## AWS CLI

El siguiente comando recupera la configuración de sitio web estática de un bucket denominado `my-bucket`:

```
aws s3api get-bucket-website --bucket my-bucket
```

Salida:

```
{
  "IndexDocument": {
    "Suffix": "index.html"
  },
  "ErrorDocument": {
    "Key": "error.html"
  }
}
```

- Para API obtener más información, consulte [GetBucketWebsite](#) la Referencia de AWS CLI comandos.

## get-object-acl

En el siguiente ejemplo de código, se muestra cómo usar `get-object-acl`.

### AWS CLI

El siguiente comando recupera la lista de control de acceso de un objeto en un bucket denominado `my-bucket`:

```
aws s3api get-object-acl --bucket my-bucket --key index.html
```

Salida:

```
{
  "Owner": {
    "DisplayName": "my-username",
    "ID": "7009a8971cd538e11f6b6606438875e7c86c5b672f46db45460ddcd087d36c32"
  },
  "Grants": [
    {
      "Grantee": {
        "DisplayName": "my-username",
        "ID": "7009a8971cd538e11f6b6606438875e7c86c5b672f46db45460ddcd087d36c32"
      },
      "Permission": "FULL_CONTROL"
    },
    {
      "Grantee": {
        "URI": "http://acs.amazonaws.com/groups/global/AllUsers"
      },
      "Permission": "READ"
    }
  ]
}
```

- Para API obtener más información, consulte [GetObjectAcl](#) la Referencia de AWS CLI comandos.

## get-object-attributes

En el siguiente ejemplo de código, se muestra cómo usar `get-object-attributes`.

## AWS CLI

Para recuperar metadatos de un objeto sin devolver el objeto en sí

En el siguiente ejemplo de `get-object-attributes`, se recuperan los metadatos del objeto `doc1.rtf`.

```
aws s3api get-object-attributes \  
  --bucket my-bucket \  
  --key doc1.rtf \  
  --object-attributes "StorageClass" "Etag" "ObjectSize"
```

Salida:

```
{  
  "LastModified": "2022-03-15T19:37:31+00:00",  
  "VersionId": "IuCPjXTDzHNfldAuitVBIKJpF2p1fg4P",  
  "ETag": "b662d79adeb7c8d787ea7eafb9ef6207",  
  "StorageClass": "STANDARD",  
  "ObjectSize": 405  
}
```

Para obtener más información, consulte [GetObjectAttributes](#) la API referencia de Amazon S3.

- Para API obtener más información, consulte [GetObjectAttributes](#) la Referencia de AWS CLI comandos.

## get-object-legal-hold

En el siguiente ejemplo de código, se muestra cómo usar `get-object-legal-hold`.

### AWS CLI

Recupera el estado de retención legal de un objeto

En el siguiente ejemplo de `get-object-legal-hold`, se recupera el estado de retención legal del objeto especificado.

```
aws s3api get-object-legal-hold \  
  --bucket my-bucket-with-object-lock \  
  --key doc1.rtf
```

Salida:

```
{
  "LegalHold": {
    "Status": "ON"
  }
}
```

- Para API obtener más información, consulte [GetObjectLegalHold](#) la Referencia de AWS CLI comandos.

## get-object-lock-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-object-lock-configuration`.

AWS CLI

Para recuperar una configuración de bloqueo de objetos para un bucket

En el siguiente ejemplo de `get-object-lock-configuration`, se recupera la configuración de bloqueo de objetos para el bucket especificado.

```
aws s3api get-object-lock-configuration \
  --bucket my-bucket-with-object-lock
```

Salida:

```
{
  "ObjectLockConfiguration": {
    "ObjectLockEnabled": "Enabled",
    "Rule": {
      "DefaultRetention": {
        "Mode": "COMPLIANCE",
        "Days": 50
      }
    }
  }
}
```

- Para API obtener más información, consulte [GetObjectLockConfiguration](#) la Referencia de AWS CLI comandos.

## get-object-retention

En el siguiente ejemplo de código, se muestra cómo usar `get-object-retention`.

### AWS CLI

Para recuperar la configuración de retención de un objeto

En el siguiente ejemplo de `get-object-retention`, se recupera la configuración de retención del objeto especificado.

```
aws s3api get-object-retention \  
  --bucket my-bucket-with-object-lock \  
  --key doc1.rtf
```

Salida:

```
{  
  "Retention": {  
    "Mode": "GOVERNANCE",  
    "RetainUntilDate": "2025-01-01T00:00:00.000Z"  
  }  
}
```

- Para API obtener más información, consulte [GetObjectRetention](#) la Referencia de AWS CLI comandos.

## get-object-tagging

En el siguiente ejemplo de código, se muestra cómo usar `get-object-tagging`.

### AWS CLI

Recuperar las etiquetas asociadas a un objeto

El siguiente ejemplo de `get-object-tagging` recupera los valores de la clave especificada del objeto especificado.

```
aws s3api get-object-tagging \  
  --bucket my-bucket \  
  --key doc1.rtf
```

Salida:

```
{
  "TagSet": [
    {
      "Value": "confidential",
      "Key": "designation"
    }
  ]
}
```

El siguiente ejemplo de `get-object-tagging` intenta recuperar los conjuntos de etiquetas del objeto `doc2.rtf`, que no tiene etiquetas.

```
aws s3api get-object-tagging \
  --bucket my-bucket \
  --key doc2.rtf
```

Salida:

```
{
  "TagSet": []
}
```

El siguiente ejemplo de `get-object-tagging` recupera los conjuntos de etiquetas del objeto `doc3.rtf`, que tiene varias etiquetas.

```
aws s3api get-object-tagging \
  --bucket my-bucket \
  --key doc3.rtf
```

Salida:

```
{
  "TagSet": [
    {
      "Value": "confidential",
      "Key": "designation"
    },
    {
      "Value": "finance",
```



```
        "Key": "department"
      },
      {
        "Value": "payroll",
        "Key": "team"
      }
    ]
  }
}
```

- Para API obtener más información, consulte [GetObjectTagging](#) la Referencia de AWS CLI comandos.

## get-object-torrent

En el siguiente ejemplo de código, se muestra cómo usar `get-object-torrent`.

### AWS CLI

El siguiente comando crea un torrente para un objeto de un bucket denominado `my-bucket`:

```
aws s3api get-object-torrent --bucket my-bucket --key large-video-file.mp4 large-video-file.torrent
```

El archivo torrent se guarda localmente en la carpeta actual. Tenga en cuenta que el nombre del archivo de salida (`large-video-file.torrent`) se especifica sin un nombre de opción y debe ser el último argumento del comando.

- Para API obtener más información, consulte [GetObjectTorrent](#) la Referencia de AWS CLI comandos.

## get-object

En el siguiente ejemplo de código, se muestra cómo usar `get-object`.

### AWS CLI

En el siguiente ejemplo se utiliza el comando `get-object` para descargar un objeto de Amazon S3.

```
aws s3api get-object --bucket text-content --key dir/my_images.tar.bz2 my_images.tar.bz2
```

Tenga en cuenta que el parámetro `outfile` se especifica sin un nombre de opción, como `--outfile`. El nombre del archivo de salida debe ser el último parámetro del comando.

El siguiente ejemplo muestra el uso de `--range` para descargar un intervalo de bytes específico de un objeto. Tenga en cuenta que los intervalos de bytes deben tener el prefijo `bytes=`:

```
aws s3api get-object --bucket text-content --key dir/my_data --  
range bytes=8888-9999 my_data_range
```

Para obtener más información acerca de la recuperación de objetos, consulte [Obtención de objetos](#) en la Guía para desarrolladores de Amazon S3.

- Para API obtener más información, consulte [GetObject](#) la Referencia de AWS CLI comandos.

## get-public-access-block

En el siguiente ejemplo de código, se muestra cómo usar `get-public-access-block`.

### AWS CLI

Establecer o modificar la configuración de bloqueo de acceso público de un bucket

En el siguiente ejemplo de `get-public-access-block`, se elimina la configuración de bloqueo de acceso público del bucket especificado.

```
aws s3api get-public-access-block \  
--bucket my-bucket
```

Salida:

```
{  
  "PublicAccessBlockConfiguration": {  
    "IgnorePublicAcls": true,  
    "BlockPublicPolicy": true,  
    "BlockPublicAcls": true,  
    "RestrictPublicBuckets": true  
  }  
}
```

- Para API obtener más información, consulte [GetPublicAccessBlock](#) la Referencia de AWS CLI comandos.

## head-bucket

En el siguiente ejemplo de código, se muestra cómo usar `head-bucket`.

### AWS CLI

El siguiente comando verifica el acceso a un bucket denominado `my-bucket`:

```
aws s3api head-bucket --bucket my-bucket
```

Si el bucket existe y tiene acceso a él, no se muestra ningún resultado. De lo contrario, se mostrará un mensaje de error. Por ejemplo:

```
A client error (404) occurred when calling the HeadBucket operation: Not Found
```

- Para API obtener más información, consulte [HeadBucket](#) la Referencia de AWS CLI comandos.

## head-object

En el siguiente ejemplo de código, se muestra cómo usar `head-object`.

### AWS CLI

El siguiente comando recupera los metadatos de un objeto de un bucket denominado `my-bucket`.

```
aws s3api head-object --bucket my-bucket --key index.html
```

Salida:

```
{
  "AcceptRanges": "bytes",
  "ContentType": "text/html",
  "LastModified": "Thu, 16 Apr 2015 18:19:14 GMT",
  "ContentLength": 77,
  "VersionId": "null",
  "ETag": "\"30a6ec7e1a9ad79c203d05a589c8b400\"",
  "Metadata": {}
}
```

- Para API obtener más información, consulte [HeadObject](#) la Referencia de AWS CLI comandos.

## list-bucket-analytics-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-bucket-analytics-configurations`.

### AWS CLI

Recuperar una lista de configuraciones de análisis para un bucket

El siguiente `list-bucket-analytics-configurations` recupera una lista de configuraciones de análisis para el bucket especificado.

```
aws s3api list-bucket-analytics-configurations \
  --bucket my-bucket
```

Salida:

```
{
  "AnalyticsConfigurationList": [
    {
      "StorageClassAnalysis": {},
      "Id": "1"
    }
  ],
  "IsTruncated": false
}
```

- Para API obtener más información, consulte [ListBucketAnalyticsConfigurations](#) la Referencia de AWS CLI comandos.

## list-bucket-intelligent-tiering-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-bucket-intelligent-tiering-configurations`.

### AWS CLI

Para recuperar todas las configuraciones de S3 Intelligent-Tiering en un bucket

El siguiente `list-bucket-intelligent-tiering-configurations` ejemplo recupera toda la configuración de S3 Intelligent-Tiering de un bucket.

```
aws s3api list-bucket-intelligent-tiering-configurations \  
--bucket DOC-EXAMPLE-BUCKET
```

Salida:

```
{  
  "IsTruncated": false,  
  "IntelligentTieringConfigurationList": [  
    {  
      "Id": "ExampleConfig",  
      "Filter": {  
        "Prefix": "images"  
      },  
      "Status": "Enabled",  
      "Tierings": [  
        {  
          "Days": 90,  
          "AccessTier": "ARCHIVE_ACCESS"  
        },  
        {  
          "Days": 180,  
          "AccessTier": "DEEP_ARCHIVE_ACCESS"  
        }  
      ]  
    },  
    {  
      "Id": "ExampleConfig2",  
      "Status": "Disabled",  
      "Tierings": [  
        {  
          "Days": 730,  
          "AccessTier": "ARCHIVE_ACCESS"  
        }  
      ]  
    },  
    {  
      "Id": "ExampleConfig3",  
      "Filter": {  
        "Tag": {  
          "Key": "documents",  
          "Value": "taxes"  
        }  
      },  
    }  
  ]  
}
```

```
    "Status": "Enabled",
    "Tierings": [
      {
        "Days": 90,
        "AccessTier": "ARCHIVE_ACCESS"
      },
      {
        "Days": 365,
        "AccessTier": "DEEP_ARCHIVE_ACCESS"
      }
    ]
  }
]
```

Para obtener más información, consulte [Uso de S3 Intelligent-Tiering](#) en la Guía del usuario de Amazon S3.

- Para obtener API más información, consulte la Referencia de comandos [ListBucketIntelligentTieringConfigurations](#).AWS CLI

## list-bucket-inventory-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-bucket-inventory-configurations`.

### AWS CLI

Recuperar una lista de las configuraciones de inventario de un bucket

En el siguiente ejemplo de `list-bucket-inventory-configurations`, se enumeran las configuraciones de inventario del bucket especificado.

```
aws s3api list-bucket-inventory-configurations \
  --bucket my-bucket
```

Salida:

```
{
  "InventoryConfigurationList": [
    {
      "IsEnabled": true,
```

```

    "Destination": {
      "S3BucketDestination": {
        "Format": "ORC",
        "Bucket": "arn:aws:s3:::my-bucket",
        "AccountId": "123456789012"
      }
    },
    "IncludedObjectVersions": "Current",
    "Id": "1",
    "Schedule": {
      "Frequency": "Weekly"
    }
  },
  {
    "IsEnabled": true,
    "Destination": {
      "S3BucketDestination": {
        "Format": "CSV",
        "Bucket": "arn:aws:s3:::my-bucket",
        "AccountId": "123456789012"
      }
    },
    "IncludedObjectVersions": "Current",
    "Id": "2",
    "Schedule": {
      "Frequency": "Daily"
    }
  }
],
  "IsTruncated": false
}

```

- Para API obtener más información, consulte [ListBucketInventoryConfigurations](#) la Referencia de AWS CLI comandos.

## list-bucket-metrics-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-bucket-metrics-configurations`.

### AWS CLI

Para recuperar una lista de configuraciones de métricas para un bucket

En el siguiente `list-bucket-metrics-configurations` ejemplo, se recupera una lista de configuraciones de métricas para el depósito especificado.

```
aws s3api list-bucket-metrics-configurations \  
  --bucket my-bucket
```

Salida:

```
{  
  "IsTruncated": false,  
  "MetricsConfigurationList": [  
    {  
      "Filter": {  
        "Prefix": "logs"  
      },  
      "Id": "123"  
    },  
    {  
      "Filter": {  
        "Prefix": "tmp"  
      },  
      "Id": "234"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListBucketMetricsConfigurations](#) la Referencia de AWS CLI comandos.

## list-buckets

En el siguiente ejemplo de código, se muestra cómo usar `list-buckets`.

AWS CLI

El siguiente comando usa el comando `list-buckets` para mostrar los nombres de todos los buckets de Amazon S3 (en todas las regiones):

```
aws s3api list-buckets --query "Buckets[].Name"
```



La opción de consultas filtra la salida de `list-buckets` únicamente a los nombres de los buckets.

Para obtener más información sobre los buckets, consulte Trabajo con buckets de Amazon S3 en la Guía para desarrolladores de Amazon S3.

- Para API obtener más información, consulte [ListBuckets](#) la Referencia de AWS CLI comandos.

## `list-multipart-uploads`

En el siguiente ejemplo de código, se muestra cómo usar `list-multipart-uploads`.

### AWS CLI

El siguiente comando muestra todas las cargas multiparte activas de un bucket denominado `my-bucket`:

```
aws s3api list-multipart-uploads --bucket my-bucket
```

Salida:

```
{
  "Uploads": [
    {
      "Initiator": {
        "DisplayName": "username",
        "ID": "arn:aws:iam::0123456789012:user/username"
      },
      "Initiated": "2015-06-02T18:01:30.000Z",
      "UploadId":
      "dfRtDYU0WWCCcH43C3WFbkR0NycyCpTJJvxu2i5GYkZljF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3URC",
      "StorageClass": "STANDARD",
      "Key": "multipart/01",
      "Owner": {
        "DisplayName": "aws-account-name",
        "ID":
        "100719349fc3b6dcd7c820a124bf7aec408092c3d7b51b38494939801fc248b"
      }
    },
    ],
  "CommonPrefixes": []
}
```

Las cargas multiparte en curso conllevan costos de almacenamiento en Amazon S3. Complete o anule una carga multiparte activa para eliminar sus partes de su cuenta.

- Para API obtener más información, consulte [ListMultipartUploads](#) la Referencia de AWS CLI comandos.

## list-object-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-object-versions`.

### AWS CLI

El siguiente comando recupera la información de la versión de un objeto en un bucket denominado `my-bucket`:

```
aws s3api list-object-versions --bucket my-bucket --prefix index.html
```

Salida:

```
{
  "DeleteMarkers": [
    {
      "Owner": {
        "DisplayName": "my-username",
        "ID":
"7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
      },
      "IsLatest": true,
      "VersionId": "B2VsEK5saUNNHKc0AJj7hIE86RozToyq",
      "Key": "index.html",
      "LastModified": "2015-11-10T00:57:03.000Z"
    },
    {
      "Owner": {
        "DisplayName": "my-username",
        "ID":
"7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
      },
      "IsLatest": false,
      "VersionId": ".FLQEZscLIcfxSq.jsFJ.szUkmng2Yw6",
      "Key": "index.html",
      "LastModified": "2015-11-09T23:32:20.000Z"
    }
  ]
}
```

```
],
"Versions": [
  {
    "LastModified": "2015-11-10T00:20:11.000Z",
    "VersionId": "Rb_l2T8UHDkFEwCgJjhlgPOZC0qJ.vpD",
    "ETag": "\"0622528de826c0df5db1258a23b80be5\"",
    "StorageClass": "STANDARD",
    "Key": "index.html",
    "Owner": {
      "DisplayName": "my-username",
      "ID":
"7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
    },
    "IsLatest": false,
    "Size": 38
  },
  {
    "LastModified": "2015-11-09T23:26:41.000Z",
    "VersionId": "rasWWGpgk9E4s0LyTJgusGeRQKLVIAff",
    "ETag": "\"06225825b8028de826c0df5db1a23be5\"",
    "StorageClass": "STANDARD",
    "Key": "index.html",
    "Owner": {
      "DisplayName": "my-username",
      "ID":
"7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
    },
    "IsLatest": false,
    "Size": 38
  },
  {
    "LastModified": "2015-11-09T22:50:50.000Z",
    "VersionId": "null",
    "ETag": "\"d1f45267a863c8392e07d24dd592f1b9\"",
    "StorageClass": "STANDARD",
    "Key": "index.html",
    "Owner": {
      "DisplayName": "my-username",
      "ID":
"7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
    },
    "IsLatest": false,
    "Size": 533823
  }
]
```

```
]
}
```

- Para API obtener más información, consulte [ListObjectVersions](#) la Referencia de AWS CLI comandos.

## list-objects-v2

En el siguiente ejemplo de código, se muestra cómo usar `list-objects-v2`.

### AWS CLI

Obtención de una lista de objetos en un bucket

En el siguiente ejemplo de `list-objects-v2` se muestran los objetos del bucket especificado.

```
aws s3api list-objects-v2 \
  --bucket my-bucket
```

Salida:

```
{
  "Contents": [
    {
      "LastModified": "2019-11-05T23:11:50.000Z",
      "ETag": "\"621503c373607d548b37cff8778d992c\"",
      "StorageClass": "STANDARD",
      "Key": "doc1.rtf",
      "Size": 391
    },
    {
      "LastModified": "2019-11-05T23:11:50.000Z",
      "ETag": "\"a2cecc36ab7c7fe3a71a273b9d45b1b5\"",
      "StorageClass": "STANDARD",
      "Key": "doc2.rtf",
      "Size": 373
    },
    {
      "LastModified": "2019-11-05T23:11:50.000Z",
      "ETag": "\"08210852f65a2e9cb999972539a64d68\"",
      "StorageClass": "STANDARD",
```

```
        "Key": "doc3.rtf",
        "Size": 399
    },
    {
        "LastModified": "2019-11-05T23:11:50.000Z",
        "ETag": "\"d1852dd683f404306569471af106988e\"",
        "StorageClass": "STANDARD",
        "Key": "doc4.rtf",
        "Size": 6225
    }
]
}
```

- Para API obtener más información, consulte la sección [ListObjectsV2](#) en la Referencia de AWS CLI comandos.

## list-objects

En el siguiente ejemplo de código, se muestra cómo usar `list-objects`.

### AWS CLI

En el siguiente ejemplo se utiliza el comando `list-objects` para mostrar los nombres de todos los objetos del bucket especificado:

```
aws s3api list-objects --bucket text-content --query 'Contents[].{Key: Key, Size: Size}'
```

En el ejemplo se utiliza el argumento `--query` para filtrar la salida de `list-objects` hasta el valor de la clave y el tamaño de cada objeto

Para obtener más información sobre los objetos, consulte Trabajo con objetos de Amazon S3 en la Guía para desarrolladores de Amazon S3.

- Para API obtener más información, consulte [ListObjects](#) la Referencia de AWS CLI comandos.

## list-parts

En el siguiente ejemplo de código, se muestra cómo usar `list-parts`.

## AWS CLI

El siguiente comando muestra todas las partes que se han cargado para una carga multiparte con la clave `multipart/01` del depósito `my-bucket`:

```
aws s3api list-parts --bucket my-bucket --key 'multipart/01' --upload-id dfRtDYU0WCCcH43C3WfbkRONycyCpTJJvxu2i5GYkZLjF.Yxwh6XG7WfS2vC4to6HiV6YjLx.cph0gtNBtJ8P3UR
```

Salida:

```
{
  "Owner": {
    "DisplayName": "aws-account-name",
    "ID": "100719349fc3b6dcd7c820a124bf7aec408092c3d7b51b38494939801fc248b"
  },
  "Initiator": {
    "DisplayName": "username",
    "ID": "arn:aws:iam::0123456789012:user/username"
  },
  "Parts": [
    {
      "LastModified": "2015-06-02T18:07:35.000Z",
      "PartNumber": 1,
      "ETag": "\"e868e0f4719e394144ef36531ee6824c\"",
      "Size": 5242880
    },
    {
      "LastModified": "2015-06-02T18:07:42.000Z",
      "PartNumber": 2,
      "ETag": "\"6bb2b12753d66fe86da4998aa33fffb0\"",
      "Size": 5242880
    },
    {
      "LastModified": "2015-06-02T18:07:47.000Z",
      "PartNumber": 3,
      "ETag": "\"d0a0112e841abec9c9ec83406f0159c8\"",
      "Size": 5242880
    }
  ],
  "StorageClass": "STANDARD"
}
```

- Para API obtener más información, consulte [ListParts](#) la Referencia de AWS CLI comandos.

## ls

En el siguiente ejemplo de código, se muestra cómo usar `ls`.

### AWS CLI

Ejemplo 1: Listar todos los depósitos propiedad de los usuarios

El siguiente `ls` comando muestra todos los depósitos que son propiedad del usuario. En este ejemplo, el usuario es propietario de los depósitos `mybucket` y `mybucket2`. La marca de tiempo es la fecha en que se creó el depósito y se muestra en la zona horaria de su máquina. Esta fecha puede cambiar al realizar cambios en el bucket, por ejemplo, al editar su política de bucket. Ten en cuenta que si `s3://` se usa como argumento de ruta `<S3Uri>`, también se mostrarán todos los cubos.

```
aws s3 ls
```

Salida:

```
2013-07-11 17:08:50 mybucket
2013-07-24 14:55:44 mybucket2
```

Ejemplo 2: Listar todos los prefijos y objetos de un cubo

El siguiente `ls` comando muestra los objetos y los prefijos comunes incluidos en un depósito y un prefijo específicos. En este ejemplo, el usuario es propietario del depósito `mybucket` con los objetos `test.txt` y `somePrefix/test.txt`. Los `LastWriteTime` y `Length` son arbitrarios. Tenga en cuenta que, dado que el `ls` comando no interactúa con el sistema de archivos local, el `s3://` URI esquema no es necesario para resolver la ambigüedad y puede omitirse.

```
aws s3 ls s3://mybucket
```

Salida:

```
                PRE somePrefix/
2013-07-25 17:06:27      88 test.txt
```

Ejemplo 3: Listar todos los prefijos y objetos de un depósito y un prefijo específicos

El siguiente `ls` comando muestra los objetos y los prefijos comunes incluidos en un intervalo y un prefijo específicos. Sin embargo, no hay objetos ni prefijos comunes en el compartimento y el prefijo especificados.

```
aws s3 ls s3://mybucket/noExistPrefix
```

Salida:

```
None
```

Ejemplo 4: Listar de forma recursiva todos los prefijos y objetos de un depósito

El siguiente `ls` comando mostrará una lista recursiva de los objetos de un bucket. En lugar de mostrarse `PRE dirname/` en el resultado, todo el contenido de un depósito aparecerá en orden.

```
aws s3 ls s3://mybucket \  
--recursive
```

Salida:

```
2013-09-02 21:37:53      10 a.txt  
2013-09-02 21:37:53 2863288 foo.zip  
2013-09-02 21:32:57      23 foo/bar/.baz/a  
2013-09-02 21:32:58      41 foo/bar/.baz/b  
2013-09-02 21:32:57     281 foo/bar/.baz/c  
2013-09-02 21:32:57      73 foo/bar/.baz/d  
2013-09-02 21:32:57     452 foo/bar/.baz/e  
2013-09-02 21:32:57     896 foo/bar/.baz/hooks/bar  
2013-09-02 21:32:57     189 foo/bar/.baz/hooks/foo  
2013-09-02 21:32:57     398 z.txt
```

Ejemplo 5: resumir todos los prefijos y objetos de un depósito

El siguiente `ls` comando muestra el mismo comando con las opciones `--human-readable` y `--summarize`. `--human-legible` muestra el tamaño del archivo en bytes/MIB/KIB/GIB/TIB/PIB/EIB. `--summarize` muestra el número total de objetos y el tamaño total al final de la lista de resultados:

```
aws s3 ls s3://mybucket \  
--recursive \  
--human-readable \  
--summarize
```



```
--summarize
```

Salida:

```
2013-09-02 21:37:53  10 Bytes a.txt
2013-09-02 21:37:53 2.9 MiB foo.zip
2013-09-02 21:32:57  23 Bytes foo/bar/.baz/a
2013-09-02 21:32:58  41 Bytes foo/bar/.baz/b
2013-09-02 21:32:57 281 Bytes foo/bar/.baz/c
2013-09-02 21:32:57  73 Bytes foo/bar/.baz/d
2013-09-02 21:32:57 452 Bytes foo/bar/.baz/e
2013-09-02 21:32:57 896 Bytes foo/bar/.baz/hooks/bar
2013-09-02 21:32:57 189 Bytes foo/bar/.baz/hooks/foo
2013-09-02 21:32:57 398 Bytes z.txt

Total Objects: 10
Total Size: 2.9 MiB
```

Ejemplo 6: Publicar desde un punto de acceso S3

El siguiente `ls` comando muestra los objetos del punto de acceso (myaccesspoint):

```
aws s3 ls s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/
```

Salida:

```
                PRE somePrefix/
2013-07-25 17:06:27      88 test.txt
```

- Para API obtener más información, consulte [Ls](#) en la referencia de AWS CLI comandos.

## mb

En el siguiente ejemplo de código, se muestra cómo usar `mb`.

### AWS CLI

Ejemplo 1: Crear un depósito

El siguiente `mb` comando crea un depósito. En este ejemplo, el usuario crea el depósito `mybucket`. El depósito se crea en la región especificada en el archivo de configuración del usuario:

```
aws s3 mb s3://mybucket
```

Salida:

```
make_bucket: s3://mybucket
```

Ejemplo 2: Crear un depósito en la región especificada

El siguiente mb comando crea un depósito en una región especificada por el `--region` parámetro. En este ejemplo, el usuario crea el depósito mybucket en la región `us-west-1`:

```
aws s3 mb s3://mybucket \  
  --region us-west-1
```

Salida:

```
make_bucket: s3://mybucket
```

- Para API obtener más información, consulte [Mb](#) en la Referencia de AWS CLI comandos.

## mv

En el siguiente ejemplo de código, se muestra cómo usar mv.

### AWS CLI

Ejemplo 1: mover un archivo local al depósito especificado

El siguiente mv comando mueve un solo archivo a un depósito y una clave específicos.

```
aws s3 mv test.txt s3://mybucket/test2.txt
```

Salida:

```
move: test.txt to s3://mybucket/test2.txt
```

Ejemplo 2: Mover un objeto al depósito y la clave especificados

El siguiente mv comando mueve un único objeto s3 a un depósito y una clave específicos.

```
aws s3 mv s3://mybucket/test.txt s3://mybucket/test2.txt
```

Salida:

```
move: s3://mybucket/test.txt to s3://mybucket/test2.txt
```

Ejemplo 3: mover un objeto S3 al directorio local

El siguiente mv comando mueve un único objeto a un archivo especificado de forma local.

```
aws s3 mv s3://mybucket/test.txt test2.txt
```

Salida:

```
move: s3://mybucket/test.txt to test2.txt
```

Ejemplo 4: mueve un objeto con su nombre original al depósito especificado

El siguiente mv comando mueve un único objeto a un depósito específico sin perder su nombre original:

```
aws s3 mv s3://mybucket/test.txt s3://mybucket2/
```

Salida:

```
move: s3://mybucket/test.txt to s3://mybucket2/test.txt
```

Ejemplo 5: Mueva todos los objetos y prefijos de un depósito al directorio local

Cuando se pasa el parámetro `--recursive`, el siguiente mv comando mueve de forma recursiva todos los objetos situados bajo un prefijo y un depósito especificados a un directorio específico. En este ejemplo, el depósito `mybucket` contiene los objetos `test1.txt` y `test2.txt`

```
aws s3 mv s3://mybucket . \  
--recursive
```

Salida:

```
move: s3://mybucket/test1.txt to test1.txt
move: s3://mybucket/test2.txt to test2.txt
```

Ejemplo 6: Mueve todos los objetos y prefijos de un depósito al directorio local, excepto los archivos ``.jpg``

Cuando se pasa el parámetro `--recursive`, el siguiente `mv` comando mueve de forma recursiva todos los archivos de un directorio especificado a un depósito y un prefijo específicos, excluyendo algunos archivos mediante un parámetro. `--exclude` En este ejemplo, el directorio `myDir` contiene los archivos `y. test1.txt test2.jpg`

```
aws s3 mv myDir s3://mybucket/ \
  --recursive \
  --exclude "*.jpg"
```

Salida:

```
move: myDir/test1.txt to s3://mybucket2/test1.txt
```

Ejemplo 7: Mueva todos los objetos y prefijos de un depósito al directorio local, excepto el prefijo especificado

Cuando se pasa el parámetro `--recursive`, el siguiente `mv` comando mueve de forma recursiva todos los objetos de un depósito específico a otro depósito y, al mismo tiempo, excluye algunos objetos mediante un parámetro. `--exclude` En este ejemplo, el depósito `mybucket` contiene los objetos `test1.txt` `yanother/test1.txt`.

```
aws s3 mv s3://mybucket/ s3://mybucket2/ \
  --recursive \
  --exclude "mybucket/another/*"
```

Salida:

```
move: s3://mybucket/test1.txt to s3://mybucket2/test1.txt
```

Ejemplo 8: mueve un objeto al depósito especificado y establece el ACL

El siguiente mv comando mueve un solo objeto a un depósito y una tecla específicos mientras se establece ACL elpublic-read-write.

```
aws s3 mv s3://mybucket/test.txt s3://mybucket/test2.txt \  
--acl public-read-write
```

Salida:

```
move: s3://mybucket/test.txt to s3://mybucket/test2.txt
```

Ejemplo 9: Mover un archivo local al depósito especificado y conceder permisos

El siguiente mv comando ilustra el uso de la --grants opción para conceder acceso de lectura a todos los usuarios y el control total a un usuario específico identificado por su dirección de correo electrónico.

```
aws s3 mv file.txt s3://mybucket/ \  
--grants read=uri=http://acs.amazonaws.com/groups/global/  
AllUsers full=emailaddress=user@example.com
```

Salida:

```
move: file.txt to s3://mybucket/file.txt
```

Ejemplo 10: mover un archivo a un punto de acceso S3

El siguiente mv comando mueve un único archivo mydoc.txt con el nombre del punto de acceso indicado myaccesspoint en la clave denominada mykey.

```
aws s3 mv mydoc.txt s3://arn:aws:s3:us-west-2:123456789012:accesspoint/  
myaccesspoint/mykey
```

Salida:

```
move: mydoc.txt to s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/  
mykey
```

- Para API obtener más información, consulte [Mv](#) en la referencia de AWS CLI comandos.

## presign

En el siguiente ejemplo de código, se muestra cómo usar `presign`.

### AWS CLI

Ejemplo 1: Para crear un archivo prefirmado URL con una duración predeterminada de una hora que enlace a un objeto de un bucket de S3

El siguiente `presign` comando genera una clave y una contraseña prefirmadas URL para un depósito específico que son válidas durante una hora.

```
aws s3 presign s3://DOC-EXAMPLE-BUCKET/test2.txt
```

Salida:

```
https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com/key?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAEXAMPLE123456789%2F20210621%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-Date=20210621T041609Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=host&X-Amz-Signature=EXAMBLE1234494d5fba3fed607f98018e1dfc62e2529ae96d844123456
```

Ejemplo 2: Para crear un objeto prefirmado URL con una duración personalizada que se vincule a un objeto de un bucket de S3

El siguiente `presign` comando genera una clave y un depósito prefirmados URL que son válidos durante una semana.

```
aws s3 presign s3://DOC-EXAMPLE-BUCKET/test2.txt \
  --expires-in 604800
```

Salida:

```
https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com/key?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAEXAMPLE123456789%2F20210621%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-Date=20210621T041609Z&X-Amz-Expires=604800&X-Amz-SignedHeaders=host&X-Amz-Signature=EXAMBLE1234494d5fba3fed607f98018e1dfc62e2529ae96d844123456
```

Para obtener más información, consulte [Compartir un objeto con otros](#) en la guía para desarrolladores de S3.

- Para API obtener más información, consulte [Presign](#) in AWS CLI Command Reference.

## put-bucket-accelerate-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-accelerate-configuration`.

### AWS CLI

Establecer la configuración acelerada de un bucket

En el siguiente ejemplo de `put-bucket-accelerate-configuration`, se habilita la configuración acelerada para el bucket especificado.

```
aws s3api put-bucket-accelerate-configuration \  
  --bucket my-bucket \  
  --accelerate-configuration Status=Enabled
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutBucketAccelerateConfiguration](#) la Referencia de AWS CLI comandos.

## put-bucket-acl

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-acl`.

### AWS CLI

En este ejemplo `full control` se conceden `read` permisos a dos AWS usuarios (`user1@example.com` y `user2@example.com`) y a todos:

```
aws s3api put-bucket-acl --bucket MyBucket --grant-full-  
control emailaddress=user1@example.com,emailaddress=user2@example.com --grant-  
read uri=http://acs.amazonaws.com/groups/global/AllUsers
```

Consulte <http://docs.aws.amazon.com/AmazonS3/latest/API/RESTBucketPUTacl.html> para obtener más información sobre la personalización ACLs (los ACL comandos `s3api`, por ejemplo `put-bucket-acl`, utilizan la misma notación abreviada de argumentos).

- Para obtener API más información, consulte la Referencia de comandos. [PutBucketAclAWS CLI](#)

## put-bucket-analytics-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-analytics-configuration`.

### AWS CLI

Para establecer una configuración de análisis para el bucket

En el siguiente `put-bucket-analytics-configuration` ejemplo, se configuran los análisis para el depósito especificado.

```
aws s3api put-bucket-analytics-configuration \  
  --bucket my-bucket --id 1 \  
  --analytics-configuration '{"Id": "1", "StorageClassAnalysis": {}}'
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutBucketAnalyticsConfiguration](#) la Referencia de AWS CLI comandos.

## put-bucket-cors

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-cors`.

### AWS CLI

El siguiente ejemplo habilita solicitudes PUT, POST y DELETE desde `www.ejemplo.com`, y habilita solicitudes GET desde cualquier dominio:

```
aws s3api put-bucket-cors --bucket MyBucket --cors-configuration file://cors.json  
  
cors.json:  
{  
  "CORSRules": [  
    {  
      "AllowedOrigins": ["http://www.example.com"],  
      "AllowedHeaders": ["*"],
```



```

    "AllowedMethods": ["PUT", "POST", "DELETE"],
    "MaxAgeSeconds": 3000,
    "ExposeHeaders": ["x-amz-server-side-encryption"]
  },
  {
    "AllowedOrigins": ["*"],
    "AllowedHeaders": ["Authorization"],
    "AllowedMethods": ["GET"],
    "MaxAgeSeconds": 3000
  }
]
}

```

- Para API obtener más información, consulte [PutBucketCors](#) la Referencia de AWS CLI comandos.

## put-bucket-encryption

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-encryption`.

### AWS CLI

Configurar el cifrado del lado del servidor de un bucket

En el siguiente `put-bucket-encryption` ejemplo, se establece el AES256 cifrado como predeterminado para el depósito especificado.

```

aws s3api put-bucket-encryption \
  --bucket my-bucket \
  --server-side-encryption-configuration '{"Rules":
  [{"ApplyServerSideEncryptionByDefault": {"SSEAlgorithm": "AES256"}}]}'

```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutBucketEncryption](#) la Referencia de AWS CLI comandos.

## put-bucket-intelligent-tiering-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-intelligent-tiering-configuration`.

## AWS CLI

Para actualizar una configuración de S3 Intelligent-Tiering en un bucket

El siguiente `put-bucket-intelligent-tiering-configuration` ejemplo actualiza una configuración de S3 Intelligent-Tiering, denominada, en un bucket. `ExampleConfig` La configuración transferirá los objetos a los que no se haya accedido con el prefijo `images` a `Archive Access` después de 90 días y a `Deep Archive Access` después de 180 días.

```
aws s3api put-bucket-intelligent-tiering-configuration \  
  --bucket DOC-EXAMPLE-BUCKET \  
  --id "ExampleConfig" \  
  --intelligent-tiering-configuration file://intelligent-tiering-configuration.json
```

Contenidos de `intelligent-tiering-configuration.json`:

```
{  
  "Id": "ExampleConfig",  
  "Status": "Enabled",  
  "Filter": {  
    "Prefix": "images"  
  },  
  "Tierings": [  
    {  
      "Days": 90,  
      "AccessTier": "ARCHIVE_ACCESS"  
    },  
    {  
      "Days": 180,  
      "AccessTier": "DEEP_ARCHIVE_ACCESS"  
    }  
  ]  
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Establecer la propiedad de un objeto en un bucket existente](#) en la Guía del usuario de Amazon S3.

- Para API obtener más información, consulte [PutBucketIntelligentTieringConfiguration](#) la Referencia de AWS CLI comandos.

## put-bucket-inventory-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-inventory-configuration`.

### AWS CLI

Ejemplo 1: Para establecer una configuración de inventario para un depósito

El siguiente `put-bucket-inventory-configuration` ejemplo establece un informe ORC de inventario con formato semanal para el depósito. `my-bucket`

```
aws s3api put-bucket-inventory-configuration \  
  --bucket my-bucket \  
  --id 1 \  
  --inventory-configuration '{"Destination": { "S3BucketDestination":  
  { "AccountId": "123456789012", "Bucket": "arn:aws:s3:::my-bucket", "Format":  
  "ORC" }}, "IsEnabled": true, "Id": "1", "IncludedObjectVersions": "Current",  
  "Schedule": { "Frequency": "Weekly" } }'
```

Este comando no genera ninguna salida.

Ejemplo 2: Para establecer una configuración de inventario para un depósito

El siguiente `put-bucket-inventory-configuration` ejemplo establece un informe CSV de inventario con formato diario para el depósito. `my-bucket`

```
aws s3api put-bucket-inventory-configuration \  
  --bucket my-bucket \  
  --id 2 \  
  --inventory-configuration '{"Destination": { "S3BucketDestination":  
  { "AccountId": "123456789012", "Bucket": "arn:aws:s3:::my-bucket", "Format":  
  "CSV" }}, "IsEnabled": true, "Id": "2", "IncludedObjectVersions": "Current",  
  "Schedule": { "Frequency": "Daily" } }'
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte la Referencia [PutBucketInventoryConfiguration](#) de AWS CLI comandos.

## put-bucket-lifecycle-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-lifecycle-configuration`.

### AWS CLI

El comando siguiente aplica una configuración del ciclo de vida a un bucket denominado `my-bucket`:

```
aws s3api put-bucket-lifecycle-configuration --bucket my-bucket --lifecycle-configuration file://lifecycle.json
```

El archivo `lifecycle.json` es un JSON documento de la carpeta actual que especifica dos reglas:

```
{
  "Rules": [
    {
      "ID": "Move rotated logs to Glacier",
      "Prefix": "rotated/",
      "Status": "Enabled",
      "Transitions": [
        {
          "Date": "2015-11-10T00:00:00.000Z",
          "StorageClass": "GLACIER"
        }
      ]
    },
    {
      "Status": "Enabled",
      "Prefix": "",
      "NoncurrentVersionTransitions": [
        {
          "NoncurrentDays": 2,
          "StorageClass": "GLACIER"
        }
      ],
      "ID": "Move old versions to Glacier"
    }
  ]
}
```

La primera regla mueve los archivos con el prefijo `rotated` a Glacier en la fecha especificada. La segunda regla mueve las versiones del objeto antiguas a Glacier cuando ya no están actualizadas. Para obtener información sobre los formatos de marca de tiempo aceptables, consulte Especificación de valores de parámetros en la Guía del AWS CLI usuario.

- Para API obtener más información, consulte la Referencia [PutBucketLifecycleConfiguration](#) de AWS CLI comandos.

## put-bucket-lifecycle

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-lifecycle`.

### AWS CLI

El siguiente comando aplica una configuración de ciclo de vida al bucket `my-bucket`:

```
aws s3api put-bucket-lifecycle --bucket my-bucket --lifecycle-configuration file://lifecycle.json
```

El archivo `lifecycle.json` es un JSON documento de la carpeta actual que especifica dos reglas:

```
{
  "Rules": [
    {
      "ID": "Move to Glacier after sixty days (objects in logs/2015/)",
      "Prefix": "logs/2015/",
      "Status": "Enabled",
      "Transition": {
        "Days": 60,
        "StorageClass": "GLACIER"
      }
    },
    {
      "Expiration": {
        "Date": "2016-01-01T00:00:00.000Z"
      },
      "ID": "Delete 2014 logs in 2016.",
      "Prefix": "logs/2014/",
      "Status": "Enabled"
    }
  ]
}
```

```
}
```

La primera regla mueve los archivos a Amazon Glacier después de sesenta días. La segunda regla elimina los archivos de Amazon S3 en la fecha especificada. Para obtener información sobre los formatos de marca de tiempo aceptables, consulte Especificación de valores de parámetros en la Guía del AWS CLI usuario.

Cada regla del ejemplo anterior especifica una política (`TransitionoExpiration`) y un prefijo de archivo (nombre de carpeta) a los que se aplica. También puedes crear una regla que se aplique a todo un depósito especificando un prefijo en blanco:

```
{
  "Rules": [
    {
      "ID": "Move to Glacier after sixty days (all objects in bucket)",
      "Prefix": "",
      "Status": "Enabled",
      "Transition": {
        "Days": 60,
        "StorageClass": "GLACIER"
      }
    }
  ]
}
```

- Para API obtener más información, consulte [PutBucketLifecycle](#) la Referencia de AWS CLI comandos.

## put-bucket-logging

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-logging`.

### AWS CLI

Ejemplo 1: Configuración del registro de políticas de bucket

El siguiente `put-bucket-logging` ejemplo establece la política de registro para `MyBucket`. En primer lugar, conceda al servicio de registro el permiso de entidad principal en la política de bucket mediante el comando `put-bucket-policy`.

```
aws s3api put-bucket-policy \
```

```
--bucket MyBucket \  
--policy file://policy.json
```

Contenidos de `policy.json`:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "S3ServerAccessLogsPolicy",  
      "Effect": "Allow",  
      "Principal": {"Service": "logging.s3.amazonaws.com"},  
      "Action": "s3:PutObject",  
      "Resource": "arn:aws:s3:::MyBucket/Logs/*",  
      "Condition": {  
        "ArnLike": {"aws:SourceARN": "arn:aws:s3:::SOURCE-BUCKET-NAME"},  
        "StringEquals": {"aws:SourceAccount": "SOURCE-AWS-ACCOUNT-ID"}  
      }  
    }  
  ]  
}
```

Para aplicar la política de registro, use `put-bucket-logging`.

```
aws s3api put-bucket-logging \  
--bucket MyBucket \  
--bucket-logging-status file://logging.json
```

Contenidos de `logging.json`:

```
{  
  "LoggingEnabled": {  
    "TargetBucket": "MyBucket",  
    "TargetPrefix": "Logs/"  
  }  
}
```

El comando `put-bucket-policy` es necesario para conceder permisos `s3:PutObject` a la entidad principal del servicio de registro.

Para obtener más información, consulte [Registro de acceso al servidor de Amazon S3](#) en la Guía del usuario de Amazon S3.

## Ejemplo 2: Establecimiento de una política de bucket para registrar el acceso a un solo usuario

El siguiente `put-bucket-logging` ejemplo establece la política de registro para `MyBucket`. El AWS usuario `bob@example.com` tendrá el control total sobre los archivos de registro y nadie más tendrá acceso a ellos. En primer lugar, conceda permiso de S3 mediante `put-bucket-acl`.

```
aws s3api put-bucket-acl \  
  --bucket MyBucket \  
  --grant-write URI=http://acs.amazonaws.com/groups/s3/LogDelivery \  
  --grant-read-acp URI=http://acs.amazonaws.com/groups/s3/LogDelivery
```

A continuación, aplique la política de registro mediante `put-bucket-logging`.

```
aws s3api put-bucket-logging \  
  --bucket MyBucket \  
  --bucket-logging-status file://logging.json
```

Contenidos de `logging.json`:

```
{  
  "LoggingEnabled": {  
    "TargetBucket": "MyBucket",  
    "TargetPrefix": "MyBucketLogs/",  
    "TargetGrants": [  
      {  
        "Grantee": {  
          "Type": "AmazonCustomerByEmail",  
          "EmailAddress": "bob@example.com"  
        },  
        "Permission": "FULL_CONTROL"  
      }  
    ]  
  }  
}
```

el comando `put-bucket-acl` es necesario para conceder los permisos necesarios (`write` y `read-acp`) al sistema de entrega de registros de S3.

Para obtener más información, consulte [Registro de acceso al servidor de Amazon S3](#) en la Guía para desarrolladores de Amazon S3.



- Para API obtener más información, consulte [PutBucketLogging](#) la Referencia de AWS CLI comandos.

## put-bucket-metrics-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-metrics-configuration`.

### AWS CLI

Para establecer una configuración de métricas para un bucket

El siguiente `put-bucket-metrics-configuration` ejemplo establece una configuración métrica con el ID 123 para el depósito especificado.

```
aws s3api put-bucket-metrics-configuration \  
  --bucket my-bucket \  
  --id 123 \  
  --metrics-configuration '{"Id": "123", "Filter": {"Prefix": "logs"}}'
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutBucketMetricsConfiguration](#) la Referencia de AWS CLI comandos.

## put-bucket-notification-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-notification-configuration`.

### AWS CLI

Habilitación de las notificaciones especificadas en un bucket

El siguiente ejemplo de `put-bucket-notification-configuration` se aplica una configuración de notificación a un bucket llamado `my-bucket`. El archivo `notification.json` es un JSON documento de la carpeta actual que especifica el SNS tema y el tipo de evento que se va a supervisar.

```
aws s3api put-bucket-notification-configuration \  
  --bucket my-bucket \  
  --notification-configuration notification.json
```

```
--notification-configuration file://notification.json
```

Contenidos de notification.json:

```
{
  "TopicConfigurations": [
    {
      "TopicArn": "arn:aws:sns:us-west-2:123456789012:s3-notification-topic",
      "Events": [
        "s3:ObjectCreated:*"
      ]
    }
  ]
}
```

El SNS tema debe tener una IAM política adjunta que permita a Amazon S3 publicar en él.

```
{
  "Version": "2008-10-17",
  "Id": "example-ID",
  "Statement": [
    {
      "Sid": "example-statement-ID",
      "Effect": "Allow",
      "Principal": {
        "Service": "s3.amazonaws.com"
      },
      "Action": [
        "SNS:Publish"
      ],
      "Resource": "arn:aws:sns:us-west-2:123456789012::s3-notification-topic",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:s3:*:*:my-bucket"
        }
      }
    }
  ]
}
```

- Para API obtener más información, consulte [PutBucketNotificationConfiguration](#) la Referencia de AWS CLI comandos.

## put-bucket-notification

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-notification`.

### AWS CLI

Aplica una configuración de notificación a un bucket denominado `my-bucket`:

```
aws s3api put-bucket-notification --bucket my-bucket --notification-configuration file://notification.json
```

El archivo `notification.json` es un JSON documento de la carpeta actual que especifica un SNS tema y un tipo de evento que se va a supervisar:

```
{
  "TopicConfiguration": {
    "Event": "s3:ObjectCreated:*",
    "Topic": "arn:aws:sns:us-west-2:123456789012:s3-notification-topic"
  }
}
```

El SNS tema debe tener una IAM política adjunta que permita a Amazon S3 publicar en él:

```
{
  "Version": "2008-10-17",
  "Id": "example-ID",
  "Statement": [
    {
      "Sid": "example-statement-ID",
      "Effect": "Allow",
      "Principal": {
        "Service": "s3.amazonaws.com"
      },
      "Action": [
        "SNS:Publish"
      ],
      "Resource": "arn:aws:sns:us-west-2:123456789012:my-bucket",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:s3:*:*:my-bucket"
        }
      }
    }
  ]
}
```

```
]
}
```

- Para API obtener más información, consulte [PutBucketNotification](#) la Referencia de AWS CLI comandos.

## put-bucket-ownership-controls

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-ownership-controls`.

### AWS CLI

Para actualizar la configuración de propiedad de un bucket

En el siguiente `put-bucket-ownership-controls` ejemplo, se actualiza la configuración de propiedad de un bucket.

```
aws s3api put-bucket-ownership-controls \  
  --bucket DOC-EXAMPLE-BUCKET \  
  --ownership-controls="Rules=[{ObjectOwnership=BucketOwnerEnforced}]"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Establecer la propiedad de un objeto en un bucket existente](#) en la Guía del usuario de Amazon S3.

- Para API obtener más información, consulte [PutBucketOwnershipControls](#) la Referencia de AWS CLI comandos.

## put-bucket-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-policy`.

### AWS CLI

Este ejemplo permite a todos los usuarios recuperar cualquier objeto `MyBucket` excepto los del `MySecretFolder`. También concede `put` un `delete` permiso al usuario raíz de la AWS cuenta `1234-5678-9012`:

```
aws s3api put-bucket-policy --bucket MyBucket --policy file://policy.json
```

```

policy.json:
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::MyBucket/*"
    },
    {
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::MyBucket/MySecretFolder/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:root"
      },
      "Action": [
        "s3:DeleteObject",
        "s3:PutObject"
      ],
      "Resource": "arn:aws:s3:::MyBucket/*"
    }
  ]
}

```

- Para API obtener más información, consulte [PutBucketPolicy](#) la Referencia de AWS CLI comandos.

## put-bucket-replication

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-replication`.

### AWS CLI

Configurar la replicación de un bucket de S3

El siguiente ejemplo de `put-bucket-replication` aplica una configuración de replicación al bucket de S3 especificado.

```
aws s3api put-bucket-replication \  
  --bucket AWSDOC-EXAMPLE-BUCKET1 \  
  --replication-configuration file://replication.json
```

Contenidos de replication.json:

```
{  
  "Role": "arn:aws:iam::123456789012:role/s3-replication-role",  
  "Rules": [  
    {  
      "Status": "Enabled",  
      "Priority": 1,  
      "DeleteMarkerReplication": { "Status": "Disabled" },  
      "Filter" : { "Prefix": ""},  
      "Destination": {  
        "Bucket": "arn:aws:s3:::AWSDOC-EXAMPLE-BUCKET2"  
      }  
    }  
  ]  
}
```

El bucket de destino debe tener habilitado el control de versiones. El rol especificado debe tener permiso para escribir en el bucket de destino y tener una relación de confianza que permita a Amazon S3 asumir el rol.

Ejemplo de política de permisos de roles:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "s3:GetReplicationConfiguration",  
        "s3:ListBucket"  
      ],  
      "Resource": [  
        "arn:aws:s3:::AWSDOC-EXAMPLE-BUCKET1"  
      ]  
    },  
    {  
      "Effect": "Allow",
```

```

    "Action": [
      "s3:GetObjectVersion",
      "s3:GetObjectVersionAcl",
      "s3:GetObjectVersionTagging"
    ],
    "Resource": [
      "arn:aws:s3:::AWSDOC-EXAMPLE-BUCKET1/*"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "s3:ReplicateObject",
      "s3:ReplicateDelete",
      "s3:ReplicateTags"
    ],
    "Resource": "arn:aws:s3:::AWSDOC-EXAMPLE-BUCKET2/*"
  }
]
}

```

Ejemplo de política de relación de confianza:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "s3.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Este es el título del tema](#) en la guía del usuario de la consola de Amazon Simple Storage Service:

- Para API obtener más información, consulte [PutBucketReplication](#) la Referencia de AWS CLI comandos.

## put-bucket-request-payment

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-request-payment`.

### AWS CLI

Ejemplo 1: habilitar la configuración de “el solicitante paga” para un bucket

El siguiente ejemplo de `put-bucket-request-payment` habilita `requester pays` para el bucket especificado.

```
aws s3api put-bucket-request-payment \  
  --bucket my-bucket \  
  --request-payment-configuration '{"Payer":"Requester"}'
```

Este comando no genera ninguna salida.

Ejemplo 2: deshabilitar la configuración de “el solicitante paga” para un bucket

El siguiente ejemplo de `put-bucket-request-payment` deshabilita `requester pays` para el bucket especificado.

```
aws s3api put-bucket-request-payment \  
  --bucket my-bucket \  
  --request-payment-configuration '{"Payer":"BucketOwner"}'
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutBucketRequestPayment](#) la Referencia de AWS CLI comandos.

## put-bucket-tagging

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-tagging`.

### AWS CLI

El siguiente comando aplica una configuración de etiquetado a un bucket denominado `my-bucket`:

```
aws s3api put-bucket-tagging --bucket my-bucket --tagging file://tagging.json
```



El archivo `tagging.json` es un JSON documento de la carpeta actual que especifica las etiquetas:

```
{
  "TagSet": [
    {
      "Key": "organization",
      "Value": "marketing"
    }
  ]
}
```

O aplique una configuración de etiquetado a `my-bucket` directamente desde la línea de comandos:

```
aws s3api put-bucket-tagging --bucket my-bucket --tagging
'TagSet=[{Key=organization,Value=marketing}]'
```

- Para API obtener más información, consulte [PutBucketTagging](#) la Referencia de AWS CLI comandos.

## put-bucket-versioning

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-versioning`.

### AWS CLI

El siguiente comando habilita el control de versiones en un bucket denominado `my-bucket`:

```
aws s3api put-bucket-versioning --bucket my-bucket --versioning-
configuration Status=Enabled
```

El siguiente comando habilita el control de versiones y usa un código mfa

```
aws s3api put-bucket-versioning --bucket my-bucket --versioning-
configuration Status=Enabled --mfa "SERIAL 123456"
```

- Para API obtener más información, consulte [PutBucketVersioning](#) la Referencia de AWS CLI comandos.

## put-bucket-website

En el siguiente ejemplo de código, se muestra cómo usar `put-bucket-website`.

### AWS CLI

Aplica una configuración de sitio web estática a un bucket llamado `my-bucket`:

```
aws s3api put-bucket-website --bucket my-bucket --website-configuration file://website.json
```

El archivo `website.json` es un JSON documento de la carpeta actual que especifica las páginas de índice y error del sitio web:

```
{
  "IndexDocument": {
    "Suffix": "index.html"
  },
  "ErrorDocument": {
    "Key": "error.html"
  }
}
```

- Para API obtener más información, consulte [PutBucketWebsite](#) la Referencia de AWS CLI comandos.

## put-object-acl

En el siguiente ejemplo de código, se muestra cómo usar `put-object-acl`.

### AWS CLI

El siguiente comando concede `read` permisos `full control` a dos AWS usuarios (`user1@example.com` y `user2@example.com`) y a todos:

```
aws s3api put-object-acl --bucket MyBucket --key file.txt --grant-full-control emailaddress=user1@example.com,emailaddress=user2@example.com --grant-read uri=http://acs.amazonaws.com/groups/global/AllUsers
```

Consulte <http://docs.aws.amazon.com/AmazonS3/latest/API/RESTBucketPUTacl.html> para obtener más información sobre la personalización ACLs (los ACL comandos s3api, por ejemplo `put-object-acl`, utilizan la misma notación abreviada de argumentos).

- Para obtener API más información, consulte la Referencia de comandos. [PutObjectAcl](#) AWS CLI

## **put-object-legal-hold**

En el siguiente ejemplo de código, se muestra cómo usar `put-object-legal-hold`.

### AWS CLI

Para aplicar una retención legal a un objeto

En el siguiente ejemplo de `put-object-legal-hold`, se establece una retención legal sobre el objeto `doc1.rtf`.

```
aws s3api put-object-legal-hold \  
  --bucket my-bucket-with-object-lock \  
  --key doc1.rtf \  
  --legal-hold Status=ON
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutObjectLegalHold](#) la Referencia de AWS CLI comandos.

## **put-object-lock-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `put-object-lock-configuration`.

### AWS CLI

Para establecer la configuración de bloqueo de objetos en un bucket

En el siguiente ejemplo de `put-object-lock-configuration`, se establece un bloqueo de objetos de 50 días en el bucket especificado.

```
aws s3api put-object-lock-configuration \  
  --bucket my-bucket-with-object-lock \  
  --lock-mode COMPLIANCE \  
  --lock-duration-days 50
```

```
--object-lock-configuration '{ "ObjectLockEnabled": "Enabled", "Rule":  
{ "DefaultRetention": { "Mode": "COMPLIANCE", "Days": 50 } } }'
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutObjectLockConfiguration](#) la Referencia de AWS CLI comandos.

## put-object-retention

En el siguiente ejemplo de código, se muestra cómo usar `put-object-retention`.

### AWS CLI

Para establecer la configuración de retención de un objeto

En el siguiente ejemplo de `put-object-retention`, se establece una configuración de retención del objeto especificado hasta el 1 de enero de 2025.

```
aws s3api put-object-retention \  
  --bucket my-bucket-with-object-lock \  
  --key doc1.rtf \  
  --retention '{ "Mode": "GOVERNANCE", "RetainUntilDate": "2025-01-01T00:00:00" }'
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutObjectRetention](#) la Referencia de AWS CLI comandos.

## put-object-tagging

En el siguiente ejemplo de código, se muestra cómo usar `put-object-tagging`.

### AWS CLI

Para establecer una etiqueta en un objeto

El siguiente `put-object-tagging` ejemplo establece una etiqueta con la clave `designation` y `confidential` el valor del objeto especificado.

```
aws s3api put-object-tagging \  
  --bucket my-bucket-with-object-lock \  
  --key doc1.rtf \  
  --tagging '{ "TagSet": [ { "Key": "designation", "Value": "confidential" } ] }'
```

```
--bucket my-bucket \  
--key doc1.rtf \  
--tagging '{"TagSet": [{ "Key": "designation", "Value": "confidential" } ]}'
```

Este comando no genera ninguna salida.

El siguiente `put-object-tagging` ejemplo establece varios conjuntos de etiquetas en el objeto especificado.

```
aws s3api put-object-tagging \  
  --bucket my-bucket-example \  
  --key doc3.rtf \  
  --tagging '{"TagSet": [{ "Key": "designation", "Value": "confidential" },  
    { "Key": "department", "Value": "finance" }, { "Key": "team", "Value":  
    "payroll" } ]}'
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutObjectTagging](#) la Referencia de AWS CLI comandos.

## put-object

En el siguiente ejemplo de código, se muestra cómo usar `put-object`.

### AWS CLI

En el siguiente ejemplo se utiliza el comando `put-object` para cargar un objeto a Amazon S3:

```
aws s3api put-object --bucket text-content --key dir-1/my_images.tar.bz2 --  
body my_images.tar.bz2
```

En el siguiente ejemplo se muestra la carga de un archivo de vídeo (el archivo de vídeo se especifica mediante la sintaxis del sistema de archivos de Windows):

```
aws s3api put-object --bucket text-content --key dir-1/big-video-file.mp4 --body e:  
\media\videos\f-sharp-3-data-services.mp4
```

Para obtener más información acerca de la carga de objetos, consulte [Carga de objetos](#) en la Guía para desarrolladores de Amazon S3.

- Para API obtener más información, consulte [PutObject](#) la Referencia de AWS CLI comandos.

## put-public-access-block

En el siguiente ejemplo de código, se muestra cómo usar `put-public-access-block`.

### AWS CLI

Para establecer la configuración de acceso público en bloque para un bucket

En el siguiente `put-public-access-block` ejemplo, se establece una configuración restrictiva de acceso público en bloque para el depósito especificado.

```
aws s3api put-public-access-block \
  --bucket my-bucket \
  --public-access-block-
configuration "BlockPublicAcls=true,IgnorePublicAcls=true,BlockPublicPolicy=true,RestrictPub
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutPublicAccessBlock](#) la Referencia de AWS CLI comandos.

## rb

En el siguiente ejemplo de código, se muestra cómo usar `rb`.

### AWS CLI

Ejemplo 1: Eliminar un depósito

El siguiente `rb` comando elimina un depósito. En este ejemplo, el depósito del usuario `esmybucket`. Ten en cuenta que el depósito debe estar vacío para poder eliminar:

```
aws s3 rb s3://mybucket
```

Salida:

```
remove_bucket: mybucket
```

Ejemplo 2: Forzar la eliminación de un depósito

El siguiente `rb` comando utiliza el `--force` parámetro para eliminar primero todos los objetos del depósito y, a continuación, eliminar el propio depósito. En este ejemplo, el depósito del usuario es `mybucket` y los objetos que contiene `mybucket` son `test1.txt` y `test2.txt`:

```
aws s3 rb s3://mybucket \  
  --force
```

Salida:

```
delete: s3://mybucket/test1.txt  
delete: s3://mybucket/test2.txt  
remove_bucket: mybucket
```

- Para API obtener más información, consulte [Rb](#) en la referencia de AWS CLI comandos.

## restore-object

En el siguiente ejemplo de código, se muestra cómo usar `restore-object`.

### AWS CLI

Creación de una solicitud de restauración para un objeto

En el siguiente ejemplo de `restore-object` se restaura el objeto de Amazon S3 Glacier especificado para el bucket `my-glacier-bucket` durante 10 días.

```
aws s3api restore-object \  
  --bucket my-glacier-bucket \  
  --key doc1.rtf \  
  --restore-request Days=10
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [RestoreObject](#) la Referencia de AWS CLI comandos.

## rm

En el siguiente ejemplo de código, se muestra cómo usar `rm`.

## AWS CLI

### Ejemplo 1: Eliminar un objeto S3

El siguiente `rm` comando elimina un único objeto s3:

```
aws s3 rm s3://mybucket/test2.txt
```

Salida:

```
delete: s3://mybucket/test2.txt
```

### Ejemplo 2: Eliminar todo el contenido de un bucket

El siguiente `rm` comando elimina de forma recursiva todos los objetos incluidos en un depósito y un prefijo especificados cuando se le pasa el parámetro. `--recursive` En este ejemplo, el depósito `mybucket` contiene los objetos `test1.txt` `test2.txt`

```
aws s3 rm s3://mybucket \  
--recursive
```

Salida:

```
delete: s3://mybucket/test1.txt  
delete: s3://mybucket/test2.txt
```

### Ejemplo 3: Eliminar todo el contenido de un depósito, excepto los archivos ``.jpg``

El siguiente `rm` comando elimina de forma recursiva todos los objetos incluidos en un depósito y un prefijo específicos cuando se le pasa el parámetro, mientras que excluye algunos objetos mediante un parámetro `--recursive`. `--exclude` En este ejemplo, el depósito `mybucket` contiene los objetos `test1.txt` `test2.jpg`

```
aws s3 rm s3://mybucket/ \  
--recursive \  
--exclude "*.jpg"
```

Salida:



```
delete: s3://mybucket/test1.txt
```

Ejemplo 4: Eliminar todo el contenido de un depósito, excepto los objetos con el prefijo especificado

El siguiente `rm` comando elimina de forma recursiva todos los objetos incluidos en un depósito y un prefijo específicos cuando se les pasa el parámetro `y`, al `--recursive` mismo tiempo, excluye todos los objetos que estén bajo un prefijo concreto mediante un parámetro. `--exclude` En este ejemplo, el depósito contiene los objetos `mybucket y: test1.txt another/test.txt`

```
aws s3 rm s3://mybucket/ \  
--recursive \  
--exclude "another/*"
```

Salida:

```
delete: s3://mybucket/test1.txt
```

Ejemplo 5: Eliminar un objeto de un punto de acceso S3

El siguiente `rm` comando elimina un único objeto (`mykey`) del punto de acceso (`myaccesspoint`). :: El siguiente `rm` comando elimina un único objeto (`mykey`) del punto de acceso (`myaccesspoint`).

```
aws s3 rm s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/mykey
```

Salida:

```
delete: s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/mykey
```

- Para API obtener más información, consulte [Rm](#) en la referencia de AWS CLI comandos.

## **select-object-content**

En el siguiente ejemplo de código, se muestra cómo usar `select-object-content`.

### AWS CLI

Para filtrar el contenido de un objeto de Amazon S3 en función de una SQL declaración

El siguiente `select-object-content` ejemplo filtra el objeto `my-data-file.csv` con la SQL sentencia especificada y envía el resultado a un archivo.

```
aws s3api select-object-content \  
  --bucket my-bucket \  
  --key my-data-file.csv \  
  --expression "select * from s3object limit 100" \  
  --expression-type 'SQL' \  
  --input-serialization '{"CSV": {}, "CompressionType": "NONE"}' \  
  --output-serialization '{"CSV": {}}' "output.csv"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SelectObjectContent](#) la Referencia de AWS CLI comandos.

## sync

En el siguiente ejemplo de código, se muestra cómo usar `sync`.

### AWS CLI

Ejemplo 1: Sincronizar todos los objetos locales con el bucket especificado

El siguiente `sync` comando sincroniza los objetos de un directorio local con el prefijo y el bucket especificados cargando los archivos locales en S3. Será necesario cargar un archivo local si el tamaño del archivo local es diferente al tamaño del objeto de S3, si la hora de la última modificación del archivo local es posterior a la hora de la última modificación del objeto de S3 o si el archivo local no existe en el bucket y el prefijo especificados. En este ejemplo, el usuario sincroniza el depósito `mybucket` con el directorio actual local. El directorio actual local contiene los archivos `test.txt` y `test2.txt`. El depósito `mybucket` contiene ningún objeto.

```
aws s3 sync . s3://mybucket
```

Salida:

```
upload: test.txt to s3://mybucket/test.txt  
upload: test2.txt to s3://mybucket/test2.txt
```

Ejemplo 2: Sincronice todos los objetos de S3 del depósito de S3 especificado con otro depósito

El siguiente sync comando sincroniza los objetos con un prefijo y un bucket específicos con los objetos con otro prefijo y bucket especificados copiando los objetos de S3. Será necesario copiar un objeto de S3 si los tamaños de los dos objetos de S3 son diferentes, si la hora de la última modificación del origen es más reciente que la hora de la última modificación del destino o si el objeto de S3 no existe en el depósito y el prefijo de destino especificados.

En este ejemplo, el usuario sincroniza el depósito mybucket con el depósito. mybucket2 El depósito mybucket contiene los objetos test.txt y test2.txt El depósito no mybucket2 contiene ningún objeto:

```
aws s3 sync s3://mybucket s3://mybucket2
```

Salida:

```
copy: s3://mybucket/test.txt to s3://mybucket2/test.txt
copy: s3://mybucket/test2.txt to s3://mybucket2/test2.txt
```

Ejemplo 3: Sincronice todos los objetos de S3 del depósito de S3 especificado con el directorio local

El siguiente sync comando sincroniza los archivos del bucket de S3 especificado con el directorio local mediante la descarga de los objetos de S3. Será necesario descargar un objeto de S3 si su tamaño es diferente del tamaño del archivo local, si la hora de la última modificación del objeto de S3 es posterior a la hora de la última modificación del archivo local o si el objeto de S3 no existe en el directorio local. Tenga en cuenta que cuando los objetos se descargan de S3, la hora de la última modificación del archivo local se cambia por la hora de la última modificación del objeto de S3. En este ejemplo, el usuario sincroniza el depósito mybucket con el directorio local actual. El depósito mybucket contiene los objetos test.txt y test2.txt El directorio local actual no tiene archivos:

```
aws s3 sync s3://mybucket .
```

Salida:

```
download: s3://mybucket/test.txt to test.txt
download: s3://mybucket/test2.txt to test2.txt
```

Ejemplo 4: Sincronice todos los objetos locales con el depósito especificado y elimine todos los archivos que no coincidan

El siguiente sync comando sincroniza los objetos con un prefijo y un bucket específicos con los archivos de un directorio local cargando los archivos locales en S3. Gracias a este `--delete` parámetro, se eliminarán todos los archivos que existan con el prefijo y el bucket especificados, pero que no estén en el directorio local. En este ejemplo, el usuario sincroniza el depósito mybucket con el directorio actual local. El directorio actual local contiene los archivos `test.txt` y `test2.txt`. El depósito mybucket contiene el objeto `test3.txt`:

```
aws s3 sync . s3://mybucket \  
--delete
```

Salida:

```
upload: test.txt to s3://mybucket/test.txt  
upload: test2.txt to s3://mybucket/test2.txt  
delete: s3://mybucket/test3.txt
```

Ejemplo 5: Sincroniza todos los objetos locales con el depósito especificado excepto los archivos `*.jpg`

El siguiente sync comando sincroniza los objetos con un prefijo y un bucket específicos con los archivos de un directorio local cargando los archivos locales en S3. Gracias a este `--exclude` parámetro, se excluirán de la sincronización todos los archivos que coincidan con el patrón existente tanto en S3 como a nivel local. En este ejemplo, el usuario sincroniza el depósito mybucket con el directorio actual local. El directorio actual local contiene los archivos `test.jpg` y `test2.txt`. El depósito mybucket contiene el objeto `test.jpg` de un tamaño diferente al `localtest.jpg`:

```
aws s3 sync . s3://mybucket \  
--exclude "*.jpg"
```

Salida:

```
upload: test2.txt to s3://mybucket/test2.txt
```

Ejemplo 6: Sincroniza todos los objetos locales con el depósito especificado excepto los archivos `*.jpg`

El siguiente sync comando sincroniza los archivos de un directorio local con los objetos que tengan un prefijo y un bucket específicos mediante la descarga de objetos de S3. En este

ejemplo, se utiliza el `--exclude` parámetro flag para excluir un directorio y un prefijo S3 específicos del comando. `sync` En este ejemplo, el usuario sincroniza el directorio actual local con el bucket. `mybucket` El directorio actual local contiene los archivos `test.txt` y `another/test2.txt`. El depósito `mybucket` contiene los objetos `another/test5.txt` y `test1.txt`:

```
aws s3 sync s3://mybucket/ . \
  --exclude "*another/*"
```

Salida:

```
download: s3://mybucket/test1.txt to test1.txt
```

Ejemplo 7: Sincronizar todos los objetos entre depósitos de diferentes regiones

El siguiente `sync` comando sincroniza los archivos entre dos depósitos en diferentes regiones:

```
aws s3 sync s3://my-us-west-2-bucket s3://my-us-east-1-bucket \
  --source-region us-west-2 \
  --region us-east-1
```

Salida:

```
download: s3://my-us-west-2-bucket/test1.txt to s3://my-us-east-1-bucket/test1.txt
```

Ejemplo 8: Sincronización con un punto de acceso S3

El siguiente `sync` comando sincroniza el directorio actual con el punto de acceso (`myaccesspoint`):

```
aws s3 sync . s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/
```

Salida:

```
upload: test.txt to s3://arn:aws:s3:us-west-2:123456789012:accesspoint/
myaccesspoint/test.txt
upload: test2.txt to s3://arn:aws:s3:us-west-2:123456789012:accesspoint/
myaccesspoint/test2.txt
```

- Para API obtener más información, consulte [Sync](#) in AWS CLI Command Reference.

## upload-part-copy

En el siguiente ejemplo de código, se muestra cómo usar `upload-part-copy`.

### AWS CLI

Para cargar parte de un objeto copiando datos de un objeto existente como fuente de datos

El siguiente `upload-part-copy` ejemplo carga una parte copiando datos de un objeto existente como fuente de datos.

```
aws s3api upload-part-copy \  
  --bucket my-bucket \  
  --key "Map_Data_June.mp4" \  
  --copy-source "my-bucket/copy_of_Map_Data_June.mp4" \  
  --part-number 1 \  
  --upload-  
id "bq0tdE1CDpWQYRPLHuNG50xAT6pA5D.m_RiBy0gg0H6b13pVRY7QjvLlf75iFdJqp_2wztk5hvpUM2SesXgrzbeh"
```

Salida:

```
{  
  "CopyPartResult": {  
    "LastModified": "2019-12-13T23:16:03.000Z",  
    "ETag": "\"711470fc377698c393d94aed6305e245\""  
  }  
}
```

- Para API obtener más información, consulte [UploadPartCopy](#) la Referencia de AWS CLI comandos.

## upload-part

En el siguiente ejemplo de código, se muestra cómo usar `upload-part`.

### AWS CLI

El siguiente comando carga la primera parte de una carga multiparte iniciada con el comando `create-multipart-upload`:

```
aws s3api upload-part --bucket my-bucket --key 'multipart/01' --part-number 1 --
body part01 --upload-id
"dfRtDYU0WCCcH43C3WfbkR0NycyCpTJJvxu2i5GYkZljF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3UR"
```

La opción `body` toma el nombre o la ruta de un archivo local para la carga (no utilice el prefijo `file://`). El tamaño mínimo de parte es de 5 MB. El ID de carga lo devuelve `create-multipart-upload` y también se puede recuperar con `list-multipart-uploads`. El bucket y la clave se especifican al crear la carga multiparte.

Salida:

```
{
  "ETag": "\"e868e0f4719e394144ef36531ee6824c\""
}
```

Guarde el ETag valor de cada parte para más adelante. Son necesarios para completar la carga multiparte.

- Para API obtener más información, consulte [UploadPart](#) la Referencia de AWS CLI comandos.

## website

En el siguiente ejemplo de código, se muestra cómo usar `website`.

### AWS CLI

Configure un bucket de S3 como un sitio web estático

El siguiente comando configura un bucket `my-bucket` denominado sitio web estático. La opción del documento de índice especifica el archivo al `my-bucket` que se dirigirá a los visitantes cuando naveguen por el sitio web URL. En este caso, el depósito se encuentra en la región `us-west-2`, por lo que el sitio aparecería en `http://my-bucket.s3-website-us-west-2.amazonaws.com`

Todos los archivos del depósito que aparecen en el sitio estático deben estar configurados para que los visitantes puedan abrirlas. Los permisos de los archivos se configuran de forma independiente de la configuración del sitio web del bucket.

```
aws s3 website s3://my-bucket/ \
  --index-document index.html \
```

```
--error-document error.html
```

Para obtener información sobre el alojamiento de un sitio web estático en Amazon S3, consulte [Hospedaje de un sitio web estático](#) en la Guía para desarrolladores de Amazon Simple Storage Service.

- Para API obtener más información, consulte [Website](#) in AWS CLI Command Reference.

## Ejemplos de Amazon S3 Control que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante el uso AWS Command Line Interface de Amazon S3 Control.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-access-point**

En el siguiente ejemplo de código, se muestra cómo usar `create-access-point`.

AWS CLI

Para crear un punto de acceso

En el siguiente `create-access-point` ejemplo, se crea un punto de acceso con el nombre `finance-ap` del depósito `business-records` de la cuenta `123456789012`. Antes de ejecutar este ejemplo, sustituya el nombre del punto de acceso, el nombre del depósito y el número de cuenta por los valores adecuados para su caso de uso.

```
aws s3control create-access-point \
```



```
--account-id 123456789012 \  
--bucket business-records \  
--name finance-ap
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Creación de puntos de acceso](#) en la Guía para desarrolladores de Amazon Simple Storage Service.

- Para API obtener más información, consulte [CreateAccessPoint](#) la Referencia de AWS CLI comandos.

## create-job

En el siguiente ejemplo de código, se muestra cómo usar `create-job`.

### AWS CLI

Para crear un trabajo de operaciones por lotes en Amazon S3

El siguiente `create-job` ejemplo crea un trabajo de operaciones por lotes de Amazon S3 para etiquetar objetos como `confidential` in the bucket `employee-records`.

```
aws s3control create-job \  
  --account-id 123456789012 \  
  --operation '{"S3PutObjectTagging": { "TagSet": [{"Key":"confidential",  
  "Value":"true"}] }}' \  
  --report '{"Bucket":"arn:aws:s3:::employee-records-logs","Prefix":"batch-op-  
create-job",  
  "Format":"Report_CSV_20180820","Enabled":true,"ReportScope":"AllTasks"}' \  
  --manifest '{"Spec":{"Format":"S3BatchOperations_CSV_20180820","Fields":  
  [{"Bucket","Key"}],"Location":{"ObjectArn":"arn:aws:s3:::employee-records-logs/inv-  
report/7a6a9be4-072c-407e-85a2-  
ec3e982f773e.csv","ETag":"69f52a4e9f797e987155d9c8f5880897"}}' \  
  --priority 42 \  
  --role-arn arn:aws:iam::123456789012:role/S3BatchJobRole
```

Salida:

```
{  
  "JobId": "93735294-df46-44d5-8638-6356f335324e"
```

```
}
```

- Para API obtener más información, consulte [CreateJob](#) la Referencia de AWS CLI comandos.

## **delete-access-point-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-access-point-policy`.

### AWS CLI

Para eliminar una política de puntos de acceso

En el siguiente `delete-access-point-policy` ejemplo, se elimina la política de puntos de acceso del punto de acceso nombrado `finance-ap` en la cuenta `123456789012`. Antes de ejecutar este ejemplo, sustituya el nombre y el número de cuenta del punto de acceso por los valores adecuados para su caso de uso.

```
aws s3control delete-access-point-policy \  
  --account-id 123456789012 \  
  --name finance-ap
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión del acceso a los datos con puntos de acceso de Amazon S3](#) en la Guía para desarrolladores de Amazon Simple Storage Service.

- Para API obtener más información, consulte [DeleteAccessPointPolicy](#) la Referencia de AWS CLI comandos.

## **delete-access-point**

En el siguiente ejemplo de código, se muestra cómo usar `delete-access-point`.

### AWS CLI

Para eliminar un punto de acceso

En el siguiente `delete-access-point` ejemplo, se elimina un punto de acceso denominado `finance-ap` en la cuenta `123456789012`. Antes de ejecutar este ejemplo, sustituya el nombre y el número de cuenta del punto de acceso por los valores adecuados para su caso de uso.

```
aws s3control delete-access-point \  
  --account-id 123456789012 \  
  --name finance-ap
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión del acceso a los datos con puntos de acceso de Amazon S3](#) en la Guía para desarrolladores de Amazon Simple Storage Service.

- Para API obtener más información, consulte [DeleteAccessPoint](#) la Referencia de AWS CLI comandos.

## **delete-public-access-block**

En el siguiente ejemplo de código, se muestra cómo usar `delete-public-access-block`.

### AWS CLI

Para eliminar la configuración de bloqueo de acceso público de una cuenta

En el siguiente `delete-public-access-block` ejemplo, se elimina la configuración de bloqueo de acceso público de la cuenta especificada.

```
aws s3control delete-public-access-block \  
  --account-id 123456789012
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeletePublicAccessBlock](#) la Referencia de AWS CLI comandos.

## **describe-job**

En el siguiente ejemplo de código, se muestra cómo usar `describe-job`.

### AWS CLI

Para describir un trabajo de operaciones por lotes de Amazon S3

A continuación, se `describe-job` proporcionan los parámetros de configuración y el estado del trabajo de operaciones por lotes especificado.

```
aws s3control describe-job \  
--account-id 123456789012 \  
--job-id 93735294-df46-44d5-8638-6356f335324e
```

Salida:

```
{  
  "Job": {  
    "TerminationDate": "2019-10-03T21:49:53.944Z",  
    "JobId": "93735294-df46-44d5-8638-6356f335324e",  
    "FailureReasons": [],  
    "Manifest": {  
      "Spec": {  
        "Fields": [  
          "Bucket",  
          "Key"  
        ],  
        "Format": "S3BatchOperations_CSV_20180820"  
      },  
      "Location": {  
        "ETag": "69f52a4e9f797e987155d9c8f5880897",  
        "ObjectArn": "arn:aws:s3:::employee-records-logs/inv-report/7a6a9be4-072c-407e-85a2-ec3e982f773e.csv"  
      }  
    },  
    "Operation": {  
      "S3PutObjectTagging": {  
        "TagSet": [  
          {  
            "Value": "true",  
            "Key": "confidential"  
          }  
        ]  
      }  
    },  
    "RoleArn": "arn:aws:iam::123456789012:role/S3BatchJobRole",  
    "ProgressSummary": {  
      "TotalNumberOfTasks": 8,  
      "NumberOfTasksFailed": 0,  
      "NumberOfTasksSucceeded": 8  
    },  
    "Priority": 42,  
    "Report": {
```

```

        "ReportScope": "AllTasks",
        "Format": "Report_CSV_20180820",
        "Enabled": true,
        "Prefix": "batch-op-create-job",
        "Bucket": "arn:aws:s3:::employee-records-logs"
    },
    "JobArn": "arn:aws:s3:us-west-2:123456789012:job/93735294-
df46-44d5-8638-6356f335324e",
    "CreationTime": "2019-10-03T21:48:48.048Z",
    "Status": "Complete"
}
}

```

- Para API obtener más información, consulte [DescribeJob](#) la Referencia de AWS CLI comandos.

## get-access-point-policy-status

En el siguiente ejemplo de código, se muestra cómo usar `get-access-point-policy-status`.

### AWS CLI

Para recuperar el estado de la política del punto de acceso

En el siguiente `get-access-point-policy-status` ejemplo, se recupera el estado de la política del punto de acceso del punto de acceso nombrado `finance-ap` en la cuenta `123456789012`. El estado de la política del punto de acceso indica si la política del punto de acceso permite el acceso público. Antes de ejecutar este ejemplo, sustituya el nombre y el número de cuenta del punto de acceso por los valores adecuados para su caso de uso.

```

aws s3control get-access-point-policy-status \
  --account-id 123456789012 \
  --name finance-ap

```

Salida:

```

{
  "PolicyStatus": {
    "IsPublic": false
  }
}

```

Para obtener más información sobre cuándo una política de puntos de acceso se considera pública, consulte [El significado de «público»](#) en la Guía para desarrolladores de Amazon Simple Storage Service.

- Para API obtener más información, consulte [GetAccessPointPolicyStatus](#) la Referencia de AWS CLI comandos.

## get-access-point-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-access-point-policy`.

### AWS CLI

Para recuperar una política de puntos de acceso

En el siguiente `get-access-point-policy` ejemplo, se recupera la política de puntos de acceso del punto de acceso nombrado `finance-ap` en la cuenta `123456789012`. Antes de ejecutar este ejemplo, sustituya el nombre y el número de cuenta del punto de acceso por los valores adecuados para su caso de uso.

```
aws s3control get-access-point-policy \  
  --account-id 123456789012 \  
  --name finance-ap
```

Salida:

```
{  
  "Policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow\",  
  \"Principal\":{\"AWS\":\"arn:aws:iam:123456789012:role/Admin\"},\"Action\":  
  \"s3:GetObject\",\"Resource\":\"arn:aws:s3:us-west-2:123456789012:accesspoint/  
  finance-ap/object/records/*\"}]}"  
}
```

Para obtener más información, consulte [Gestión del acceso a los datos con puntos de acceso de Amazon S3](#) en la Guía para desarrolladores de Amazon Simple Storage Service.

- Para API obtener más información, consulte [GetAccessPointPolicy](#) la Referencia de AWS CLI comandos.

## get-access-point

En el siguiente ejemplo de código, se muestra cómo usar `get-access-point`.

### AWS CLI

Para recuperar los detalles de configuración del punto de acceso

En el siguiente `get-access-point` ejemplo, se recuperan los detalles de configuración del punto de acceso nombrado `finance-ap` en la cuenta `123456789012`. Antes de ejecutar este ejemplo, sustituya el nombre y el número de cuenta del punto de acceso por los valores adecuados para su caso de uso.

```
aws s3control get-access-point \  
  --account-id 123456789012 \  
  --name finance-ap
```

Salida:

```
{  
  "Name": "finance-ap",  
  "Bucket": "business-records",  
  "NetworkOrigin": "Internet",  
  "PublicAccessBlockConfiguration": {  
    "BlockPublicAcls": false,  
    "IgnorePublicAcls": false,  
    "BlockPublicPolicy": false,  
    "RestrictPublicBuckets": false  
  },  
  "CreationDate": "2020-01-01T00:00:00Z"  
}
```

Para obtener más información, consulte [Gestión del acceso a los datos con puntos de acceso de Amazon S3](#) en la Guía para desarrolladores de Amazon Simple Storage Service.

- Para API obtener más información, consulte [GetAccessPoint](#) la Referencia de AWS CLI comandos.

## get-multi-region-access-point-routes

En el siguiente ejemplo de código, se muestra cómo usar `get-multi-region-access-point-routes`.

## AWS CLI

Para consultar la configuración actual de la ruta del punto de acceso multirregional

El siguiente `get-multi-region-access-point-routes` ejemplo devuelve la configuración de enrutamiento actual del punto de acceso multirregional especificado.

```
aws s3control get-multi-region-access-point-routes \  
  --region Region \  
  --account-id 111122223333 \  
  --mrap MultiRegionAccessPoint_ARN
```

Salida:

```
{  
  "Mrap": "arn:aws:s3::111122223333:accesspoint/0000000000000000.mrap",  
  "Routes": [  
    {  
      "Bucket": "DOC-EXAMPLE-BUCKET-1",  
      "Region": "ap-southeast-2",  
      "TrafficDialPercentage": 100  
    },  
    {  
      "Bucket": "DOC-EXAMPLE-BUCKET-2",  
      "Region": "us-west-1",  
      "TrafficDialPercentage": 0  
    }  
  ]  
}
```

- Para API obtener más información, consulte [GetMultiRegionAccessPointRoutes](#) la Referencia de AWS CLI comandos.

## **get-public-access-block**

En el siguiente ejemplo de código, se muestra cómo usar `get-public-access-block`.

## AWS CLI

Para ver una lista de la configuración de acceso público por bloqueo de una cuenta



En el siguiente `get-public-access-block` ejemplo, se muestra la configuración de bloqueo del acceso público de la cuenta especificada.

```
aws s3control get-public-access-block \  
  --account-id 123456789012
```

Salida:

```
{  
  "PublicAccessBlockConfiguration": {  
    "BlockPublicPolicy": true,  
    "RestrictPublicBuckets": true,  
    "IgnorePublicAcls": true,  
    "BlockPublicAcls": true  
  }  
}
```

- Para API obtener más información, consulte [GetPublicAccessBlock](#) la Referencia de AWS CLI comandos.

## list-access-points

En el siguiente ejemplo de código, se muestra cómo usar `list-access-points`.

AWS CLI

Ejemplo 1: Para recuperar una lista de todos los puntos de acceso de una cuenta

En el siguiente `list-access-points` ejemplo, se muestra una lista de todos los puntos de acceso asociados a los depósitos propiedad de la cuenta 123456789012.

```
aws s3control list-access-points \  
  --account-id 123456789012
```

Salida:

```
{  
  "AccessPointList": [  
    {  
      "Name": "finance-ap",  
      "NetworkOrigin": "Internet",
```

```

    "Bucket": "business-records"
  },
  {
    "Name": "managers-ap",
    "NetworkOrigin": "Internet",
    "Bucket": "business-records"
  },
  {
    "Name": "private-network-ap",
    "NetworkOrigin": "VPC",
    "VpcConfiguration": {
      "VpcId": "1a2b3c"
    },
    "Bucket": "business-records"
  },
  {
    "Name": "customer-ap",
    "NetworkOrigin": "Internet",
    "Bucket": "external-docs"
  },
  {
    "Name": "public-ap",
    "NetworkOrigin": "Internet",
    "Bucket": "external-docs"
  }
]
}

```

Ejemplo 2: Para recuperar una lista de todos los puntos de acceso de un bucket

En el siguiente `list-access-points` ejemplo, se recupera una lista de todos los puntos de acceso adjuntos al depósito `external-docs` propiedad de la cuenta `123456789012`.

```

aws s3control list-access-points \
  --account-id 123456789012 \
  --bucket external-docs

```

Salida:

```

{
  "AccessPointList": [
    {
      "Name": "customer-ap",

```

```
        "NetworkOrigin": "Internet",
        "Bucket": "external-docs"
    },
    {
        "Name": "public-ap",
        "NetworkOrigin": "Internet",
        "Bucket": "external-docs"
    }
]
}
```

Para obtener más información, consulte [Gestión del acceso a los datos con puntos de acceso de Amazon S3](#) en la Guía para desarrolladores de Amazon Simple Storage Service.

- Para API obtener más información, consulte [ListAccessPoints](#) la Referencia de AWS CLI comandos.

## list-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-jobs`.

AWS CLI

Para enumerar los trabajos de operaciones por lotes de Amazon S3 de una cuenta

En el siguiente `list-jobs` ejemplo, se enumeran todos los trabajos de operaciones por lotes recientes de la cuenta especificada.

```
aws s3control list-jobs \
  --account-id 123456789012
```

Salida:

```
{
  "Jobs": [
    {
      "Operation": "S3PutObjectTagging",
      "ProgressSummary": {
        "NumberOfTasksFailed": 0,
        "NumberOfTasksSucceeded": 8,
        "TotalNumberOfTasks": 8
      }
    },
  ],
}
```

```

    "CreationTime": "2019-10-03T21:48:48.048Z",
    "Status": "Complete",
    "JobId": "93735294-df46-44d5-8638-6356f335324e",
    "Priority": 42
  },
  {
    "Operation": "S3PutObjectTagging",
    "ProgressSummary": {
      "NumberOfTasksFailed": 0,
      "NumberOfTasksSucceeded": 0,
      "TotalNumberOfTasks": 0
    },
    "CreationTime": "2019-10-03T21:46:07.084Z",
    "Status": "Failed",
    "JobId": "3f3c7619-02d3-4779-97f6-1d98dd313108",
    "Priority": 42
  },
]
}

```

- Para API obtener más información, consulte [ListJobs](#) la Referencia de AWS CLI comandos.

## put-access-point-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-access-point-policy`.

### AWS CLI

Para establecer una política de puntos de acceso

El siguiente `put-access-point-policy` ejemplo coloca la política de punto de acceso especificada para el punto de acceso `finance-ap` en la cuenta `123456789012`. Si el punto de acceso `finance-ap` ya tiene una política, este comando reemplaza la política existente por la especificada en este comando. Antes de ejecutar este ejemplo, sustituya el número de cuenta, el nombre del punto de acceso y las declaraciones de política por los valores adecuados para su caso de uso.

```

aws s3control put-access-point-policy \
  --account-id 123456789012 \
  --name finance-ap \
  --policy file://ap-policy.json

```

## Contenidos de ap-policy.json:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:user/Alice"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:us-west-2:123456789012:accesspoint/finance-ap/object/Alice/*"
    }
  ]
}
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión del acceso a los datos con puntos de acceso de Amazon S3](#) en la Guía para desarrolladores de Amazon Simple Storage Service.

- Para API obtener más información, consulte [PutAccessPointPolicy](#) la Referencia de AWS CLI comandos.

## put-public-access-block

En el siguiente ejemplo de código, se muestra cómo usar put-public-access-block.

### AWS CLI

Para editar la configuración de bloqueo de acceso público de una cuenta

En el siguiente put-public-access-block ejemplo, se cambian todas las configuraciones de bloqueo del acceso público a las true de la cuenta especificada.

```
aws s3control put-public-access-block \
  --account-id 123456789012 \
  --public-access-block-configuration '{"BlockPublicAcls": true,
  "IgnorePublicAcls": true, "BlockPublicPolicy": true, "RestrictPublicBuckets":
  true}'
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutPublicAccessBlock](#) la Referencia de AWS CLI comandos.

## **submit-multi-region-access-point-routes**

En el siguiente ejemplo de código, se muestra cómo usar `submit-multi-region-access-point-routes`.

### AWS CLI

Para actualizar la configuración de enrutamiento de su punto de acceso multirregional

El siguiente `submit-multi-region-access-point-routes` ejemplo actualiza los estados de enrutamiento de `DOC-EXAMPLE-BUCKET-1` y dentro de la `DOC-EXAMPLE-BUCKET-2` `ap-southeast-2` región para su punto de acceso multirregional.

```
aws s3control submit-multi-region-access-point-routes \  
  --region ap-southeast-2 \  
  --account-id 111122223333 \  
  --mrap MultiRegionAccessPoint_ARN \  
  --route-updates Bucket=DOC-EXAMPLE-  
BUCKET-1,TrafficDialPercentage=100 Bucket=DOC-EXAMPLE-  
BUCKET-2,TrafficDialPercentage=0
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte la Referencia [SubmitMultiRegionAccessPointRoutes](#) de AWS CLI comandos.

## **update-job-priority**

En el siguiente ejemplo de código, se muestra cómo usar `update-job-priority`.

### AWS CLI

Para actualizar la prioridad de un trabajo de operaciones por lotes de Amazon S3

En el siguiente `update-job-priority` ejemplo, se actualiza el trabajo especificado a una nueva prioridad.

```
aws s3control update-job-priority \  
  --account-id 123456789012 \  
  --job-id 8d9a18fe-c303-4d39-8ccc-860d372da386 \  
  --priority 52
```

Salida:

```
{  
  "JobId": "8d9a18fe-c303-4d39-8ccc-860d372da386",  
  "Priority": 52  
}
```

- Para API obtener más información, consulte [UpdateJobPriority](#) la Referencia de AWS CLI comandos.

## update-job-status

En el siguiente ejemplo de código, se muestra cómo usar `update-job-status`.

AWS CLI

Para actualizar el estado de un trabajo de operaciones por lotes de Amazon S3

El siguiente `update-job-status` ejemplo cancela el trabajo especificado que está pendiente de aprobación.

```
aws s3control update-job-status \  
  --account-id 123456789012 \  
  --job-id 8d9a18fe-c303-4d39-8ccc-860d372da386 \  
  --requested-job-status Cancelled
```

Salida:

```
{  
  "Status": "Cancelled",  
  "JobId": "8d9a18fe-c303-4d39-8ccc-860d372da386"  
}
```

El siguiente `update-job-status` ejemplo confirma y ejecuta el especificado que está pendiente de aprobación.

```
aws s3control update-job-status \  
  --account-id 123456789012 \  
  --job-id 5782949f-3301-4fb3-be34-8d5bab54dbca \  
  --requested-job-status Ready
```

**Output::**

```
{  
  "Status": "Ready",  
  "JobId": "5782949f-3301-4fb3-  
be34-8d5bab54dbca"  
}
```

En el `update-job-status` ejemplo siguiente se cancela el trabajo especificado que se está ejecutando.

```
aws s3control update-job-status \  
  --account-id 123456789012 \  
  --job-id 5782949f-3301-4fb3-be34-8d5bab54dbca \  
  --requested-job-status Cancelled
```

**Output::**

```
{  
  "Status": "Cancelling",  
  "JobId": "5782949f-3301-4fb3-be34-8d5bab54dbca"  
}
```

- Para API obtener más información, consulte [UpdateJobStatus](#) la Referencia de AWS CLI comandos.

## Ejemplos de S3 Glacier utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con S3 Glacier.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.



## Temas

- [Acciones](#)

## Acciones

### **abort-multipart-upload**

En el siguiente ejemplo de código, se muestra cómo usar `abort-multipart-upload`.

#### AWS CLI

El siguiente comando elimina una carga multiparte en curso a un almacén denominado: `my-vault`

```
aws glacier abort-multipart-upload --account-id - --vault-name my-vault
--upload-id 19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-0ssZtLqyFu7sY1_LR7vgFuJV6NtcV5zpsJ
```

Este comando no produce ninguna salida. Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso. El comando `aws glacier initiate-multipart-upload` devuelve el ID de carga y también se puede obtener mediante `aws glacier list-multipart-uploads`.

Para obtener más información sobre las cargas multiparte a Amazon Glacier mediante el AWS CLI, consulte [Uso de Amazon Glacier](#) en la Guía del AWS CLI usuario.

- Para API obtener más información, consulte la Referencia [AbortMultipartUpload](#) de AWS CLI comandos.

### **abort-vault-lock**

En el siguiente ejemplo de código, se muestra cómo usar `abort-vault-lock`.

#### AWS CLI

Para anular un proceso de bloqueo de bóveda en curso

En el siguiente `abort-vault-lock` ejemplo, se elimina una política de bloqueo del depósito especificado y se restablece el estado de bloqueo del depósito a desbloqueado.

```
aws glacier abort-vault-lock \
```

```
--account-id - \  
--vault-name MyVaultName
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Abortar Vault DELETE Lock \(política de bloqueo\)](#) en la Guía para desarrolladores de Amazon API Glacier.

- Para API obtener más información, consulte la Referencia de [AbortVaultLock](#) comandos AWS CLI .

## add-tags-to-vault

En el siguiente ejemplo de código, se muestra cómo usar `add-tags-to-vault`.

AWS CLI

El siguiente comando añade dos etiquetas a un almacén denominado `my-vault`:

```
aws glacier add-tags-to-vault --account-id - --vault-name my-vault --  
tags id=1234,date=july2015
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

- Para API obtener más información, consulte [AddTagsToVault](#) la Referencia de AWS CLI comandos.

## complete-multipart-upload

En el siguiente ejemplo de código, se muestra cómo usar `complete-multipart-upload`.

AWS CLI

El siguiente comando completa la carga multiparte de un archivo de 3 MiB:

```
aws glacier complete-multipart-upload --archive-size 3145728 --  
checksum 9628195fcdbcbbe76cdde456d4646fa7de5f219fb39823836d81f0cc0e18aa67  
--upload-id 19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-  
OssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ --account-id - --vault-name my-vault
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

El comando `aws glacier initiate-multipart-upload` devuelve el ID de carga y también se puede obtener mediante `aws glacier list-multipart-uploads`. El parámetro `checksum` toma un hash de árbol SHA de -256 bits del archivo en formato hexadecimal.

Para obtener más información sobre las cargas de varias partes a Amazon Glacier mediante el uso de Amazon Glacier AWS CLI, incluidas las instrucciones sobre cómo calcular un hash de árbol, consulte [Uso de Amazon Glacier en la Guía del AWS CLI usuario](#).

- Para API obtener más información, consulte la Referencia [CompleteMultipartUpload](#) de AWS CLI comandos.

## **complete-vault-lock**

En el siguiente ejemplo de código, se muestra cómo usar `complete-vault-lock`.

### AWS CLI

Para completar un proceso de bloqueo de bóveda en curso

En el siguiente `complete-vault-lock` ejemplo, se completa el proceso de bloqueo en curso del depósito especificado y se establece el estado de bloqueo del depósito en. `Locked` El valor del `lock-id` parámetro se obtiene cuando se ejecuta `initiate-lock-process`.

```
aws glacier complete-vault-lock \  
  --account-id - \  
  --vault-name MyVaultName \  
  --lock-id 9QZgEXAMPLEPhvL6xEXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Complete Vault Lock \(POSTlockId\)](#) en la Guía para API desarrolladores de Amazon Glacier.

- Para API obtener más información, consulte [CompleteVaultLock](#) la Referencia de AWS CLI comandos.

## **create-vault**

En el siguiente ejemplo de código, se muestra cómo usar `create-vault`.

## AWS CLI

El comando siguiente crea un nuevo almacén llamado `my-vault`.

```
aws glacier create-vault --vault-name my-vault --account-id -
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

- Para API obtener más información, consulte [CreateVault](#) la Referencia de AWS CLI comandos.

## `delete-archive`

En el siguiente ejemplo de código, se muestra cómo usar `delete-archive`.

## AWS CLI

Eliminación de un archivo desde un almacén

En el siguiente ejemplo de `delete-archive` se elimina el archivo especificado de `example_vault`.

```
aws glacier delete-archive \  
  --account-id 111122223333 \  
  --vault-name example_vault \  
  --archive-id Sc0u9ZP8yaWkmh-XGLIvAVprtLhaLCGnNwNL5I5x9HqPIkX5mjc0DrId3Ln-Gi_k2HzmLIDZUz117KSdVMdMXLuFWi9PJUiTxW073edQ43eTLMWkH0pd9zVSAuV_XXZBVhKhyGhJ7w
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteArchive](#) la Referencia de AWS CLI comandos.

## `delete-vault-access-policy`

En el siguiente ejemplo de código, se muestra cómo usar `delete-vault-access-policy`.

## AWS CLI

Para eliminar la política de acceso de un almacén

En el siguiente `delete-vault-access-policy` ejemplo, se elimina la política de acceso del almacén especificado.

```
aws glacier delete-vault-access-policy \  
  --account-id 111122223333 \  
  --vault-name example_vault
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteVaultAccessPolicy](#) la Referencia de AWS CLI comandos.

## **delete-vault-notifications**

En el siguiente ejemplo de código, se muestra cómo usar `delete-vault-notifications`.

### AWS CLI

Para eliminar las SNS notificaciones de un almacén

El siguiente `delete-vault-notifications` ejemplo elimina las notificaciones enviadas por Amazon Simple Notification Service (AmazonSNS) para el almacén especificado.

```
aws glacier delete-vault-notifications \  
  --account-id 111122223333 \  
  --vault-name example_vault
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteVaultNotifications](#) la Referencia de AWS CLI comandos.

## **delete-vault**

En el siguiente ejemplo de código, se muestra cómo usar `delete-vault`.

### AWS CLI

El comando siguiente elimina un almacén llamado `my-vault`.

```
aws glacier delete-vault --vault-name my-vault --account-id -
```

Este comando no produce ninguna salida. Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

- Para API obtener más información, consulte [DeleteVault](#) la Referencia de AWS CLI comandos.

## describe-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-job`.

### AWS CLI

El siguiente comando recupera información sobre un trabajo de recuperación de inventario en un almacén denominado `my-vault`:

```
aws glacier describe-job --account-id - --vault-name my-  
vault --job-id zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-  
R047Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqLNHS61ds04CnMW
```

Salida:

```
{  
  "InventoryRetrievalParameters": {  
    "Format": "JSON"  
  },  
  "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",  
  "Completed": false,  
  "JobId": "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-  
R047Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqLNHS61ds04CnMW",  
  "Action": "InventoryRetrieval",  
  "CreationDate": "2015-07-17T20:23:41.616Z",  
  "StatusCode": "InProgress"  
}
```

El ID del trabajo se encuentra en la salida de `aws glacier initiate-job` y `aws glacier list-jobs`. Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

- Para API obtener más información, consulte [DescribeJob](#) la Referencia de AWS CLI comandos.

## describe-vault

En el siguiente ejemplo de código, se muestra cómo usar `describe-vault`.

## AWS CLI

El siguiente comando recupera datos sobre un almacén denominado `my-vault`:

```
aws glacier describe-vault --vault-name my-vault --account-id -
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

- Para API obtener más información, consulte [DescribeVault](#) la Referencia de AWS CLI comandos.

## get-data-retrieval-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-data-retrieval-policy`.

## AWS CLI

El siguiente comando obtiene la política de recuperación de datos de la cuenta en uso:

```
aws glacier get-data-retrieval-policy --account-id -
```

Salida:

```
{
  "Policy": {
    "Rules": [
      {
        "BytesPerHour": 10737418240,
        "Strategy": "BytesPerHour"
      }
    ]
  }
}
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

- Para API obtener más información, consulte la Referencia [GetDataRetrievalPolicy](#) de AWS CLI comandos.

## get-job-output

En el siguiente ejemplo de código, se muestra cómo usar `get-job-output`.

### AWS CLI

El siguiente comando guarda el resultado de un trabajo de inventario de almacén en un archivo del directorio actual denominado `output.json`:

```
aws glacier get-job-output --account-id - --vault-name my-vault --job-id zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3R1oGduS7Eg-R047Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqLNHS61ds04CnMW output.json
```

El `job-id` está disponible en la salida de `aws glacier list-jobs`. Tenga en cuenta que el nombre de archivo de salida es un argumento posicional que no va precedido de un nombre de opción. Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

Salida:

```
{
  "status": 200,
  "acceptRanges": "bytes",
  "contentType": "application/json"
}
```

`output.json`:

```
{"VaultARN":"arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault", "InventoryDate":"2015-04-07T00:26:18Z", "ArchiveList": [{"ArchiveId":"kKB7ymWJVpPSwhGP6ycS0Aekp9ZYe_--zM_mw6k76ZFGIWX-ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-AJV1u2ccmDSyDUmZwKbwbpAdGATGDiB3hH00bjbGehXTcApVud_wyDw", "ArchiveDescription":"multipart upload test", "CreationDate":"2015-04-06T22:24:34Z", "Size":3145728, "SHA256TreeHash":"9628195fcdcbcb
```

- Para API obtener más información, consulte [GetJobOutput](#) la Referencia de AWS CLI comandos.



## get-vault-access-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-vault-access-policy`.

### AWS CLI

Para recuperar la política de acceso de un almacén

En el siguiente `get-vault-access-policy` ejemplo, se recupera la política de acceso del almacén especificado.

```
aws glacier get-vault-access-policy \
  --account-id 111122223333 \
  --vault-name example_vault
```

Salida:

```
{
  "policy": {
    "Policy": "{\"Version\":\"2012-10-17\",\"Statement\": [{\"Effect\": \"Allow\", \"Principal\": {\"AWS\": \"arn:aws:iam::444455556666:root\"}, \"Action\": \"glacier:ListJobs\", \"Resource\": \"arn:aws:glacier:us-east-1:111122223333:vaults/example_vault\"}, {\"Effect\": \"Allow\", \"Principal\": {\"AWS\": \"arn:aws:iam::444455556666:root\"}, \"Action\": \"glacier:UploadArchive\", \"Resource\": \"arn:aws:glacier:us-east-1:111122223333:vaults/example_vault\"}]}"
  }
}
```

- Para API obtener más información, consulte [GetVaultAccessPolicy](#) la Referencia de AWS CLI comandos.

## get-vault-lock

En el siguiente ejemplo de código, se muestra cómo usar `get-vault-lock`.

### AWS CLI

Para obtener los detalles de una cerradura de bóveda

En el siguiente `get-vault-lock` ejemplo, se recuperaron los detalles sobre el candado de la bóveda especificada.

```
aws glacier get-vault-lock \  
  --account-id - \  
  --vault-name MyVaultName
```

Salida:

```
{  
  "Policy": "{\n\"Version\":\n\"2012-10-17\",  
\"Statement\":  
[[{\n\"Sid\":\n\"Define-vault-lock\",  
\"Effect\":\n\"Deny\",  
\"Principal\":  
{\n\"AWS\":\n\"arn:aws:iam:  
999999999999:root\"},  
\"Action\":\n\"glacier:DeleteArchive\",  
\"Resource\":\n\"arn:aws:glacier:us-west-2:  
999999999999:vaults/MyVaultName\",  
\"Condition\":  
{\n\"NumericLessThanEquals\":  
{\n\"glacier:ArchiveAgeinDays\":\n\"365\"}}}}]]",  
  "State": "Locked",  
  "CreationDate": "2019-07-29T22:25:28.640Z"  
}
```

Para [obtener más información, consulte `Get Vault GET Lock \(política de bloqueo\)`](#) en la Guía para API desarrolladores de Amazon Glacier.

- Para API obtener más información, consulte la Referencia [GetVaultLock](#) de AWS CLI comandos.

## get-vault-notifications

En el siguiente ejemplo de código, se muestra cómo usar `get-vault-notifications`.

AWS CLI

El siguiente comando obtiene una descripción de la configuración de notificaciones de un almacén denominado `my-vault`:

```
aws glacier get-vault-notifications --account-id - --vault-name my-vault
```

Salida:

```
{  
  "vaultNotificationConfig": {  
    "Events": [  
      "InventoryRetrievalCompleted",  
      "ArchiveRetrievalCompleted"  
    ]  
  }  
}
```

```

    ],
    "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault"
  }
}

```

Si no se ha configurado ninguna notificación para el almacén, se devuelve un error. Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

- Para API obtener más información, consulte [GetVaultNotifications](#) la Referencia de AWS CLI comandos.

## initiate-job

En el siguiente ejemplo de código, se muestra cómo usar `initiate-job`.

### AWS CLI

El siguiente comando inicia un trabajo para obtener un inventario del almacén `my-vault`:

```
aws glacier initiate-job --account-id - --vault-name my-vault --job-parameters '{"Type": "inventory-retrieval"}'
```

Salida:

```
{
  "location": "/0123456789012/vaults/my-vault/jobs/
zbxcm3Z_3z5UkooroF7SuZKrxgGoDc3RloGduS7Eg-
R047Yc6FxsdBgf_Q2DK5Ejh18CnTS5XW4_Xq1NHS61ds04CnMW",
  "jobId": "zbxcm3Z_3z5UkooroF7SuZKrxgGoDc3RloGduS7Eg-
R047Yc6FxsdBgf_Q2DK5Ejh18CnTS5XW4_Xq1NHS61ds04CnMW"
}
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

El siguiente comando inicia un trabajo para recuperar un archivo del almacén: `my-vault`

```
aws glacier initiate-job --account-id - --vault-name my-vault --job-parameters file://job-archive-retrieval.json
```

`job-archive-retrieval.json` es un JSON archivo de la carpeta local que especifica el tipo de trabajo, el identificador del archivo y algunos parámetros opcionales:

```
{
  "Type": "archive-retrieval",
  "ArchiveId": "kKB7ymWJVpPSwhGP6ycS0Aekp9ZYe_--zM_mw6k76ZFGElWQX-
ybtRDvc2VkJPSDtfKmQrj0IRQLSGsNuDp-
AJV1u2ccmDSyDUmZwKbwbpAdGATGDiB3hH00bjbGehXTcApVud_wyDw",
  "Description": "Retrieve archive on 2015-07-17",
  "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-topic"
}
```

Los archivos están disponibles en la salida de `aws glacier upload-archive` y `aws glacier get-job-output`.

Salida:

```
{
  "location": "/011685312445/vaults/mwunderl/jobs/17IL5-
EkXyEY9Ws95fClzIbk205uLYaFdAY0i-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJ059-8N9aHWav",
  "jobId": "17IL5-EkXy205uLYaFdAY0iEY9Ws95fClzIbk-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJ059-8N9aHWav"
}
```

Consulte [Initiate Job in the Amazon Glacier API Reference](#) para obtener más información sobre el formato de los parámetros del trabajo.

- Para API obtener más información, consulte [InitiateJob](#) la Referencia de AWS CLI comandos.

## **initiate-multipart-upload**

En el siguiente ejemplo de código, se muestra cómo usar `initiate-multipart-upload`.

### AWS CLI

El siguiente comando inicia una carga multiparte en un almacén cuyo nombre `my-vault` tiene un tamaño de parte de 1 MiB (1024 x 1024 bytes) por archivo:

```
aws glacier initiate-multipart-upload --account-id - --part-size 1048576 --vault-
name my-vault --archive-description "multipart upload test"
```

El parámetro de descripción del archivo es opcional. Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

Este comando genera un identificador de carga si se realiza correctamente. Usa el ID de carga cuando subas cada parte de tu archivo con `aws glacier upload-multipart-part`. Para obtener más información sobre las cargas multiparte a Amazon Glacier mediante el AWS CLI, consulte [Uso de Amazon Glacier en la Guía del AWS CLI usuario](#).

- Para API obtener más información, consulte la Referencia [InitiateMultipartUpload](#) de AWS CLI comandos.

## initiate-vault-lock

En el siguiente ejemplo de código, se muestra cómo usar `initiate-vault-lock`.

### AWS CLI

Para iniciar el proceso de bloqueo del depósito

El siguiente `initiate-vault-lock` ejemplo instala una política de bloqueo de bóveda en la bóveda especificada y establece el estado de bloqueo de la bóveda en `InProgress`. Debe completar el proceso llamando `complete-vault-lock` en un plazo de 24 horas para establecer el estado de bloqueo de la bóveda. `Locked`

```
aws glacier initiate-vault-lock \
  --account-id - \
  --vault-name MyVaultName \
  --policy file://vault_lock_policy.json
```

Contenidos de `vault_lock_policy.json`:

```
{"Policy":{"Version":"2012-10-17","Statement":[{"Sid":"Define-vault-lock","Effect":"Deny","Principal":{"AWS":"arn:aws:iam:999999999999:root"},"Action":"glacier:DeleteArchive","Resource":"arn:aws:glacier:us-west-2:999999999999:vaults/examplevault","Condition":{"NumericLessThanEquals":{"glacier:ArchiveAgeInDays":"365"}}}]}}
```

El resultado es el ID de bloqueo de la bóveda que puede utilizar para completar el proceso de bloqueo de la bóveda.

```
{
```

```
"lockId": "9QZgEXAMPLEPhvL6xEXAMPLE"
}
```

Para obtener más información, consulte [Iniciar Vault POST Lock \(política de bloqueo\)](#) en la Guía para API desarrolladores de Amazon Glacier.

- Para API obtener más información, consulte la Referencia [InitiateVaultLock](#) de AWS CLI comandos.

## list-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-jobs`.

### AWS CLI

El siguiente comando muestra los trabajos en curso y los que se han completado recientemente para un almacén denominado `my-vault`:

```
aws glacier list-jobs --account-id - --vault-name my-vault
```

Salida:

```
{
  "JobList": [
    {
      "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
      "RetrievalByteRange": "0-3145727",
      "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault",
      "Completed": false,
      "SHA256TreeHash":
"9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67",
      "JobId": "17IL5-EkXyEY9Ws95fClzIbk205uLYaFdAY0i-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJ059-8N9aHWav",
      "ArchiveId": "kKB7ymWJVpPSwhGP6ycS0Aekp9ZYe_--zM_mw6k76ZFGEIWQX-
ybtRDvc2VkJPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hH00bjbGehXTcApVud_wyDw",
      "JobDescription": "Retrieve archive on 2015-07-17",
      "ArchiveSizeInBytes": 3145728,
      "Action": "ArchiveRetrieval",
      "ArchiveSHA256TreeHash":
"9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67",
```

```

    "CreationDate": "2015-07-17T21:16:13.840Z",
    "StatusCode": "InProgress"
  },
  {
    "InventoryRetrievalParameters": {
      "Format": "JSON"
    },
    "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
    "Completed": false,
    "JobId": "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
R047Yc6FxsdBgf_Q2DK5Ejh18CnTS5XW4_Xq1NHS61ds04CnMW",
    "Action": "InventoryRetrieval",
    "CreationDate": "2015-07-17T20:23:41.616Z",
    "StatusCode": ""InProgress""
  }
]
}

```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

- Para API obtener más información, consulte [ListJobs](#) la Referencia de AWS CLI comandos.

## list-multipart-uploads

En el siguiente ejemplo de código, se muestra cómo usar `list-multipart-uploads`.

### AWS CLI

El siguiente comando muestra todas las cargas multiparte en curso para un almacén denominado: `my-vault`

```
aws glacier list-multipart-uploads --account-id - --vault-name my-vault
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

Para obtener más información sobre las cargas multiparte a Amazon Glacier mediante el AWS CLI, consulte [Uso de Amazon Glacier](#) en la Guía del AWS CLI usuario.

- Para API obtener más información, consulte la Referencia [ListMultipartUploads](#) de AWS CLI comandos.

## list-parts

En el siguiente ejemplo de código, se muestra cómo usar `list-parts`.

### AWS CLI

El siguiente comando muestra las partes cargadas para una carga de varias partes en un almacén denominado `my-vault`:

```
aws glacier list-parts --account-id - --vault-name my-vault --upload-id "SYZi7qnL-YGqGwAm8Kn3BLP2E1NCvnB-5961R09CSaPmPwkYGH0qeN_nX3-Vhnd2yF0KfB5FkmbnBU9GubbdıCs8ut-D"
```

Salida:

```
{
  "MultipartUploadId": "SYZi7qnL-YGqGwAm8Kn3BLP2E1NCvnB-5961R09CSaPmPwkYGH0qeN_nX3-Vhnd2yF0KfB5FkmbnBU9GubbdıCs8ut-D",
  "Parts": [
    {
      "RangeInBytes": "0-1048575",
      "SHA256TreeHash":
      "e1f2a7cd6e047350f69b9f8cfa60fa60fe2f02802097a9a026360a7edc1f553"
    },
    {
      "RangeInBytes": "1048576-2097151",
      "SHA256TreeHash":
      "43cf3061fb95796aed99a11a6aa3cd8f839eed15e655ab0a597126210636aee6"
    }
  ],
  "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
  "CreationDate": "2015-07-18T00:05:23.830Z",
  "PartSizeInBytes": 1048576
}
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

Para obtener más información sobre las cargas multiparte a Amazon Glacier mediante el AWS CLI, consulte [Uso de Amazon Glacier en la Guía del AWS CLI usuario](#).

- Para API obtener más información, consulte la Referencia [ListParts](#) de AWS CLI comandos.



## list-provisioned-capacity

En el siguiente ejemplo de código, se muestra cómo usar `list-provisioned-capacity`.

### AWS CLI

Para recuperar las unidades de capacidad aprovisionadas

En el siguiente `list-provisioned-capacity` ejemplo, se recuperan los detalles de cualquier unidad de capacidad aprovisionada para la cuenta especificada.

```
aws glacier list-provisioned-capacity \  
  --account-id 111122223333
```

Salida:

```
{  
  "ProvisionedCapacityList": [  
    {  
      "CapacityId": "HpASAUvfRFiVDb0jMfEIcr8K",  
      "ExpirationDate": "2020-03-18T19:59:24.000Z",  
      "StartDate": "2020-02-18T19:59:24.912Z"  
    }  
  ]  
}
```

- Para API obtener más información, consulte la Referencia [ListProvisionedCapacity](#) de AWS CLI comandos.

## list-tags-for-vault

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-vault`.

### AWS CLI

El siguiente comando muestra las etiquetas aplicadas a un almacén denominado `my-vault`:

```
aws glacier list-tags-for-vault --account-id - --vault-name my-vault
```

Salida:

```
{
  "Tags": {
    "date": "july2015",
    "id": "1234"
  }
}
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

- Para API obtener más información, consulte [ListTagsForVault](#) la Referencia de AWS CLI comandos.

## list-vaults

En el siguiente ejemplo de código, se muestra cómo usar `list-vaults`.

### AWS CLI

El siguiente comando muestra los almacenes de la cuenta y región predeterminadas:

```
aws glacier list-vaults --account-id -
```

### Salida:

```
{
  "VaultList": [
    {
      "SizeInBytes": 3178496,
      "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
      "LastInventoryDate": "2015-04-07T00:26:19.028Z",
      "VaultName": "my-vault",
      "NumberOfArchives": 1,
      "CreationDate": "2015-04-06T21:23:45.708Z"
    }
  ]
}
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

- Para API obtener más información, consulte [ListVaults](#) la Referencia de AWS CLI comandos.

## **purchase-provisioned-capacity**

En el siguiente ejemplo de código, se muestra cómo usar `purchase-provisioned-capacity`.

AWS CLI

Para comprar una unidad de capacidad aprovisionada

En el siguiente `purchase-provisioned-capacity` ejemplo, se compra una unidad de capacidad aprovisionada.

```
aws glacier purchase-provisioned-capacity \  
  --account-id 111122223333
```

Salida:

```
{  
  "capacityId": "HpASAUvfRFiVDb0jMfEIcr8K"  
}
```

- Para API obtener más información, consulte [PurchaseProvisionedCapacity](#) la Referencia de AWS CLI comandos.

## **remove-tags-from-vault**

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags-from-vault`.

AWS CLI

El siguiente comando elimina una etiqueta con la clave `date` de un almacén denominado `my-vault`:

```
aws glacier remove-tags-from-vault --account-id - --vault-name my-vault --tag-  
keys date
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

- Para API obtener más información, consulte [RemoveTagsFromVault](#) la Referencia de AWS CLI comandos.

## set-data-retrieval-policy

En el siguiente ejemplo de código, se muestra cómo usar `set-data-retrieval-policy`.

### AWS CLI

El siguiente comando configura una política de recuperación de datos para la cuenta en uso:

```
aws glacier set-data-retrieval-policy --account-id - --policy file://data-retrieval-policy.json
```

`data-retrieval-policy.json` es un JSON archivo de la carpeta actual que especifica una política de recuperación de datos:

```
{
  "Rules": [
    {
      "Strategy": "BytesPerHour",
      "BytesPerHour": 10737418240
    }
  ]
}
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

El siguiente comando establece que la política de recuperación de datos se `FreeTier` utilice en línea: JSON

```
aws glacier set-data-retrieval-policy --account-id - --policy '{"Rules": [{"Strategy": "FreeTier"}]}'
```

Consulte Establecer una política de recuperación de datos en la API referencia de Amazon Glacier para obtener más información sobre el formato de la política.

- Para API obtener más información, consulte [SetDataRetrievalPolicy](#) la Referencia de AWS CLI comandos.

## set-vault-access-policy

En el siguiente ejemplo de código, se muestra cómo usar `set-vault-access-policy`.

### AWS CLI

Para configurar la política de acceso de un almacén

En el siguiente `set-vault-access-policy` ejemplo, se adjunta una política de permisos al almacén especificado.

```
aws glacier set-vault-access-policy \
  --account-id 111122223333 \
  --vault-name example_vault
  --policy '{"Policy": "{\"Version\":\"2012-10-17\",\"Statement\":
  [ {\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::444455556666:root
  \"},\"Action\":\"glacier:ListJobs\",\"Resource\":\"arn:aws:glacier:us-
  east-1:111122223333:vaults/example_vault\"}, {\"Effect\":\"Allow\",\"Principal\":
  {\"AWS\":\"arn:aws:iam::444455556666:root\"},\"Action\":\"glacier:UploadArchive\",
  \"Resource\":\"arn:aws:glacier:us-east-1:111122223333:vaults/example_vault\"} ] }"'
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SetVaultAccessPolicy](#) la Referencia de AWS CLI comandos.

## set-vault-notifications

En el siguiente ejemplo de código, se muestra cómo usar `set-vault-notifications`.

### AWS CLI

El siguiente comando configura SNS las notificaciones de un almacén denominado `my-vault`:

```
aws glacier set-vault-notifications --account-id - --vault-name my-vault --vault-
notification-config file://notificationconfig.json
```

`notificationconfig.json` es un JSON archivo de la carpeta actual que especifica un SNS tema y los eventos que se van a publicar:

```
{
```

```
"SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault",
"Events": ["ArchiveRetrievalCompleted", "InventoryRetrievalCompleted"]
}
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

- Para API obtener más información, consulte [SetVaultNotifications](#) la Referencia de AWS CLI comandos.

## upload-archive

En el siguiente ejemplo de código, se muestra cómo usar `upload-archive`.

### AWS CLI

El siguiente comando carga un archivo de la carpeta actual denominado `archive.zip` en un almacén denominado `my-vault`:

```
aws glacier upload-archive --account-id - --vault-name my-vault --body archive.zip
```

Salida:

```
{
  "archiveId": "kKB7ymWJVpPSwhGP6ycS0Aekp9ZYe_--zM_mw6k76ZFGIEWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hH00bjbGehXTcApVud_wyDw",
  "checksum": "969fb39823836d81f0cc028195fcdcbbe76cdde932d4646fa7de5f21e18aa67",
  "location": "/0123456789012/vaults/my-vault/archives/
kKB7ymWJVpPSwhGP6ycS0Aekp9ZYe_--zM_mw6k76ZFGIEWQX-ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hH00bjbGehXTcApVud_wyDw"
}
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

Para recuperar un archivo cargado, inicie un trabajo de recuperación con el comando `aws glacier initiate-job`.

- Para API obtener más información, consulte [UploadArchive](#) la Referencia de AWS CLI comandos.

## upload-multipart-part

En el siguiente ejemplo de código, se muestra cómo usar `upload-multipart-part`.

### AWS CLI

El siguiente comando carga la primera parte de 1 MiB (1024 x 1024 bytes) de un archivo:

```
aws glacier upload-multipart-part --body part1 --range 'bytes  
0-1048575/*' --account-id - --vault-name my-vault --upload-  
id 19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-  
0ssZtLqyFu7sY1_1R7vgFuJV6NtcV5zpsJ
```

Amazon Glacier requiere un argumento de ID de cuenta al realizar operaciones, pero puede usar un guion para especificar la cuenta en uso.

El parámetro `body` toma una ruta a un archivo de parte del sistema de archivos local. El parámetro `range` toma un rango de HTTP contenido que indica los bytes que ocupa la pieza en el archivo completo. El comando `aws glacier initiate-multipart-upload` devuelve el ID de carga y también se puede obtener mediante `aws glacier list-multipart-uploads`.

Para obtener más información sobre las cargas multiparte a Amazon Glacier mediante el AWS CLI, consulte [Uso de Amazon Glacier](#) en la Guía del AWS CLI usuario.

- Para API obtener más información, consulte la Referencia [UploadMultipartPart](#) de AWS CLI comandos.

## Ejemplos de Secrets Manager usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Secrets Manager.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### batch-get-secret-value

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-secret-value`.

#### AWS CLI

Ejemplo 1: Para recuperar el valor secreto de un grupo de secretos enumerados por nombre

En el siguiente `batch-get-secret-value` ejemplo, se obtienen los valores secretos de tres secretos.

```
aws secretsmanager batch-get-secret-value \  
  --secret-id-list MySecret1 MySecret2 MySecret3
```

Salida:

```
{  
  "SecretValues": [  
    {  
      "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret1-  
a1b2c3",  
      "Name": "MySecret1",  
      "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaa",  
      "SecretString": "{\"username\":\"diego_ramirez\",\"password\":\"EXAMPLE-  
PASSWORD\",\"engine\":\"mysql\",\"host\":\"secretsmanagertutorial.cluster.us-  
west-2.rds.amazonaws.com\",\"port\":3306,\"dbClusterIdentifier\":  
\"secretsmanagertutorial\"}",  
      "VersionStages": [  
        "AWSCURRENT"  
      ],  
      "CreateDate": "1523477145.729"  
    },  
    {  
      "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret2-  
a1b2c3",  
      "Name": "MySecret2",  
      "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbbb",
```



```

        "SecretString": "{\"username\":\"akua_mansa\", \"password\":\"EXAMPLE-
PASSWORD\""},
        "VersionStages": [
            "AWSCURRENT"
        ],
        "CreateDate": "1673477781.275"
    },
    {
        "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret3-
a1b2c3",
        "Name": "MySecret3",
        "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEecccc",
        "SecretString": "{\"username\":\"jie_liu\", \"password\":\"EXAMPLE-
PASSWORD\""},
        "VersionStages": [
            "AWSCURRENT"
        ],
        "CreateDate": "1373477721.124"
    }
],
"Errors": []
}

```

Para obtener más información, consulte [Recuperar un grupo de secretos en un lote](#) en la Guía del usuario de AWS Secrets Manager.

Ejemplo 2: Para recuperar el valor secreto de un grupo de secretos seleccionado por el filtro

En el siguiente `batch-get-secret-value` ejemplo, se obtienen los valores secretos de tu cuenta que figuran `MySecret` en el nombre. El filtrado por nombres distingue entre mayúsculas y minúsculas.

```

aws secretsmanager batch-get-secret-value \
  --filters Key="name",Values="MySecret"

```

Salida:

```

{
  "SecretValues": [
    {
      "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret1-
a1b2c3",

```

```

        "Name": "MySecret1",
        "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa",
        "SecretString": "{\"username\":\"diego_ramirez\", \"password\":\"EXAMPLE-
PASSWORD\", \"engine\":\"mysql\", \"host\":\"secretsmanagertutorial.cluster.us-
west-2.rds.amazonaws.com\", \"port\":3306, \"dbClusterIdentifier\":
\"secretsmanagertutorial\"}",
        "VersionStages": [
            "AWSCURRENT"
        ],
        "CreateDate": "1523477145.729"
    },
    {
        "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret2-
a1b2c3",
        "Name": "MySecret2",
        "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbbb",
        "SecretString": "{\"username\":\"akua_mansa\", \"password\":\"EXAMPLE-
PASSWORD\"}",
        "VersionStages": [
            "AWSCURRENT"
        ],
        "CreateDate": "1673477781.275"
    },
    {
        "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret3-
a1b2c3",
        "Name": "MySecret3",
        "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEccccc",
        "SecretString": "{\"username\":\"jie_liu\", \"password\":\"EXAMPLE-
PASSWORD\"}",
        "VersionStages": [
            "AWSCURRENT"
        ],
        "CreateDate": "1373477721.124"
    }
    ],
    "Errors": []
}

```

Para obtener más información, consulte [Recuperar un grupo de secretos en un lote](#) en la Guía del usuario de AWS Secrets Manager.

- Para API obtener más información, consulte [BatchGetSecretValue](#) la Referencia de AWS CLI comandos.

## cancel-rotate-secret

En el siguiente ejemplo de código, se muestra cómo usar `cancel-rotate-secret`.

### AWS CLI

Para desactivar la rotación automática de un secreto

En el siguiente `cancel-rotate-secret` ejemplo, se desactiva la rotación automática de un secreto. Para reanudar la rotación, llamar `rotate-secret`.

```
aws secretsmanager cancel-rotate-secret \  
  --secret-id MyTestSecret
```

Salida:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-a1b2c3",  
  "Name": "MyTestSecret"  
}
```

Para obtener más información, consulte [Rotar un secreto](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [CancelRotateSecret](#) la Referencia de AWS CLI comandos.

## create-secret

En el siguiente ejemplo de código, se muestra cómo usar `create-secret`.

### AWS CLI

Ejemplo 1: Creación de un secreto

En el siguiente ejemplo de `create-secret` se crea un secreto con dos pares clave-valor.

```
aws secretsmanager create-secret \  
  --name MyTestSecret \  
  --description "My test secret created with the CLI." \  
  --value test
```

```
--secret-string "{\"user\":\"diegor\",\"password\":\"EXAMPLE-PASSWORD\"}"
```

Salida:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-a1b2c3",
  "Name": "MyTestSecret",
  "VersionId": "EXAMPLE1-90ab-cdef-fedc-ba987EXAMPLE"
}
```

Para obtener más información, consulte [Creación de un secreto](#) en la Guía del usuario de Secrets Manager.

Ejemplo 2: Para crear un secreto a partir de las credenciales de un JSON archivo

En el siguiente ejemplo de `create-secret`, se crea un secreto a partir de las credenciales de un archivo. Para obtener más información, consulte [Carga de AWS CLI parámetros de un archivo](#) en la Guía del AWS CLI usuario.

```
aws secretsmanager create-secret \
  --name MyTestSecret \
  --secret-string file://mycreds.json
```

Contenidos de `mycreds.json`:

```
{
  "engine": "mysql",
  "username": "saanvis",
  "password": "EXAMPLE-PASSWORD",
  "host": "my-database-endpoint.us-west-2.rds.amazonaws.com",
  "dbname": "myDatabase",
  "port": "3306"
}
```

Salida:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-a1b2c3",
  "Name": "MyTestSecret",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Para obtener más información, consulte [Creación de un secreto](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [CreateSecret](#) la Referencia de AWS CLI comandos.

## **delete-resource-policy**

En el siguiente ejemplo de código, se muestra cómo usar `delete-resource-policy`.

### AWS CLI

Para eliminar la política basada en recursos adjunta a un secreto

En el siguiente ejemplo de `delete-resource-policy` se elimina la política basada en recursos asociada a un secreto.

```
aws secretsmanager delete-resource-policy \  
--secret-id MyTestSecret
```

Salida:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret"  
}
```

Para obtener más información, consulte [Autenticación y control de acceso](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [DeleteResourcePolicy](#) la Referencia de AWS CLI comandos.

## **delete-secret**

En el siguiente ejemplo de código, se muestra cómo usar `delete-secret`.

### AWS CLI

Ejemplo 1: Eliminación de un secreto

En el siguiente ejemplo de `delete-secret` se elimina un secreto. Se puede recuperar el secreto con `restore-secret` hasta la fecha y hora del campo de respuesta `DeletionDate`. Para eliminar un secreto que se replica en otras regiones, primero elimine sus réplicas con `remove-regions-from-replication` y, a continuación, llame a `delete-secret`.

```
aws secretsmanager delete-secret \  
  --secret-id MyTestSecret \  
  --recovery-window-in-days 7
```

Salida:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret",  
  "DeletionDate": 1524085349.095  
}
```

Para obtener más información, consulte [Eliminación de un secreto](#) en la Guía del usuario de Secrets Manager.

Ejemplo 2: Eliminación de un secreto inmediatamente

En el siguiente ejemplo de `delete-secret` se elimina un secreto inmediatamente sin periodo de recuperación. Este secreto no se puede recuperar.

```
aws secretsmanager delete-secret \  
  --secret-id MyTestSecret \  
  --force-delete-without-recovery
```

Salida:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret",  
  "DeletionDate": 1508750180.309  
}
```

Para obtener más información, consulte [Eliminación de un secreto](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [DeleteSecret](#) la Referencia de AWS CLI comandos.

## describe-secret

En el siguiente ejemplo de código, se muestra cómo usar `describe-secret`.

### AWS CLI

Recuperación de los detalles de un secreto

En el siguiente ejemplo `describe-secret`, se muestran detalles de un secreto.

```
aws secretsmanager describe-secret \  
  --secret-id MyTestSecret
```

Salida:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
Ca8JGt",  
  "Name": "MyTestSecret",  
  "Description": "My test secret",  
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/EXAMPLE1-90ab-cdef-fedc-  
ba987EXAMPLE",  
  "RotationEnabled": true,  
  "RotationLambdaARN": "arn:aws:lambda:us-  
west-2:123456789012:function:MyTestRotationLambda",  
  "RotationRules": {  
    "AutomaticallyAfterDays": 2,  
    "Duration": "2h",  
    "ScheduleExpression": "cron(0 16 1,15 * ? *)"  
  },  
  "LastRotatedDate": 1525747253.72,  
  "LastChangedDate": 1523477145.729,  
  "LastAccessedDate": 1524572133.25,  
  "Tags": [  
    {  
      "Key": "SecondTag",  
      "Value": "AnotherValue"  
    },  
    {  
      "Key": "FirstTag",  
      "Value": "SomeValue"  
    }  
  ]  
}
```

```

    }
  ],
  "VersionIdsToStages": {
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111": [
      "AWSPREVIOUS"
    ],
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222": [
      "AWSCURRENT"
    ],
    "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333": [
      "AWSPENDING"
    ]
  },
  "CreateDate": 1521534252.66,
  "PrimaryRegion": "us-west-2",
  "ReplicationStatus": [
    {
      "Region": "eu-west-3",
      "KmsKeyId": "alias/aws/secretsmanager",
      "Status": "InSync",
      "StatusMessage": "Replication succeeded"
    }
  ]
}

```

Para obtener más información, consulte [Secreto](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [DescribeSecret](#) la Referencia de AWS CLI comandos.

## get-random-password

En el siguiente ejemplo de código, se muestra cómo usar `get-random-password`.

### AWS CLI

Para generar una contraseña aleatoria

El siguiente `get-random-password` ejemplo genera una contraseña aleatoria de 20 caracteres que incluye al menos una letra mayúscula, una letra minúscula, un número y un signo de puntuación.

```
aws secretsmanager get-random-password \
```



```
--require-each-included-type \
--password-length 20
```

Salida:

```
{
  "RandomPassword": "EXAMPLE-PASSWORD"
}
```

Para obtener más información, consulte [Crear y administrar secretos](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [GetRandomPassword](#) la Referencia de AWS CLI comandos.

## get-resource-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-resource-policy`.

AWS CLI

Para recuperar la política basada en recursos adjunta a un secreto

En el siguiente ejemplo de `get-resource-policy` se recupera la política basada en recursos asociada a un secreto.

```
aws secretsmanager get-resource-policy \
--secret-id MyTestSecret
```

Salida:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
  "Name": "MyTestSecret",
  "ResourcePolicy": "{\n  \"Version\": \"2012-10-17\",\n  \"Statement\": [\n    {\n      \"Effect\":\n\"Allow\",\n      \"Principal\": {\n        \"AWS\": \"arn:aws:iam::123456789012:root\"\n      },\n      \"Action\":\n\"secretsmanager:GetSecretValue\",\n      \"Resource\": \"*\"\n    }\n  ]\n}"
}
```

Para obtener más información, consulte [Autenticación y control de acceso](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [GetResourcePolicy](#) la Referencia de AWS CLI comandos.

## get-secret-value

En el siguiente ejemplo de código, se muestra cómo usar `get-secret-value`.

### AWS CLI

Ejemplo 1: Recuperación del valor de secreto cifrado de un secreto

El siguiente ejemplo de `get-secret-value` obtiene el valor de secreto actual.

```
aws secretsmanager get-secret-value \  
  --secret-id MyTestSecret
```

Salida:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret",  
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "SecretString": "{\"user\":\"diegor\",\"password\":\"EXAMPLE-PASSWORD\"}",  
  "VersionStages": [  
    "AWSCURRENT"  
  ],  
  "CreateDate": 1523477145.713  
}
```

Para obtener más información, consulte [Recuperación de un secreto](#) en la Guía del usuario de Secrets Manager.

Ejemplo 2: Recuperación del valor de secreto anterior

El siguiente ejemplo de `get-secret-value` obtiene el valor de secreto anterior:

```
aws secretsmanager get-secret-value \  
  --secret-id MyTestSecret
```

```
--version-stage AWSPREVIOUS
```

Salida:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
  "Name": "MyTestSecret",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "SecretString": "{\"user\":\"diegor\",\"password\":\"PREVIOUS-EXAMPLE-PASSWORD
}\"",
  "VersionStages": [
    "AWSPREVIOUS"
  ],
  "CreateDate": 1523477145.713
}
```

Para obtener más información, consulte [Recuperación de un secreto](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [GetSecretValue](#) la Referencia de AWS CLI comandos.

## **list-secret-version-ids**

En el siguiente ejemplo de código, se muestra cómo usar `list-secret-version-ids`.

AWS CLI

Para enumerar todas las versiones secretas asociadas a un secreto

En el `list-secret-version-ids` ejemplo siguiente se obtiene una lista de todas las versiones de un secreto.

```
aws secretsmanager list-secret-version-ids \
  --secret-id MyTestSecret
```

Salida:

```
{
  "Versions": [
    {
```

```

    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "VersionStages": [
      "AWSPREVIOUS"
    ],
    "LastAccessedDate": 1523477145.713,
    "CreatedDate": 1523477145.713
  },
  {
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "VersionStages": [
      "AWSCURRENT"
    ],
    "LastAccessedDate": 1523477145.713,
    "CreatedDate": 1523486221.391
  },
  {
    "CreatedDate": 1.51197446236E9,
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333;"
  }
],
"ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
"Name": "MyTestSecret"
}

```

Para obtener más información, consulte la [versión](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [ListSecretVersionIds](#) la Referencia de AWS CLI comandos.

## list-secrets

En el siguiente ejemplo de código, se muestra cómo usar `list-secrets`.

### AWS CLI

Ejemplo 1: Creación de una lista de los secretos de la cuenta

En el siguiente ejemplo de `list-secrets` se obtiene una lista de los secretos de la cuenta.

```
aws secretsmanager list-secrets
```

Salida:

```
{
  "SecretList": [
    {
      "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-a1b2c3",
      "Name": "MyTestSecret",
      "LastChangedDate": 1523477145.729,
      "SecretVersionsToStages": {
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111": [
          "AWSCURRENT"
        ]
      }
    },
    {
      "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:AnotherSecret-d4e5f6",
      "Name": "AnotherSecret",
      "LastChangedDate": 1523482025.685,
      "SecretVersionsToStages": {
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222": [
          "AWSCURRENT"
        ]
      }
    }
  ]
}
```

Para obtener más información acerca de Secrets Manager, consulte [Encontrar un secreto](#) en la Guía del usuario de Secrets Manager.

### Ejemplo 2: Filtrado de la lista de secretos de la cuenta

En el siguiente ejemplo de `list-secrets` se obtiene una lista de los secretos de la cuenta que incluyen Test en su nombre. El filtrado por nombres distingue entre mayúsculas y minúsculas.

```
aws secretsmanager list-secrets \
  --filter Key="name",Values="Test"
```

Salida:

```
{
  "SecretList": [
```

```

    {
      "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-a1b2c3",
      "Name": "MyTestSecret",
      "LastChangedDate": 1523477145.729,
      "SecretVersionsToStages": {
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111": [
          "AWSCURRENT"
        ]
      }
    }
  ]
}

```

Para obtener más información acerca de Secrets Manager, consulte [Encontrar un secreto](#) en la Guía del usuario de Secrets Manager.

Ejemplo 3: Creación de una lista de los secretos de su cuenta administrada por otro servicio

En el siguiente `list-secrets` ejemplo, se muestran los secretos de tu cuenta gestionados por AmazonRDS.

```

aws secretsmanager list-secrets \
  --filter Key="owning-service",Values="rds"

```

Salida:

```

{
  "SecretList": [
    {
      "Name": "rds!cluster-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Tags": [
        {
          "Value": "arn:aws:rds:us-
west-2:123456789012:cluster:database-1",
          "Key": "aws:rds:primaryDBClusterArn"
        },
        {
          "Value": "rds",
          "Key": "aws:secretsmanager:owningService"
        }
      ]
    }
  ],
}

```

```

    "RotationRules": {
      "AutomaticallyAfterDays": 1
    },
    "LastChangedDate": 1673477781.275,
    "LastRotatedDate": 1673477781.26,
    "SecretVersionsToStages": {
      "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa": [
        "AWSPREVIOUS"
      ],
      "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbb": [
        "AWSCURRENT",
        "AWSPENDING"
      ]
    },
    "OwningService": "rds",
    "RotationEnabled": true,
    "CreatedDate": 1673467300.7,
    "LastAccessedDate": 1673395200.0,
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:rds!
cluster-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111-a1b2c3",
    "Description": "Secret associated with primary RDS DB cluster:
arn:aws:rds:us-west-2:123456789012:cluster:database-1"
  }
]
}

```

Para obtener más información, consulte [Secretos gestionados por otros servicios](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [ListSecrets](#) la Referencia de AWS CLI comandos.

## put-resource-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-resource-policy`.

### AWS CLI

Para añadir una política basada en recursos a un secreto

En el siguiente ejemplo de `put-resource-policy` se agrega una política de permisos a un secreto, pero primero se comprueba que la política no proporciona un acceso amplio al secreto. La política se lee desde un archivo. Para obtener más información, consulte [Carga de AWS CLI parámetros desde un archivo](#) en la Guía del AWS CLI usuario.

```
aws secretsmanager put-resource-policy \  
  --secret-id MyTestSecret \  
  --resource-policy file://mypolicy.json \  
  --block-public-policy
```

Contenidos de `mypolicy.json`:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Principal": {  
        "AWS": "arn:aws:iam::123456789012:role/MyRole"  
      },  
      "Action": "secretsmanager:GetSecretValue",  
      "Resource": "*"   
    }  
  ]  
}
```

Salida:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-a1b2c3",  
  "Name": "MyTestSecret"  
}
```

Para obtener más información, consulte [Adjuntar una política de permisos a un secreto](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [PutResourcePolicy](#) la Referencia de AWS CLI comandos.

## put-secret-value

En el siguiente ejemplo de código, se muestra cómo usar `put-secret-value`.

### AWS CLI

Ejemplo 1: Almacenamiento de un nuevo valor de secreto en un secreto



En el siguiente ejemplo de `put-secret-value` se crea una nueva versión de un secreto con dos pares clave-valor.

```
aws secretsmanager put-secret-value \  
  --secret-id MyTestSecret \  
  --secret-string "{\"user\":\"diegor\",\"password\":\"EXAMPLE-PASSWORD\"}"
```

Salida:

```
{  
  "ARN": "arn:aws:secretsmanager:us-  
west-2:123456789012:secret:MyTestSecret-1a2b3c",  
  "Name": "MyTestSecret",  
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
  "VersionStages": [  
    "AWSCURRENT"  
  ]  
}
```

Para obtener más información, consulte [Modificación de un secreto](#) en la Guía del usuario de Secrets Manager.

Ejemplo 2: Para almacenar un nuevo valor secreto de las credenciales en un JSON archivo

En el siguiente ejemplo de `put-secret-value` se crea una nueva versión de un secreto a partir de las credenciales de un archivo. Para obtener más información, consulte [Carga de AWS CLI parámetros de un archivo](#) en la Guía del AWS CLI usuario.

```
aws secretsmanager put-secret-value \  
  --secret-id MyTestSecret \  
  --secret-string file://mycreds.json
```

Contenidos de `mycreds.json`:

```
{  
  "engine": "mysql",  
  "username": "saanvis",  
  "password": "EXAMPLE-PASSWORD",  
  "host": "my-database-endpoint.us-west-2.rds.amazonaws.com",  
  "dbname": "myDatabase",
```

```
"port": "3306"
}
```

Salida:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
  "Name": "MyTestSecret",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "VersionStages": [
    "AWSCURRENT"
  ]
}
```

Para obtener más información, consulte [Modificación de un secreto](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [PutSecretValue](#) la Referencia de AWS CLI comandos.

## **remove-regions-from-replication**

En el siguiente ejemplo de código, se muestra cómo usar `remove-regions-from-replication`.

AWS CLI

Para eliminar un secreto de réplica

En el siguiente ejemplo de `remove-regions-from-replication` se elimina un secreto de réplica de `eu-west-3`. Para eliminar un secreto principal que se replica en otras regiones, primero elimine las réplicas y, a continuación, llame a `delete-secret`.

```
aws secretsmanager remove-regions-from-replication \
  --secret-id MyTestSecret \
  --remove-replica-regions eu-west-3
```

Salida:

```
{
```

```
"ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-1a2b3c",
  "ReplicationStatus": []
}
```

Para obtener más información, consulte [Eliminar un secreto de réplica](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [RemoveRegionsFromReplication](#) la Referencia de AWS CLI comandos.

## replicate-secret-to-regions

En el siguiente ejemplo de código, se muestra cómo usar `replicate-secret-to-regions`.

### AWS CLI

Para replicar un secreto en otra región

En el siguiente ejemplo de `replicate-secret-to-regions` se replica un secreto en eu-west-3. La réplica se cifra con la clave AWS gestionada `aws/secretsmanager`.

```
aws secretsmanager replicate-secret-to-regions \
  --secret-id MyTestSecret \
  --add-replica-regions Region=eu-west-3
```

Salida:

```
{
  "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-1a2b3c",
  "ReplicationStatus": [
    {
      "Region": "eu-west-3",
      "KmsKeyId": "alias/aws/secretsmanager",
      "Status": "InProgress"
    }
  ]
}
```

Para obtener más información, consulte [Replicar un secreto en otra región](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [ReplicateSecretToRegions](#) la Referencia de AWS CLI comandos.

## restore-secret

En el siguiente ejemplo de código, se muestra cómo usar `restore-secret`.

### AWS CLI

Para restaurar un secreto previamente borrado

En el siguiente ejemplo de `restore-secret` se restaura un secreto cuya eliminación se había programado previamente.

```
aws secretsmanager restore-secret \  
  --secret-id MyTestSecret
```

Salida:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret"  
}
```

Para obtener más información, consulte [Eliminación de un secreto](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [RestoreSecret](#) la Referencia de AWS CLI comandos.

## rotate-secret

En el siguiente ejemplo de código, se muestra cómo usar `rotate-secret`.

### AWS CLI

Ejemplo 1: Para configurar e iniciar la rotación automática de un secreto

El siguiente `rotate-secret` ejemplo configura e inicia la rotación automática de un secreto. Secrets Manager cambia el secreto una vez de forma inmediata y, después, cada ocho horas en

un período de dos horas. El resultado muestra `VersionId` la nueva versión secreta creada por rotación.

```
aws secretsmanager rotate-secret \
  --secret-id MyTestDatabaseSecret \
  --rotation-lambda-arn arn:aws:lambda:us-west-2:1234566789012:function:SecretsManagerTestRotationLambda \
  --rotation-rules "{\"ScheduleExpression\": \"cron(0 8/8 * * ? *)\", \"Duration\": \"2h\"}"
```

Salida:

```
{
  "ARN": "aws:arn:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-a1b2c3",
  "Name": "MyTestDatabaseSecret",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Para obtener más información, consulte [Rotar secretos](#) en la Guía del usuario de Secrets Manager.

Ejemplo 2: Para configurar e iniciar la rotación automática en un intervalo de rotación

El siguiente `rotate-secret` ejemplo configura e inicia la rotación automática de un secreto. Secrets Manager cambia el secreto una vez inmediatamente y luego cada 10 días. El resultado muestra `VersionId` la nueva versión secreta creada por rotación.

```
aws secretsmanager rotate-secret \
  --secret-id MyTestDatabaseSecret \
  --rotation-lambda-arn arn:aws:lambda:us-west-2:1234566789012:function:SecretsManagerTestRotationLambda \
  --rotation-rules "{\"ScheduleExpression\": \"rate(10 days)\", \"Duration\": \"10d\"}"
```

Salida:

```
{
  "ARN": "aws:arn:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-a1b2c3",
  "Name": "MyTestDatabaseSecret",
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

```
}
```

Para obtener más información, consulte [Rotar secretos](#) en la Guía del usuario de Secrets Manager.

Ejemplo 3: Para rotar un secreto de forma inmediata

En el siguiente ejemplo de `rotate-secret` se inicia una rotación inmediata. El resultado muestra `VersionId` la nueva versión secreta creada por rotación. El secreto ya debe tener configurada la rotación.

```
aws secretsmanager rotate-secret \  
  --secret-id MyTestDatabaseSecret
```

Salida:

```
{  
  "ARN": "aws:arn:secretsmanager:us-  
west-2:123456789012:secret:MyTestDatabaseSecret-a1b2c3",  
  "Name": "MyTestDatabaseSecret",  
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
}
```

Para obtener más información, consulte [Rotar secretos](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [RotateSecret](#) la Referencia de AWS CLI comandos.

## stop-replication-to-replica

En el siguiente ejemplo de código, se muestra cómo usar `stop-replication-to-replica`.

### AWS CLI

Para convertir un secreto de réplica en uno principal

En el siguiente ejemplo de `stop-replication-to-replica`, se elimina el enlace entre un secreto de réplica y el principal. El secreto de réplica se promociona a secreto principal en la región de réplica. Debe llamar a `stop-replication-to-replica` desde la región de réplica.

```
aws secretsmanager stop-replication-to-replica \  
  --secret-id MyTestDatabaseSecret
```

```
--secret-id MyTestSecret
```

Salida:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3"
}
```

Para obtener más información, consulte [Promocionar un secreto de réplica](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [StopReplicationToReplica](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

AWS CLI

Ejemplo 1: Para añadir una etiqueta a un secreto

En el siguiente ejemplo de se muestra cómo asociar una etiqueta con sintaxis abreviada.

```
aws secretsmanager tag-resource \  
  --secret-id MyTestSecret \  
  --tags Key=FirstTag,Value=FirstValue
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar sus secretos](#) en la Guía del usuario de Secrets Manager.

Ejemplo 2: Para añadir varias etiquetas a un secreto

En el siguiente ejemplo de `tag-resource` se asocian dos etiquetas de clave-valor a un secreto.

```
aws secretsmanager tag-resource \  
  --secret-id MyTestSecret \  
  --tags ' [{"Key": "FirstTag", "Value": "FirstValue"}, {"Key": "SecondTag",  
"Value": "SecondValue"} ]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar secretos](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar etiquetas de un secreto

En el siguiente ejemplo de `untag-resource` se eliminan dos etiquetas de un secreto. Se eliminan tanto la clave como el valor de cada etiqueta.

```
aws secretsmanager untag-resource \  
  --secret-id MyTestSecret \  
  --tag-keys '[ "FirstTag", "SecondTag" ]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetar secretos](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-secret-version-stage

En el siguiente ejemplo de código, se muestra cómo usar `update-secret-version-stage`.

### AWS CLI

Ejemplo 1: Para revertir un secreto a la versión anterior

En el siguiente `update-secret-version-stage` ejemplo, se mueve `AWS CURRENT` la etiqueta provisional a la versión anterior de un secreto, lo que revierte el secreto a la versión anterior. Para encontrar el ID de la versión anterior, utilice `list-secret-version-ids`



Para este ejemplo, la versión con la etiqueta es `a1b2c3d4-5678-90ab-cdef-` y la versión con la `AWS CURRENT` etiqueta es `a1b2c3d4-5678-90ab-cdef-`. `EXAMPLE11111` `AWS PREVIOUS` `EXAMPLE22222` En este ejemplo, mueves `AWS CURRENT` la etiqueta de la versión `11111` a la `22222`. Como la `AWS CURRENT` etiqueta se ha eliminado de una versión, la `AWS PREVIOUS` mueve `update-secret-version-stage` automáticamente a esa versión (`11111`). El efecto es que las `AWS PREVIOUS` versiones `AWS CURRENT` y se intercambian.

```
aws secretsmanager update-secret-version-stage \
  --secret-id MyTestSecret \
  --version-stage AWSCURRENT \
  --move-to-version-id a1b2c3d4-5678-90ab-cdef-EXAMPLE22222 \
  --remove-from-version-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
  "Name": "MyTestSecret"
}
```

Para obtener más información, consulte la [versión](#) en la Guía del usuario de Secrets Manager.

Ejemplo 2: Para añadir una etiqueta provisional adjunta a una versión de un secreto

En el siguiente `update-secret-version-stage` ejemplo, se agrega una etiqueta provisional a una versión de un secreto. Para revisar los resultados, ejecute `list-secret-version-ids` y visualice el campo de `VersionStages` respuesta de la versión afectada.

```
aws secretsmanager update-secret-version-stage \
  --secret-id MyTestSecret \
  --version-stage STAGINGLABEL1 \
  --move-to-version-id EXAMPLE1-90ab-cdef-fedc-ba987EXAMPLE
```

Salida:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
  "Name": "MyTestSecret"
```

```
}
```

Para obtener más información, consulte la [versión](#) en la Guía del usuario de Secrets Manager.

Ejemplo 3: Para eliminar una etiqueta provisional adjunta a una versión de un secreto

En el siguiente `update-secret-version-stage` ejemplo, se elimina una etiqueta provisional adjunta a una versión de un secreto. Para revisar los resultados, ejecute `list-secret-version-ids` y visualice el campo de `VersionStages` respuesta de la versión afectada.

```
aws secretsmanager update-secret-version-stage \  
  --secret-id MyTestSecret \  
  --version-stage STAGINGLABEL1 \  
  --remove-from-version-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-  
a1b2c3",  
  "Name": "MyTestSecret"  
}
```

Para obtener más información, consulte la [versión](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [UpdateSecretVersionStage](#) la Referencia de AWS CLI comandos.

## update-secret

En el siguiente ejemplo de código, se muestra cómo usar `update-secret`.

### AWS CLI

Ejemplo 1: Actualización de la descripción de un secreto

En el siguiente ejemplo de `update-secret` se actualiza la descripción de un secreto.

```
aws secretsmanager update-secret \  
  --secret-id MyTestSecret \  
  --description "This is a new description for the secret."
```

Salida:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
  "Name": "MyTestSecret"
}
```

Para obtener más información, consulte [Modificación de un secreto](#) en la Guía del usuario de Secrets Manager.

Ejemplo 2: Actualización de la clave de cifrado asociada a un secreto

El siguiente `update-secret` ejemplo actualiza la KMS clave utilizada para cifrar el valor secreto. La KMS clave debe estar en la misma región que el secreto.

```
aws secretsmanager update-secret \
  --secret-id MyTestSecret \
  --kms-key-id arn:aws:kms:us-west-2:123456789012:key/EXAMPLE1-90ab-cdef-fedc-
ba987EXAMPLE
```

Salida:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
  "Name": "MyTestSecret"
}
```

Para obtener más información, consulte [Modificación de un secreto](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [UpdateSecret](#) la Referencia de AWS CLI comandos.

## **validate-resource-policy**

En el siguiente ejemplo de código, se muestra cómo usar `validate-resource-policy`.

AWS CLI

Para validar una política de recursos

En el siguiente `validate-resource-policy` ejemplo, se comprueba que una política de recursos no concede un acceso amplio a un secreto. La política se lee desde un archivo del disco. Para obtener más información, consulte [Carga de AWS CLI parámetros desde un archivo](#) en la Guía del AWS CLI usuario.

```
aws secretsmanager validate-resource-policy \  
--resource-policy file://mypolicy.json
```

Contenidos de `mypolicy.json`:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Principal": {  
        "AWS": "arn:aws:iam::123456789012:role/MyRole"  
      },  
      "Action": "secretsmanager:GetSecretValue",  
      "Resource": "*"   
    }  
  ]  
}
```

Salida:

```
{  
  "PolicyValidationPassed": true,  
  "ValidationErrors": []  
}
```

Para obtener más información, consulte la [referencia de permisos para Secrets Manager](#) en la Guía del usuario de Secrets Manager.

- Para API obtener más información, consulte [ValidateResourcePolicy](#) la Referencia de AWS CLI comandos.

## Ejemplos de Security Hub que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface with Security Hub.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **accept-administrator-invitation**

En el siguiente ejemplo de código, se muestra cómo usar `accept-administrator-invitation`.

AWS CLI

Para aceptar una invitación de una cuenta de administrador

El siguiente `accept-administrator-invitation` ejemplo acepta la invitación especificada de la cuenta de administrador especificada.

```
aws securityhub accept-invitation \  
  --administrator-id 123456789012 \  
  --invitation-id 7ab938c5d52d7904ad09f9e7c20cc4eb
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [AcceptAdministratorInvitation](#) la Referencia de AWS CLI comandos.

### **accept-invitation**

En el siguiente ejemplo de código, se muestra cómo usar `accept-invitation`.

AWS CLI

Para aceptar una invitación de una cuenta de administrador

El siguiente `accept-invitation` ejemplo acepta la invitación especificada de la cuenta de administrador especificada.

```
aws securityhub accept-invitation \  
  --master-id 123456789012 \  
  --invitation-id 7ab938c5d52d7904ad09f9e7c20cc4eb
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [AcceptInvitation](#) la Referencia de AWS CLI comandos.

## batch-delete-automation-rules

En el siguiente ejemplo de código, se muestra cómo usar `batch-delete-automation-rules`.

### AWS CLI

Para eliminar reglas de automatización

En el siguiente `batch-delete-automation-rules` ejemplo, se elimina la regla de automatización especificada. Puede eliminar una o más reglas con un solo comando. Solo la cuenta de administrador de Security Hub puede ejecutar este comando.

```
aws securityhub batch-delete-automation-rules \  
  --automation-rules-arns '["arn:aws:securityhub:us-  
east-1:123456789012:automation-rule/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"]'
```

Salida:

```
{  
  "ProcessedAutomationRules": [  
    "arn:aws:securityhub:us-east-1:123456789012:automation-rule/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
  ],  
  "UnprocessedAutomationRules": []  
}
```

Para obtener más información, consulte [Eliminar reglas de automatización](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [BatchDeleteAutomationRules](#) la Referencia de AWS CLI comandos.

## batch-disable-standards

En el siguiente ejemplo de código, se muestra cómo usar batch-disable-standards.

### AWS CLI

Para deshabilitar un estándar

El siguiente batch-disable-standards ejemplo deshabilita el estándar asociado a la suscripción ARN especificada.

```
aws securityhub batch-disable-standards \  
  --standards-subscription-arns "arn:aws:securityhub:us-  
west-1:123456789012:subscription/pci-dss/v/3.2.1"
```

Salida:

```
{  
  "StandardsSubscriptions": [  
    {  
      "StandardsArn": "arn:aws:securityhub:eu-central-1::standards/pci-dss/  
v/3.2.1",  
      "StandardsInput": { },  
      "StandardsStatus": "DELETING",  
      "StandardsSubscriptionArn": "arn:aws:securityhub:us-  
west-1:123456789012:subscription/pci-dss/v/3.2.1"  
    }  
  ]  
}
```

Para obtener más información, consulte [Deshabilitar o habilitar un estándar de seguridad](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [BatchDisableStandards](#) la Referencia de AWS CLI comandos.

## batch-enable-standards

En el siguiente ejemplo de código, se muestra cómo usar `batch-enable-standards`.

### AWS CLI

Para habilitar un estándar

El siguiente `batch-enable-standards` ejemplo habilita el PCI DSS estándar para la cuenta solicitante.

```
aws securityhub batch-enable-standards \  
  --standards-subscription-requests '{"StandardsArn":"arn:aws:securityhub:us-  
west-1::standards/pci-dss/v/3.2.1"}'
```

Salida:

```
{  
  "StandardsSubscriptions": [  
    {  
      "StandardsArn": "arn:aws:securityhub:us-west-1::standards/pci-dss/  
v/3.2.1",  
      "StandardsInput": { },  
      "StandardsStatus": "PENDING",  
      "StandardsSubscriptionArn": "arn:aws:securityhub:us-  
west-1:123456789012:subscription/pci-dss/v/3.2.1"  
    }  
  ]  
}
```

Para obtener más información, consulte [Deshabilitar o habilitar un estándar de seguridad](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [BatchEnableStandards](#) la Referencia de AWS CLI comandos.

## batch-get-automation-rules

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-automation-rules`.

### AWS CLI

Para obtener detalles sobre las reglas de automatización



En el siguiente `batch-get-automation-rules` ejemplo, se obtienen detalles de la regla de automatización especificada. Puede obtener detalles de una o más reglas de automatización con un solo comando.

```
aws securityhub batch-get-automation-rules \  
  --automation-rules-arns '["arn:aws:securityhub:us-  
east-1:123456789012:automation-rule/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"]'
```

Salida:

```
{  
  "Rules": [  
    {  
      "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "RuleStatus": "ENABLED",  
      "RuleOrder": 1,  
      "RuleName": "Suppress informational findings",  
      "Description": "Suppress GuardDuty findings with Informational  
severity",  
      "IsTerminal": false,  
      "Criteria": {  
        "ProductName": [  
          {  
            "Value": "GuardDuty",  
            "Comparison": "EQUALS"  
          }  
        ],  
        "SeverityLabel": [  
          {  
            "Value": "INFORMATIONAL",  
            "Comparison": "EQUALS"  
          }  
        ],  
        "WorkflowStatus": [  
          {  
            "Value": "NEW",  
            "Comparison": "EQUALS"  
          }  
        ],  
        "RecordState": [  
          {  
            "Value": "ACTIVE",
```

```

        "Comparison": "EQUALS"
      }
    ]
  },
  "Actions": [
    {
      "Type": "FINDING_FIELDS_UPDATE",
      "FindingFieldsUpdate": {
        "Note": {
          "Text": "Automatically suppress GuardDuty findings with
Informational severity",
          "UpdatedBy": "sechub-automation"
        },
        "Workflow": {
          "Status": "SUPPRESSED"
        }
      }
    }
  ],
  "CreatedAt": "2023-05-31T17:56:14.837000+00:00",
  "UpdatedAt": "2023-05-31T17:59:38.466000+00:00",
  "CreatedBy": "arn:aws:iam::123456789012:role/Admin"
}
],
"UnprocessedAutomationRules": []
}

```

Para obtener más información, consulte [Visualización de las reglas de automatización](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [BatchGetAutomationRules](#) la Referencia de AWS CLI comandos.

## batch-get-configuration-policy-associations

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-configuration-policy-associations`.

### AWS CLI

Para obtener los detalles de la asociación de configuración de un lote de objetivos

En el siguiente `batch-get-configuration-policy-associations` ejemplo, se recuperan los detalles de la asociación de los objetivos especificados. Puedes proporcionar la cuentaIDs, la unidad IDs organizativa o el ID raíz del objetivo.

```
aws securityhub batch-get-configuration-policy-associations \  
  --target '{"OrganizationalUnitId": "ou-6hi7-8j91k12m"}'
```

Salida:

```
{  
  "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",  
  "TargetId": "ou-6hi7-8j91k12m",  
  "TargetType": "ORGANIZATIONAL_UNIT",  
  "AssociationType": "APPLIED",  
  "UpdatedAt": "2023-09-26T21:13:01.816000+00:00",  
  "AssociationStatus": "SUCCESS",  
  "AssociationStatusMessage": "Association applied successfully on this target."  
}
```

Para obtener más información, consulte [Visualización de las políticas de configuración de Security Hub](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [BatchGetConfigurationPolicyAssociations](#) la Referencia de AWS CLI comandos.

## batch-get-security-controls

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-security-controls`.

### AWS CLI

Para obtener los detalles del control de seguridad

En el siguiente `batch-get-security-controls` ejemplo, se obtienen los detalles de los controles de seguridad ACM IAM .1 y .1 de la AWS cuenta corriente y AWS la región.

```
aws securityhub batch-get-security-controls \  
  --security-control-ids ['ACM.1', 'IAM.1']
```

Salida:

```
{
  "SecurityControls": [
    {
      "SecurityControlId": "ACM.1",
      "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/ACM.1",
      "Title": "Imported and ACM-issued certificates should be renewed after a
specified time period",
      "Description": "This control checks whether an AWS Certificate Manager
(ACM) certificate is renewed within the specified time period. It checks both
imported certificates and certificates provided by ACM. The control fails if the
certificate isn't renewed within the specified time period. Unless you provide a
custom parameter value for the renewal period, Security Hub uses a default value of
30 days.",
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
ACM.1/remediation",
      "SeverityRating": "MEDIUM",
      "SecurityControlStatus": "ENABLED"
      "UpdateStatus": "READY",
      "Parameters": {
        "daysToExpiration": {
          "ValueType": CUSTOM,
          "Value": {
            "Integer": 15
          }
        }
      },
      "LastUpdateReason": "Updated control parameter"
    },
    {
      "SecurityControlId": "IAM.1",
      "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/IAM.1",
      "Title": "IAM policies should not allow full \"*\" administrative
privileges",
      "Description": "This AWS control checks whether the default version of
AWS Identity and Access Management (IAM) policies (also known as customer managed
policies) do not have administrator access with a statement that has \"Effect\":
\"Allow\" with \"Action\": \"*\" over \"Resource\": \"*\". It only checks for
the Customer Managed Policies that you created, but not inline and AWS Managed
Policies.",
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
IAM.1/remediation",
```

```

        "SeverityRating": "HIGH",
        "SecurityControlStatus": "ENABLED"
        "UpdateStatus": "READY",
        "Parameters": {}
    }
]
}

```

Para obtener más información, consulte [Visualización de los detalles de un control](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [BatchGetSecurityControls](#) la Referencia de AWS CLI comandos.

## batch-get-standards-control-associations

En el siguiente ejemplo de código, se muestra cómo usar `batch-get-standards-control-associations`.

### AWS CLI

Para obtener el estado de activación de un control

El siguiente `batch-get-standards-control-associations` ejemplo identifica si los controles especificados están habilitados en los estándares especificados.

```

aws securityhub batch-get-standards-control-associations \
  --standards-control-association-ids '["SecurityControlId":
  "Config.1", "StandardsArn": "arn:aws:securityhub:us-east-1:123456789012:ruleset/cis-
  aws-foundations-benchmark/v/1.2.0"}, {"SecurityControlId": "IAM.6", "StandardsArn":
  "arn:aws:securityhub:us-east-1:123456789012:standards/aws-foundational-security-
  best-practices/v/1.0.0"}]'

```

Salida:

```

{
  "StandardsControlAssociationDetails": [
    {
      "StandardsArn": "arn:aws:securityhub::ruleset/cis-aws-foundations-
      benchmark/v/1.2.0",
      "SecurityControlId": "Config.1",

```

```

    "SecurityControlArn": "arn:aws:securityhub:us-
east-1:068873283051:security-control/Config.1",
    "AssociationStatus": "ENABLED",
    "RelatedRequirements": [
        "CIS AWS Foundations 2.5"
    ],
    "UpdatedAt": "2022-10-27T16:07:12.960000+00:00",
    "StandardsControlTitle": "Ensure AWS Config is enabled",
    "StandardsControlDescription": "AWS Config is a web service that
performs configuration management of supported AWS resources within your account
and delivers log files to you. The recorded information includes the configuration
item (AWS resource), relationships between configuration items (AWS resources), and
any configuration changes between resources. It is recommended to enable AWS Config
in all regions.",
    "StandardsControlArns": [
        "arn:aws:securityhub:us-east-1:068873283051:control/cis-aws-
foundations-benchmark/v/1.2.0/2.5"
    ]
},
{
    "StandardsArn": "arn:aws:securityhub:us-east-1::standards/aws-
foundational-security-best-practices/v/1.0.0",
    "SecurityControlId": "IAM.6",
    "SecurityControlArn": "arn:aws:securityhub:us-
east-1:068873283051:security-control/IAM.6",
    "AssociationStatus": "DISABLED",
    "RelatedRequirements": [],
    "UpdatedAt": "2022-11-22T21:30:35.080000+00:00",
    "UpdatedReason": "test",
    "StandardsControlTitle": "Hardware MFA should be enabled for the root
user",
    "StandardsControlDescription": "This AWS control checks whether your AWS
account is enabled to use a hardware multi-factor authentication (MFA) device to
sign in with root user credentials.",
    "StandardsControlArns": [
        "arn:aws:securityhub:us-east-1:068873283051:control/aws-
foundational-security-best-practices/v/1.0.0/IAM.6"
    ]
}
]
}

```

Para obtener más información, consulte [Habilitar y deshabilitar controles en estándares específicos en](#) la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [BatchGetStandardsControlAssociations](#) la Referencia de AWS CLI comandos.

## batch-import-findings

En el siguiente ejemplo de código, se muestra cómo usar `batch-import-findings`.

### AWS CLI

Para actualizar un hallazgo

En el `batch-import-findings` ejemplo siguiente se actualiza un hallazgo.

```
aws securityhub batch-import-findings \
  --findings '
    [{
      "AwsAccountId": "123456789012",
      "CreatedAt": "2020-05-27T17:05:54.832Z",
      "Description": "Vulnerability in a CloudTrail trail",
      "FindingProviderFields": {
        "Severity": {
          "Label": "LOW",
          "Original": "10"
        },
        "Types": [
          "Software and Configuration Checks/Vulnerabilities/CVE"
        ]
      },
      "GeneratorId": "TestGeneratorId",
      "Id": "Id1",
      "ProductArn": "arn:aws:securityhub:us-
west-1:123456789012:product/123456789012/default",
      "Resources": [
        {
          "Id": "arn:aws:cloudtrail:us-west-1:123456789012:trail/
TrailName",
          "Partition": "aws",
          "Region": "us-west-1",
          "Type": "AwsCloudTrailTrail"
        }
      ]
    }
  ]
```

```

    ],
    "SchemaVersion": "2018-10-08",
    "Title": "CloudTrail trail vulnerability",
    "UpdatedAt": "2020-06-02T16:05:54.832Z"
  }]'

```

Salida:

```

{
  "FailedCount": 0,
  "SuccessCount": 1,
  "FailedFindings": []
}

```

Para obtener más información, consulte [Usar BatchImportFindings para crear y actualizar hallazgos](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [BatchImportFindings](#) la Referencia de AWS CLI comandos.

## batch-update-automation-rules

En el siguiente ejemplo de código, se muestra cómo usar batch-update-automation-rules.

AWS CLI

Para actualizar las reglas de automatización

En el siguiente batch-update-automation-rules ejemplo, se actualiza la regla de automatización especificada. Puede actualizar una o más reglas con un solo comando. Solo la cuenta de administrador de Security Hub puede ejecutar este comando.

```

aws securityhub batch-update-automation-rules \
  --update-automation-rules-request-items '[ \
    { \
      "Actions": [{ \
        "Type": "FINDING_FIELDS_UPDATE", \
        "FindingFieldsUpdate": { \
          "Note": { \
            "Text": "Known issue that is a risk", \
            "UpdatedBy": "sechub-automation" \

```



```

        }, \
        "Workflow": { \
            "Status": "NEW" \
        } \
    } \
}], \
"Criteria": { \
    "SeverityLabel": [{ \
        "Value": "LOW", \
        "Comparison": "EQUALS" \
    }] \
}, \
"RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111", \
"RuleOrder": 1, \
"RuleStatus": "DISABLED" \
} \
]'

```

Salida:

```

{
  "ProcessedAutomationRules": [
    "arn:aws:securityhub:us-east-1:123456789012:automation-rule/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  ],
  "UnprocessedAutomationRules": []
}

```

Para obtener más información, consulte [Edición de reglas de automatización](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [BatchUpdateAutomationRules](#) la Referencia de AWS CLI comandos.

## batch-update-findings

En el siguiente ejemplo de código, se muestra cómo usar batch-update-findings.

### AWS CLI

Ejemplo 1: Para actualizar un hallazgo

En el siguiente `batch-update-findings` ejemplo, se actualizan dos resultados para añadir una nota, cambiar la etiqueta de gravedad y resolverlos.

```
aws securityhub batch-update-findings \
  --finding-identifiers '[{"Id": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111", "ProductArn": "arn:aws:securityhub:us-
west-1::product/aws/securityhub"}, {"Id": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222", "ProductArn": "arn:aws:securityhub:us-
west-1::product/aws/securityhub"}]' \
  --note '{"Text": "Known issue that is not a risk.", "UpdatedBy": "user1"}' \
  --severity '{"Label": "LOW"}' \
  --workflow '{"Status": "RESOLVED"}'
```

Salida:

```
{
  "ProcessedFindings": [
    {
      "Id": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/
v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub"
    },
    {
      "Id": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/
v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub"
    }
  ],
  "UnprocessedFindings": []
}
```

Para obtener más información, consulte [Uso BatchUpdateFindings para actualizar un hallazgo](#) en la Guía del usuario de AWS Security Hub.

Ejemplo 2: Para actualizar un hallazgo mediante una sintaxis abreviada

En el siguiente `batch-update-findings` ejemplo, se actualizan dos resultados para añadir una nota, cambiar la etiqueta de gravedad y resolverlos mediante una sintaxis abreviada.

```
aws securityhub batch-update-findings \
```

```

--finding-identifiers Id="arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",ProductArn="arn:aws:securityhub:us-
west-1::product/aws/securityhub" Id="arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",ProductArn="arn:aws:securityhub:us-
west-1::product/aws/securityhub" \
--note Text="Known issue that is not a risk.",UpdatedBy="user1" \
--severity Label="LOW" \
--workflow Status="RESOLVED"

```

Salida:

```

{
  "ProcessedFindings": [
    {
      "Id": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/
v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub"
    },
    {
      "Id": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/
v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
      "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub"
    }
  ],
  "UnprocessedFindings": []
}

```

Para obtener más información, consulte [Uso BatchUpdateFindings para actualizar un hallazgo](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [BatchUpdateFindings](#) la Referencia de AWS CLI comandos.

## batch-update-standards-control-associations

En el siguiente ejemplo de código, se muestra cómo usar batch-update-standards-control-associations.

## AWS CLI

Para actualizar el estado de activación de un control en los estándares habilitados

El siguiente `batch-update-standards-control-associations` ejemplo desactiva CloudTrail .1 en los estándares especificados.

```
aws securityhub batch-update-standards-control-associations \  
  --standards-control-association-updates '[{"SecurityControlId": "CloudTrail.1",  
  "StandardsArn": "arn:aws:securityhub::ruleset/cis-aws-foundations-benchmark/  
v/1.2.0", "AssociationStatus": "DISABLED", "UpdatedReason": "Not applicable  
to environment"}, {"SecurityControlId": "CloudTrail.1", "StandardsArn":  
"arn:aws:securityhub::standards/cis-aws-foundations-benchmark/v/1.4.0",  
"AssociationStatus": "DISABLED", "UpdatedReason": "Not applicable to  
environment"}]'
```

Este comando no genera ninguna salida si se realiza correctamente.

Para obtener más información, consulte [Habilitar y deshabilitar controles en estándares específicos](#) y [Habilitar y deshabilitar controles en todos los estándares en](#) la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte la Referencia [BatchUpdateStandardsControlAssociations](#) de AWS CLI comandos.

## **create-action-target**

En el siguiente ejemplo de código, se muestra cómo usar `create-action-target`.

## AWS CLI

Para crear una acción personalizada

En el siguiente `create-action-target` ejemplo, se crea una acción personalizada. Proporciona el nombre, la descripción y el identificador de la acción.

```
aws securityhub create-action-target \  
  --name "Send to remediation" \  
  --description "Action to send the finding for remediation tracking" \  
  --id "Remediation"
```

Salida:

```
{
  "ActionTargetArn": "arn:aws:securityhub:us-west-1:123456789012:action/custom/Remediation"
}
```

Para obtener más información, consulte [Crear una acción personalizada y asociarla a una regla de CloudWatch eventos](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [CreateActionTarget](#) la Referencia de AWS CLI comandos.

## create-automation-rule

En el siguiente ejemplo de código, se muestra cómo usar `create-automation-rule`.

### AWS CLI

Para crear una regla de automatización

En el siguiente `create-automation-rule` ejemplo, se crea una regla de automatización en la AWS cuenta corriente y en AWS la región. Security Hub filtra sus hallazgos en función de los criterios especificados y aplica las acciones a los hallazgos coincidentes. Solo la cuenta de administrador de Security Hub puede ejecutar este comando.

```
aws securityhub create-automation-rule \
  --actions '[{ \
    "Type": "FINDING_FIELDS_UPDATE", \
    "FindingFieldsUpdate": { \
      "Severity": { \
        "Label": "HIGH" \
      }, \
      "Note": { \
        "Text": "Known issue that is a risk. Updated by automation rules", \
        "UpdatedBy": "sechub-automation" \
      } \
    } \
  }]' \
  --criteria '{ \
    "SeverityLabel": [{ \
      "Value": "INFORMATIONAL", \
```

```

        "Comparison": "EQUALS" \
      }] \
    }' \
    --description "A sample rule" \
    --no-is-terminal \
    --rule-name "sample rule" \
    --rule-order 1 \
    --rule-status "ENABLED"

```

Salida:

```

{
  "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}

```

Para obtener más información, consulte [Creación de reglas de automatización](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [CreateAutomationRule](#) la Referencia de AWS CLI comandos.

## create-configuration-policy

En el siguiente ejemplo de código, se muestra cómo usar `create-configuration-policy`.

AWS CLI

Para crear una política de configuración

En el siguiente `create-configuration-policy` ejemplo, se crea una política de configuración con los parámetros especificados.

```

aws securityhub create-configuration-policy \
  --name "SampleConfigurationPolicy" \
  --description "SampleDescription" \
  --configuration-policy '{"SecurityHub": {"ServiceEnabled":
    true, "EnabledStandardIdentifiers": ["arn:aws:securityhub:eu-
    central-1::standards/aws-foundational-security-best-practices/
    v/1.0.0", "arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/
    v/1.2.0"], "SecurityControlsConfiguration": {"DisabledSecurityControlIdentifiers":
    ["CloudTrail.2"], "SecurityControlCustomParameters": [{"SecurityControlId":

```

```
"ACM.1", "Parameters": {"daysToExpiration": {"ValueType": "CUSTOM", "Value": {"Integer": 15}}}}]]}}' \
--tags '{"Environment": "Prod"}'
```

Salida:

```
{
  "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Name": "SampleConfigurationPolicy",
  "Description": "SampleDescription",
  "UpdatedAt": "2023-11-28T20:28:04.494000+00:00",
  "CreatedAt": "2023-11-28T20:28:04.494000+00:00",
  "ConfigurationPolicy": {
    "SecurityHub": {
      "ServiceEnabled": true,
      "EnabledStandardIdentifiers": [
        "arn:aws:securityhub:eu-central-1::standards/aws-foundational-security-best-practices/v/1.0.0",
        "arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/v/1.2.0"
      ],
      "SecurityControlsConfiguration": {
        "DisabledSecurityControlIdentifiers": [
          "CloudTrail.2"
        ],
        "SecurityControlCustomParameters": [
          {
            "SecurityControlId": "ACM.1",
            "Parameters": {
              "daysToExpiration": {
                "ValueType": "CUSTOM",
                "Value": {
                  "Integer": 15
                }
              }
            }
          }
        ]
      }
    }
  }
}
```

```
}
```

Para obtener más información, consulte [Creación y asociación de políticas de configuración de Security Hub](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [CreateConfigurationPolicy](#) la Referencia de AWS CLI comandos.

## **create-finding-aggregator**

En el siguiente ejemplo de código, se muestra cómo usar `create-finding-aggregator`.

### AWS CLI

Para habilitar la búsqueda de agregación

El siguiente `create-finding-aggregator` ejemplo configura la búsqueda de la agregación. Se ejecuta desde EE. UU. Este (Virginia), que designa EE. UU. Este (Virginia) como región de agregación. Indica que solo se deben vincular regiones específicas y que no se deben vincular automáticamente regiones nuevas. Selecciona EE.UU. Oeste (Norte de California) y EE.UU. Oeste (Oregón) como regiones vinculadas.

```
aws securityhub create-finding-aggregator \  
  --region us-east-1 \  
  --region-linking-mode SPECIFIED_REGIONS \  
  --regions us-west-1,us-west-2
```

Salida:

```
{  
  "FindingAggregatorArn": "arn:aws:securityhub:us-east-1:222222222222:finding-  
aggregator/123e4567-e89b-12d3-a456-426652340000",  
  "FindingAggregationRegion": "us-east-1",  
  "RegionLinkingMode": "SPECIFIED_REGIONS",  
  "Regions": "us-west-1,us-west-2"  
}
```

Para obtener más información, consulte [Habilitar la búsqueda de agregación](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [CreateFindingAggregator](#) la Referencia de AWS CLI comandos.



## create-insight

En el siguiente ejemplo de código, se muestra cómo usar `create-insight`.

### AWS CLI

Para crear una visión personalizada

En el siguiente `create-insight` ejemplo, se crea una información personalizada denominada Critical role Findings que devuelve las conclusiones críticas relacionadas con las AWS funciones.

```
aws securityhub create-insight \  
  --filters '{"ResourceType": [{"Comparison": "EQUALS", "Value": "AwsIamRole"}],  
  "SeverityLabel": [{"Comparison": "EQUALS", "Value": "CRITICAL"}]}' \  
  --group-by-attribute "ResourceId" \  
  --name "Critical role findings"
```

Salida:

```
{  
  "InsightArn": "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/  
  custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE111111"  
}
```

Para obtener más información, consulte [Administrar información personalizada](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [CreateInsight](#) la Referencia de AWS CLI comandos.

## create-members

En el siguiente ejemplo de código, se muestra cómo usar `create-members`.

### AWS CLI

Para añadir cuentas como cuentas de miembros

En el siguiente `create-members` ejemplo, se agregan dos cuentas como cuentas de miembro a la cuenta de administrador solicitante.

```
aws securityhub create-members \  
  --members "arn:aws:iam::123456789012:role/Role1", "arn:aws:iam::123456789012:role/Role2"
```

```
--account-details '[{"AccountId": "123456789111"}, {"AccountId":  
"123456789222"}]'
```

Salida:

```
{  
  "UnprocessedAccounts": []  
}
```

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [CreateMembers](#) la Referencia de AWS CLI comandos.

## decline-invitations

En el siguiente ejemplo de código, se muestra cómo usar `decline-invitations`.

AWS CLI

Para rechazar una invitación a ser miembro de una cuenta

El siguiente `decline-invitations` ejemplo rechaza una invitación a ser miembro de la cuenta de administrador especificada. La cuenta de miembro es la cuenta solicitante.

```
aws securityhub decline-invitations \  
  --account-ids "123456789012"
```

Salida:

```
{  
  "UnprocessedAccounts": []  
}
```

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [DeclineInvitations](#) la Referencia de AWS CLI comandos.

## delete-action-target

En el siguiente ejemplo de código, se muestra cómo usar delete-action-target.

### AWS CLI

Para eliminar una acción personalizada

En el siguiente delete-action-target ejemplo, se elimina la acción personalizada identificada por la especificadaARN.

```
aws securityhub delete-action-target \  
  --action-target-arn "arn:aws:securityhub:us-west-1:123456789012:action/custom/  
  Remediation"
```

Salida:

```
{  
  "ActionTargetArn": "arn:aws:securityhub:us-west-1:123456789012:action/custom/  
  Remediation"  
}
```

Para obtener más información, consulte [Crear una acción personalizada y asociarla a una regla de CloudWatch eventos](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [DeleteActionTarget](#) la Referencia de AWS CLI comandos.

## delete-configuration-policy

En el siguiente ejemplo de código, se muestra cómo usar delete-configuration-policy.

### AWS CLI

Eliminación de una política de configuración

En el siguiente delete-configuration-policy ejemplo, se elimina la política de configuración especificada.

```
aws securityhub delete-configuration-policy \  
  --configuration-policy-name "arn:aws:securityhub:us-west-1:123456789012:action/custom/  
  Remediation"
```

```
--identifier "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar y desasociar las políticas de configuración de Security Hub](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte la Referencia [DeleteConfigurationPolicy](#) de AWS CLI comandos.

## delete-finding-aggregator

En el siguiente ejemplo de código, se muestra cómo usar delete-finding-aggregator.

### AWS CLI

Para dejar de buscar agregación

El siguiente delete-finding-aggregator ejemplo deja de buscar la agregación. Se ejecuta desde EE. UU. Este (Virginia), que es la región de agregación.

```
aws securityhub delete-finding-aggregator \  
  --region us-east-1 \  
  --finding-aggregator-arn arn:aws:securityhub:us-east-1:222222222222:finding-aggregator/123e4567-e89b-12d3-a456-426652340000
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Dejar de buscar agregación](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [DeleteFindingAggregator](#) la Referencia de AWS CLI comandos.

## delete-insight

En el siguiente ejemplo de código, se muestra cómo usar delete-insight.

### AWS CLI

Para eliminar una información personalizada

En el siguiente delete-insight ejemplo, se elimina la información personalizada con la especificadaARN.

```
aws securityhub delete-insight \  
  --insight-arn "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/  
  custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Salida:

```
{  
  "InsightArn": "arn:aws:securityhub:eu-  
  central-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-  
  EXAMPLE11111"  
}
```

Para obtener más información, consulte [Administrar información personalizada](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [DeleteInsight](#) la Referencia de AWS CLI comandos.

## delete-invitations

En el siguiente ejemplo de código, se muestra cómo usar delete-invitations.

AWS CLI

Para eliminar una invitación a ser miembro de una cuenta

En el siguiente delete-invitations ejemplo, se elimina una invitación a ser miembro de la cuenta de administrador especificada. La cuenta de miembro es la cuenta solicitante.

```
aws securityhub delete-invitations \  
  --account-ids "123456789012"
```

Salida:

```
{  
  "UnprocessedAccounts": []  
}
```

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [DeleteInvitations](#) la Referencia de AWS CLI comandos.

## **delete-members**

En el siguiente ejemplo de código, se muestra cómo usar `delete-members`.

### AWS CLI

Para eliminar las cuentas de los miembros

En el siguiente `delete-members` ejemplo, se eliminan las cuentas de miembro especificadas de la cuenta de administrador solicitante.

```
aws securityhub delete-members \  
  --account-ids "123456789111" "123456789222"
```

Salida:

```
{  
  "UnprocessedAccounts": []  
}
```

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [DeleteMembers](#) la Referencia de AWS CLI comandos.

## **describe-action-targets**

En el siguiente ejemplo de código, se muestra cómo usar `describe-action-targets`.

### AWS CLI

Para recuperar detalles sobre las acciones personalizadas

En el siguiente `describe-action-targets` ejemplo, se recupera información sobre la acción personalizada identificada por la especificadaARN.

```
aws securityhub describe-action-targets \  
  --action-target-arns "arn:aws:securityhub:us-west-1:123456789012:action/custom/  
Remediation"
```

Salida:

```
{  
  "ActionTargets": [  
    {  
      "ActionTargetArn": "arn:aws:securityhub:us-west-1:123456789012:action/  
custom/Remediation",  
      "Description": "Action to send the finding for remediation tracking",  
      "Name": "Send to remediation"  
    }  
  ]  
}
```

Para obtener más información, consulte [Crear una acción personalizada y asociarla a una regla de CloudWatch eventos](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [DescribeActionTargets](#) la Referencia de AWS CLI comandos.

## describe-hub

En el siguiente ejemplo de código, se muestra cómo usar describe-hub.

### AWS CLI

Para obtener información sobre un recurso central

El siguiente describe-hub ejemplo devuelve la fecha de suscripción del recurso hub especificado. El recurso hub se identifica por suARN.

```
aws securityhub describe-hub \  
  --hub-arn "arn:aws:securityhub:us-west-1:123456789012:hub/default"
```

Salida:

```
{
```

```
"HubArn": "arn:aws:securityhub:us-west-1:123456789012:hub/default",
"SubscribedAt": "2019-11-19T23:15:10.046Z"
}
```

Para obtener más información, consulte [AWS:SecurityHub: :Hub](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [DescribeHub](#) la Referencia de AWS CLI comandos.

## describe-organization-configuration

En el siguiente ejemplo de código, se muestra cómo usar `describe-organization-configuration`.

### AWS CLI

Para ver cómo se configura Security Hub para una organización

El siguiente `describe-organization-configuration` ejemplo devuelve información sobre la forma en que se configura una organización en Security Hub. En este ejemplo, la organización utiliza la configuración central. Solo la cuenta de administrador de Security Hub puede ejecutar este comando.

```
aws securityhub describe-organization-configuration
```

Salida:

```
{
  "AutoEnable": false,
  "MemberAccountLimitReached": false,
  "AutoEnableStandards": "NONE",
  "OrganizationConfiguration": {
    "ConfigurationType": "LOCAL",
    "Status": "ENABLED",
    "StatusMessage": "Central configuration has been enabled successfully"
  }
}
```

Para obtener más información, consulte [Administrar cuentas con AWS Organizations](#) en la Guía del usuario de AWS Security Hub.



- Para API obtener más información, consulte [DescribeOrganizationConfiguration](#) la Referencia de AWS CLI comandos.

## describe-products

En el siguiente ejemplo de código, se muestra cómo usar describe-products.

### AWS CLI

Para devolver información sobre las integraciones de productos disponibles

En el siguiente describe-products ejemplo, se muestran las integraciones de productos disponibles de una en una.

```
aws securityhub describe-products \
  --max-results 1
```

Salida:

```
{
  "NextToken": "U2FsdGVkX18vvP10qb7RDrWRWVFBJI46M0IAb+nZmRjM15NoRi2gm13sdQEn30/
  pq/78dGs+bKpgA+7HMPH00qX33/zoRI+uIG/F9yLNhc0r0WzFUdy36JcXLQji3Rpnn/
  cD1SVkGA98qI3zPOSDg==",
  "Products": [
    {
      "ProductArn": "arn:aws:securityhub:us-west-1:123456789333:product/
      crowdstrike/crowdstrike-falcon",
      "ProductName": "CrowdStrike Falcon",
      "CompanyName": "CrowdStrike",
      "Description": "CrowdStrike Falcon's single lightweight sensor unifies
      next-gen antivirus, endpoint detection and response, and 24/7 managed hunting, via
      the cloud.",
      "Categories": [
        "Endpoint Detection and Response (EDR)",
        "AV Scanning and Sandboxing",
        "Threat Intelligence Feeds and Reports",
        "Endpoint Forensics",
        "Network Forensics"
      ],
      "IntegrationTypes": [
        "SEND_FINDINGS_TO_SECURITY_HUB"
      ],
    }
  ],
}
```

```

    "MarketplaceUrl": "https://aws.amazon.com/marketplace/seller-profile?
id=a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "ActivationUrl": "https://falcon.crowdstrike.com/support/documentation",
    "ProductSubscriptionResourcePolicy": "{\"Version\":
\\\"2012-10-17\\\", \\\"Statement\\\": [{\\\"Effect\\\": \\\"Allow\\\", \\\"Principal\\\": {\\\"AWS\\\":
\\\"123456789333\\\"}, \\\"Action\\\": [\\\"securityhub:BatchImportFindings\\\"], \\\"Resource\\\":
\\\"arn:aws:securityhub:us-west-1:123456789012:product-subscription/crowdstrike/
crowdstrike-falcon\\\", \\\"Condition\\\": {\\\"StringEquals\\\": {\\\"securityhub:TargetAccount
\\\": \\\"123456789012\\\"}}}, {\\\"Effect\\\": \\\"Allow\\\", \\\"Principal\\\": {\\\"AWS\\\":
\\\"123456789012\\\"}, \\\"Action\\\": [\\\"securityhub:BatchImportFindings\\\"], \\\"Resource
\\\": \\\"arn:aws:securityhub:us-west-1:123456789333:product/crowdstrike/crowdstrike-
falcon\\\", \\\"Condition\\\": {\\\"StringEquals\\\": {\\\"securityhub:TargetAccount\\\":
\\\"123456789012\\\"}}}}]}\"
  }
]
}

```

Para obtener más información, consulte [Gestión de integraciones de productos](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [DescribeProducts](#) la Referencia de AWS CLI comandos.

## describe-standards-controls

En el siguiente ejemplo de código, se muestra cómo usar `describe-standards-controls`.

### AWS CLI

Para solicitar la lista de controles de un estándar habilitado

En el siguiente `describe-standards-controls` ejemplo, se solicita la lista de controles de la suscripción de la cuenta solicitante al PCI DSS estándar. La solicitud devuelve dos controles a la vez.

```

aws securityhub describe-standards-controls \
  --standards-subscription-arn "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1" \
  --max-results 2

```

Salida:

```

{
  "Controls": [
    {
      "StandardsControlArn": "arn:aws:securityhub:us-
west-1:123456789012:control/pci-dss/v/3.2.1/PCI.AutoScaling.1",
      "ControlStatus": "ENABLED",
      "ControlStatusUpdatedAt": "2020-05-15T18:49:04.473000+00:00",
      "ControlId": "PCI.AutoScaling.1",
      "Title": "Auto scaling groups associated with a load balancer should use
health checks",
      "Description": "This AWS control checks whether your Auto Scaling groups
that are associated with a load balancer are using Elastic Load Balancing health
checks.",
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
PCI.AutoScaling.1/remediation",
      "SeverityRating": "LOW",
      "RelatedRequirements": [
        "PCI DSS 2.2"
      ]
    },
    {
      "StandardsControlArn": "arn:aws:securityhub:us-
west-1:123456789012:control/pci-dss/v/3.2.1/PCI.CW.1",
      "ControlStatus": "ENABLED",
      "ControlStatusUpdatedAt": "2020-05-15T18:49:04.498000+00:00",
      "ControlId": "PCI.CW.1",
      "Title": "A log metric filter and alarm should exist for usage of the
\"root\" user",
      "Description": "This control checks for the CloudWatch metric
filters using the following pattern { $.userIdentity.type = \"Root\" &&
$.userIdentity.invokedBy NOT EXISTS && $.eventType != \"AwsServiceEvent\" }
It checks that the log group name is configured for use with active multi-
region CloudTrail, that there is at least one Event Selector for a Trail with
IncludeManagementEvents set to true and ReadWriteType set to All, and that there is
at least one active subscriber to an SNS topic associated with the alarm.",
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
PCI.CW.1/remediation",
      "SeverityRating": "MEDIUM",
      "RelatedRequirements": [
        "PCI DSS 7.2.1"
      ]
    }
  ],
}

```

```

    "NextToken": "U2FsdGVkX1+eNkPoZHV111ip5HUYQPWSWZGmftcmJiHL8JoKEsCDuaKayiPDyLK
+LiTkShveo0dvfxXCK0BaGhohIXhsIedN+LSjQV/
17kfCfJcq4PziNC1N9xe9aq2pjlLVZnznTfSImrodT5bRNHe4fELCQq/z+5ka
+5Lzmc11axcwTd5lKgQyQqmUVoeriHZhyIiBgWKf7oNYdBVG80EortVWvSkoUTt
+B2ThcnC7143kI0UNx1kZ6sc64AsW"
}

```

Para obtener más información, consulte [Visualización de los detalles de los controles](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [DescribeStandardsControls](#) la Referencia de AWS CLI comandos.

## describe-standards

En el siguiente ejemplo de código, se muestra cómo usar `describe-standards`.

### AWS CLI

Para devolver una lista de los estándares disponibles

El siguiente `describe-standards` ejemplo devuelve la lista de normas disponibles.

```
aws securityhub describe-standards
```

Salida:

```

{
  "Standards": [
    {
      "StandardsArn": "arn:aws:securityhub:us-west-1::standards/aws-
foundational-security-best-practices/v/1.0.0",
      "Name": "AWS Foundational Security Best Practices v1.0.0",
      "Description": "The AWS Foundational Security Best Practices standard
is a set of automated security checks that detect when AWS accounts and deployed
resources do not align to security best practices. The standard is defined by AWS
security experts. This curated set of controls helps improve your security posture
in AWS, and cover AWS's most popular and foundational services.",
      "EnabledByDefault": true
    },
    {
      "StandardsArn": "arn:aws:securityhub:::ruleset/cis-aws-foundations-
benchmark/v/1.2.0",

```

```

        "Name": "CIS AWS Foundations Benchmark v1.2.0",
        "Description": "The Center for Internet Security (CIS) AWS Foundations
Benchmark v1.2.0 is a set of security configuration best practices for AWS. This
Security Hub standard automatically checks for your compliance readiness against a
subset of CIS requirements.",
        "EnabledByDefault": true
    },
    {
        "StandardsArn": "arn:aws:securityhub:us-west-1::standards/pci-dss/
v/3.2.1",
        "Name": "PCI DSS v3.2.1",
        "Description": "The Payment Card Industry Data Security Standard (PCI
DSS) v3.2.1 is an information security standard for entities that store, process,
and/or transmit cardholder data. This Security Hub standard automatically checks
for your compliance readiness against a subset of PCI DSS requirements.",
        "EnabledByDefault": false
    }
]
}

```

Para obtener más información, consulte [los estándares de AWS seguridad de Security Hub](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [DescribeStandards](#) la Referencia de AWS CLI comandos.

## **disable-import-findings-for-product**

En el siguiente ejemplo de código, se muestra cómo usar `disable-import-findings-for-product`.

### AWS CLI

Para dejar de recibir los resultados de la integración de un producto

El siguiente `disable-import-findings-for-product` ejemplo desactiva el flujo de resultados para la suscripción especificada a una integración de productos.

```

aws securityhub disable-import-findings-for-product \
  --product-subscription-arn "arn:aws:securityhub:us-west-1:123456789012:product-
subscription/crowdstrike/crowdstrike-falcon"

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión de integraciones de productos](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [DisableImportFindingsForProduct](#) la Referencia de AWS CLI comandos.

## **disable-organization-admin-account**

En el siguiente ejemplo de código, se muestra cómo usar `disable-organization-admin-account`.

### AWS CLI

Para eliminar una cuenta de administrador de Security Hub

El siguiente `disable-organization-admin-account` ejemplo revoca la asignación de la cuenta especificada como cuenta de administrador de Security Hub para AWS Organizations.

```
aws securityhub disable-organization-admin-account \  
  --admin-account-id 777788889999
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Designación de una cuenta de administrador de Security Hub](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [DisableOrganizationAdminAccount](#) la Referencia de AWS CLI comandos.

## **disable-security-hub**

En el siguiente ejemplo de código, se muestra cómo usar `disable-security-hub`.

### AWS CLI

Para deshabilitar AWS Security Hub

En el siguiente `disable-security-hub` ejemplo, se deshabilita AWS Security Hub para la cuenta solicitante.

```
aws securityhub disable-security-hub
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Desactivación del AWS Security Hub](#) en la Guía del usuario del AWS Security Hub.

- Para API obtener más información, consulte [DisableSecurityHub](#) la Referencia de AWS CLI comandos.

## **disassociate-from-administrator-account**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-from-administrator-account`.

AWS CLI

Para desasociarse de una cuenta de administrador

El siguiente `disassociate-from-administrator-account` ejemplo desvincula la cuenta solicitante de su cuenta de administrador actual.

```
aws securityhub disassociate-from-administrator-account
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [DisassociateFromAdministratorAccount](#) la Referencia de AWS CLI comandos.

## **disassociate-from-master-account**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-from-master-account`.

AWS CLI

Para desasociarse de una cuenta de administrador

El siguiente `disassociate-from-master-account` ejemplo desvincula la cuenta solicitante de su cuenta de administrador actual.

```
aws securityhub disassociate-from-master-account
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [DisassociateFromMasterAccount](#) la Referencia de AWS CLI comandos.

## **disassociate-members**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-members`.

### AWS CLI

Desasociación de cuentas de miembros

El siguiente `disassociate-members` ejemplo desasocia las cuentas de los miembros especificadas de la cuenta de administrador solicitante.

```
aws securityhub disassociate-members \  
  --account-ids "123456789111" "123456789222"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [DisassociateMembers](#) la Referencia de AWS CLI comandos.

## **enable-import-findings-for-product**

En el siguiente ejemplo de código, se muestra cómo usar `enable-import-findings-for-product`.



## AWS CLI

Para empezar a recibir los resultados de la integración de un producto

El siguiente `enable-import-findings-for-product` ejemplo permite el flujo de los resultados de la integración de productos especificada.

```
aws securityhub enable-import-findings-for-product \  
  --product-arn "arn:aws:securityhub:us-east-1:123456789333:product/crowdstrike/  
crowdstrike-falcon"
```

Salida:

```
{  
  "ProductSubscriptionArn": "arn:aws:securityhub:us-east-1:123456789012:product-  
subscription/crowdstrike/crowdstrike-falcon"  
}
```

Para obtener más información, consulte [Gestión de integraciones de productos](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [EnableImportFindingsForProduct](#) la Referencia de AWS CLI comandos.

## `enable-organization-admin-account`

En el siguiente ejemplo de código, se muestra cómo usar `enable-organization-admin-account`.

## AWS CLI

Para designar una cuenta de organización como cuenta de administrador de Security Hub

En el siguiente `enable-organization-admin-account` ejemplo, se designa la cuenta especificada como cuenta de administrador de Security Hub.

```
aws securityhub enable-organization-admin-account \  
  --admin-account-id 777788889999
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Designación de una cuenta de administrador de Security Hub](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [EnableOrganizationAdminAccount](#) la Referencia de AWS CLI comandos.

## **enable-security-hub**

En el siguiente ejemplo de código, se muestra cómo usar `enable-security-hub`.

### AWS CLI

Para habilitar AWS Security Hub

El siguiente `enable-security-hub` ejemplo habilita AWS Security Hub para la cuenta solicitante. Configura Security Hub para habilitar los estándares predeterminados. En el caso del recurso `hub`, asigna el valor `Security` a la etiqueta. `Department`

```
aws securityhub enable-security-hub \  
  --enable-default-standards \  
  --tags '{"Department": "Security"}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Habilitar Security Hub](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [EnableSecurityHub](#) la Referencia de AWS CLI comandos.

## **get-administrator-account**

En el siguiente ejemplo de código, se muestra cómo usar `get-administrator-account`.

### AWS CLI

Para recuperar información sobre una cuenta de administrador

En el siguiente `get-administrator-account` ejemplo, se recupera información sobre la cuenta de administrador de la cuenta solicitante.

```
aws securityhub get-administrator-account
```

Salida:

```
{
  "Master": {
    "AccountId": "123456789012",
    "InvitationId": "7ab938c5d52d7904ad09f9e7c20cc4eb",
    "InvitedAt": 2020-06-01T20:21:18.042000+00:00,
    "MemberStatus": "ASSOCIATED"
  }
}
```

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [GetAdministratorAccount](#) la Referencia de AWS CLI comandos.

## get-configuration-policy-association

En el siguiente ejemplo de código, se muestra cómo usar `get-configuration-policy-association`.

### AWS CLI

Para obtener los detalles de la asociación de configuración de un objetivo

En el siguiente `get-configuration-policy-association` ejemplo, se recuperan los detalles de la asociación del objetivo especificado. Puedes proporcionar un identificador de cuenta, un identificador de unidad organizativa o el identificador raíz del objetivo.

```
aws securityhub get-configuration-policy-association \
  --target '{"OrganizationalUnitId": "ou-6hi7-8j91k12m"}'
```

Salida:

```
{
  "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
  "TargetId": "ou-6hi7-8j91k12m",
  "TargetType": "ORGANIZATIONAL_UNIT",
  "AssociationType": "APPLIED",
  "UpdatedAt": "2023-09-26T21:13:01.816000+00:00",
  "AssociationStatus": "SUCCESS",
}
```

```
"AssociationStatusMessage": "Association applied successfully on this target."
}
```

Para obtener más información, consulte [Visualización de las políticas de configuración de Security Hub](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [GetConfigurationPolicyAssociation](#) la Referencia de AWS CLI comandos.

## get-configuration-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-configuration-policy`.

### AWS CLI

Para ver los detalles de la política de configuración

El siguiente `get-configuration-policy` ejemplo recupera detalles sobre la política de configuración especificada.

```
aws securityhub get-configuration-policy \
  --identifier "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Salida:

```
{
  "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Id": "ce5ed1e7-9639-4e2f-9313-fa87fcef944b",
  "Name": "SampleConfigurationPolicy",
  "Description": "SampleDescription",
  "UpdatedAt": "2023-11-28T20:28:04.494000+00:00",
  "CreatedAt": "2023-11-28T20:28:04.494000+00:00",
  "ConfigurationPolicy": {
    "SecurityHub": {
      "ServiceEnabled": true,
      "EnabledStandardIdentifiers": [
        "arn:aws:securityhub:eu-central-1::standards/aws-foundational-
security-best-practices/v/1.0.0",
        "arn:aws:securityhub::ruleset/cis-aws-foundations-benchmark/
v/1.2.0"
      ],
    },
  },
}
```

```

    "SecurityControlsConfiguration": {
      "DisabledSecurityControlIdentifiers": [
        "CloudTrail.2"
      ],
      "SecurityControlCustomParameters": [
        {
          "SecurityControlId": "ACM.1",
          "Parameters": {
            "daysToExpiration": {
              "ValueType": "CUSTOM",
              "Value": {
                "Integer": 15
              }
            }
          }
        }
      ]
    }
  }
}

```

Para obtener más información, consulte [Visualización de las políticas de configuración de Security Hub](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [GetConfigurationPolicy](#) la Referencia de AWS CLI comandos.

## get-enabled-standards

En el siguiente ejemplo de código, se muestra cómo usar `get-enabled-standards`.

### AWS CLI

Para recuperar información sobre un estándar habilitado

En el siguiente `get-enabled-standards` ejemplo, se recupera información sobre el PCI DSS estándar.

```

aws securityhub get-enabled-standards \
  --standards-subscription-arn "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1"

```

Salida:

```
{
  "StandardsSubscriptions": [
    {
      "StandardsArn": "arn:aws:securityhub:us-west-1::standards/pci-dss/v/3.2.1",
      "StandardsInput": { },
      "StandardsStatus": "READY",
      "StandardsSubscriptionArn": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/v/3.2.1"
    }
  ]
}
```

Para obtener más información, consulte [los estándares de AWS seguridad de Security Hub](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [GetEnabledStandards](#) la Referencia de AWS CLI comandos.

## get-finding-aggregator

En el siguiente ejemplo de código, se muestra cómo usar `get-finding-aggregator`.

AWS CLI

Para recuperar la configuración de agregación de búsquedas actual

En el siguiente `get-finding-aggregator` ejemplo, se recupera la configuración de agregación de búsquedas actual.

```
aws securityhub get-finding-aggregator \
  --finding-aggregator-arn arn:aws:securityhub:us-east-1:222222222222:finding-aggregator/123e4567-e89b-12d3-a456-426652340000
```

Salida:

```
{
  "FindingAggregatorArn": "arn:aws:securityhub:us-east-1:222222222222:finding-aggregator/123e4567-e89b-12d3-a456-426652340000",
  "FindingAggregationRegion": "us-east-1",
}
```

```
"RegionLinkingMode": "SPECIFIED_REGIONS",  
"Regions": "us-west-1,us-west-2"  
}
```

Para obtener más información, consulte [Visualización de la configuración de agregación de búsquedas actual](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [GetFindingAggregator](#) la Referencia de AWS CLI comandos.

## get-finding-history

En el siguiente ejemplo de código, se muestra cómo usar `get-finding-history`.

### AWS CLI

Para obtener el historial de búsquedas

En el siguiente `get-finding-history` ejemplo, se actualiza el historial de los últimos 90 días del hallazgo especificado. En este ejemplo, los resultados se limitan a dos registros del historial de búsquedas.

```
aws securityhub get-finding-history \  
  --finding-identifier Id="arn:aws:securityhub:us-  
east-1:123456789012:security-control/S3.17/finding/a1b2c3d4-5678-90ab-cdef-  
EXAMPLE11111",ProductArn="arn:aws:securityhub:us-east-1::product/aws/securityhub"
```

Salida:

```
{  
  "Records": [  
    {  
      "FindingIdentifier": {  
        "Id": "arn:aws:securityhub:us-east-1:123456789012:security-control/  
S3.17/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
        "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/  
securityhub"  
      },  
      "UpdateTime": "2023-06-02T03:15:25.685000+00:00",  
      "FindingCreated": false,  
      "UpdateSource": {  
        "Type": "BATCH_IMPORT_FINDINGS",
```

```

    "Identity": "arn:aws:securityhub:us-east-1::product/aws/securityhub"
  },
  "Updates": [
    {
      "UpdatedField": "Compliance.RelatedRequirements",
      "OldValue": "[\"NIST.800-53.r5 SC-12(2)\",\"NIST.800-53.r5
SC-12(3)\",\"NIST.800-53.r5 SC-12(6)\",\"NIST.800-53.r5 CM-3(6)\",\"NIST.800-53.r5
SC-13\", \"NIST.800-53.r5 SC-28\", \"NIST.800-53.r5 SC-28(1)\", \"NIST.800-53.r5
SC-7(10)\"]",
      "NewValue": "[\"NIST.800-53.r5 SC-12(2)\",\"NIST.800-53.r5
CM-3(6)\",\"NIST.800-53.r5 SC-13\", \"NIST.800-53.r5 SC-28\", \"NIST.800-53.r5
SC-28(1)\", \"NIST.800-53.r5 SC-7(10)\", \"NIST.800-53.r5 CA-9(1)\", \"NIST.800-53.r5
SI-7(6)\", \"NIST.800-53.r5 AU-9\"]"
    },
    {
      "UpdatedField": "LastObservedAt",
      "OldValue": "2023-06-01T09:15:38.587Z",
      "NewValue": "2023-06-02T03:15:22.946Z"
    },
    {
      "UpdatedField": "UpdatedAt",
      "OldValue": "2023-06-01T09:15:31.049Z",
      "NewValue": "2023-06-02T03:15:14.861Z"
    },
    {
      "UpdatedField": "ProcessedAt",
      "OldValue": "2023-06-01T09:15:41.058Z",
      "NewValue": "2023-06-02T03:15:25.685Z"
    }
  ]
},
{
  "FindingIdentifier": {
    "Id": "arn:aws:securityhub:us-east-1:123456789012:security-control/
S3.17/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/
securityhub"
  },
  "UpdateTime": "2023-05-23T02:06:51.518000+00:00",
  "FindingCreated": "true",
  "UpdateSource": {
    "Type": "BATCH_IMPORT_FINDINGS",
    "Identity": "arn:aws:securityhub:us-east-1::product/aws/securityhub"
  }
},

```



```

    "Updates": []
  }
]
}

```

Para obtener más información, consulte [Búsqueda del historial](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [GetFindingHistory](#) la Referencia de AWS CLI comandos.

## get-findings

En el siguiente ejemplo de código, se muestra cómo usar `get-findings`.

### AWS CLI

Ejemplo 1: Para devolver los hallazgos generados para un estándar específico

El siguiente `get-findings` ejemplo devuelve los resultados de la PCI DSS norma.

```

aws securityhub get-findings \
  --filters '{"GeneratorId":[{"Value": "pci-dss", "Comparison": "PREFIX"}]}' \
  --max-items 1

```

Salida:

```

{
  "Findings": [
    {
      "SchemaVersion": "2018-10-08",
      "Id": "arn:aws:securityhub:eu-central-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub",
      "GeneratorId": "pci-dss/v/3.2.1/PCI.Lambda.2",
      "AwsAccountId": "123456789012",
      "Types": [
        "Software and Configuration Checks/Industry and Regulatory Standards/PCI-DSS"
      ],
      "FindingProviderFields": {
        "Severity": {
          "Original": 0,

```

```
        "Label": "INFORMATIONAL"
      },
      "Types": [
        "Software and Configuration Checks/Industry and Regulatory
Standards/PCI-DSS"
      ]
    },
    "FirstObservedAt": "2020-06-02T14:02:49.159Z",
    "LastObservedAt": "2020-06-02T14:02:52.397Z",
    "CreatedAt": "2020-06-02T14:02:49.159Z",
    "UpdatedAt": "2020-06-02T14:02:52.397Z",
    "Severity": {
      "Original": 0,
      "Label": "INFORMATIONAL",
      "Normalized": 0
    },
    "Title": "PCI.Lambda.2 Lambda functions should be in a VPC",
    "Description": "This AWS control checks whether a Lambda function is in
a VPC.",
    "Remediation": {
      "Recommendation": {
        "Text": "For directions on how to fix this issue, please consult
the AWS Security Hub PCI DSS documentation.",
        "Url": "https://docs.aws.amazon.com/console/securityhub/
PCI.Lambda.2/remediation"
      }
    },
    "ProductFields": {
      "StandardsArn": "arn:aws:securityhub::standards/pci-dss/v/3.2.1",
      "StandardsSubscriptionArn": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1",
      "ControlId": "PCI.Lambda.2",
      "RecommendationUrl": "https://docs.aws.amazon.com/console/
securityhub/PCI.Lambda.2/remediation",
      "RelatedAWSResources:0/name": "securityhub-lambda-inside-
vpc-0e904a3b",
      "RelatedAWSResources:0/type": "AWS::Config::ConfigRule",
      "StandardsControlArn": "arn:aws:securityhub:us-
west-1:123456789012:control/pci-dss/v/3.2.1/PCI.Lambda.2",
      "aws/securityhub/SeverityLabel": "INFORMATIONAL",
      "aws/securityhub/ProductName": "Security Hub",
      "aws/securityhub/CompanyName": "AWS",
      "aws/securityhub/FindingId": "arn:aws:securityhub:eu-
central-1::product/aws/securityhub/arn:aws:securityhub:eu-
```

```

central-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    },
    "Resources": [
      {
        "Type": "AwsAccount",
        "Id": "AWS:::Account:123456789012",
        "Partition": "aws",
        "Region": "us-west-1"
      }
    ],
    "Compliance": {
      "Status": "PASSED",
      "RelatedRequirements": [
        "PCI DSS 1.2.1",
        "PCI DSS 1.3.1",
        "PCI DSS 1.3.2",
        "PCI DSS 1.3.4"
      ]
    },
    "WorkflowState": "NEW",
    "Workflow": {
      "Status": "NEW"
    },
    "RecordState": "ARCHIVED"
  }
],
"NextToken": "eyJ0ZXh0VG9rZW4iOiB1b250b190cnVuY2F0ZV9hbW91bnQiOiAxfQ=="
}

```

Ejemplo 2: Para devolver hallazgos de gravedad crítica con un estado de flujo de trabajo de NOTIFIED

En el siguiente `get-findings` ejemplo, se muestran los resultados que tienen un valor de etiqueta de gravedad de CRITICAL y un estado de flujo de trabajo de. NOTIFIED Los resultados se ordenan en orden descendente según el valor de Confidence.

```

aws securityhub get-findings \
  --filters '{"SeverityLabel":[{"Value":
"CRITICAL", "Comparison": "EQUALS"}], "WorkflowStatus":
[{"Value": "NOTIFIED", "Comparison": "EQUALS"}]}' \
  --sort-criteria '{"Field": "Confidence", "SortOrder": "desc"}' \

```

```
--max-items 1
```

Salida:

```
{
  "Findings": [
    {
      "SchemaVersion": "2018-10-08",
      "Id": "arn:aws:securityhub:us-west-1: 123456789012:subscription/cis-aws-
foundations-benchmark/v/1.2.0/1.13/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ProductArn": "arn:aws:securityhub:us-west-2::product/aws/securityhub",
      "GeneratorId": "arn:aws:securityhub:::ruleset/cis-aws-foundations-
benchmark/v/1.2.0/rule/1.13",
      "AwsAccountId": "123456789012",
      "Types": [
        "Software and Configuration Checks/Industry and Regulatory
Standards/CIS AWS Foundations Benchmark"
      ],
      "FindingProviderFields" {
        "Severity": {
          "Original": 90,
          "Label": "CRITICAL"
        },
        "Types": [
          "Software and Configuration Checks/Industry and Regulatory
Standards/CIS AWS Foundations Benchmark"
        ]
      },
      "FirstObservedAt": "2020-05-21T20:16:34.752Z",
      "LastObservedAt": "2020-06-09T08:16:37.171Z",
      "CreatedAt": "2020-05-21T20:16:34.752Z",
      "UpdatedAt": "2020-06-09T08:16:36.430Z",
      "Severity": {
        "Original": 90,
        "Label": "CRITICAL",
        "Normalized": 90
      },
      "Title": "1.13 Ensure MFA is enabled for the \"root\" account",
      "Description": "The root account is the most privileged user in an AWS
account. MFA adds an extra layer of protection on top of a user name and password.
With MFA enabled, when a user signs in to an AWS website, they will be prompted for
their user name and password as well as for an authentication code from their AWS
MFA device.",
    }
  ]
}
```

```
    "Remediation": {
      "Recommendation": {
        "Text": "For directions on how to fix this issue, please consult
the AWS Security Hub CIS documentation.",
        "Url": "https://docs.aws.amazon.com/console/securityhub/
standards-cis-1.13/remediation"
      }
    },
    "ProductFields": {
      "StandardsGuideArn": "arn:aws:securityhub::ruleset/cis-aws-
foundations-benchmark/v/1.2.0",
      "StandardsGuideSubscriptionArn": "arn:aws:securityhub:us-
west-1:123456789012:subscription/cis-aws-foundations-benchmark/v/1.2.0",
      "RuleId": "1.13",
      "RecommendationUrl": "https://docs.aws.amazon.com/console/
securityhub/standards-cis-1.13/remediation",
      "RelatedAWSResources:0/name": "securityhub-root-account-mfa-
enabled-5pftha",
      "RelatedAWSResources:0/type": "AWS::Config::ConfigRule",
      "StandardsControlArn": "arn:aws:securityhub:us-
west-1:123456789012:control/cis-aws-foundations-benchmark/v/1.2.0/1.13",
      "aws/securityhub/SeverityLabel": "CRITICAL",
      "aws/securityhub/ProductName": "Security Hub",
      "aws/securityhub/CompanyName": "AWS",
      "aws/securityhub/FindingId": "arn:aws:securityhub:us-
west-1::product/aws/securityhub/arn:aws:securityhub:us-
west-1:123456789012:subscription/cis-aws-foundations-benchmark/v/1.2.0/1.13/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    },
    "Resources": [
      {
        "Type": "AwsAccount",
        "Id": "AWS:::Account:123456789012",
        "Partition": "aws",
        "Region": "us-west-1"
      }
    ],
    "Compliance": {
      "Status": "FAILED"
    },
    "WorkflowState": "NEW",
    "Workflow": {
      "Status": "NOTIFIED"
    }
  },
```

```

        "RecordState": "ACTIVE"
      }
    ]
  }

```

Para obtener más información, consulte los [resultados de filtrado y agrupamiento](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [GetFindings](#) la Referencia de AWS CLI comandos.

## get-insight-results

En el siguiente ejemplo de código, se muestra cómo usar `get-insight-results`.

### AWS CLI

Para recuperar los resultados y obtener información

En el siguiente `get-insight-results` ejemplo, se devuelve la lista de resultados de la información con la información especificadaARN.

```

aws securityhub get-insight-results \
  --insight-arn "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/
  custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"

```

Salida:

```

{
  "InsightResults": {
    "GroupByAttribute": "ResourceId",
    "InsightArn": "arn:aws:securityhub:us-
west-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111",
    "ResultValues": [
      {
        "Count": 10,
        "GroupByAttributeValue": "AWS:::Account:123456789111"
      },
      {
        "Count": 3,
        "GroupByAttributeValue": "AWS:::Account:123456789222"
      }
    ]
  }
}

```

```
}
}
```

Para obtener más información, consulte [Visualización de los resultados y hallazgos de insights y tomar medidas](#) al respecto en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [GetInsightResults](#) la Referencia de AWS CLI comandos.

## get-insights

En el siguiente ejemplo de código, se muestra cómo usar `get-insights`.

### AWS CLI

Para recuperar detalles sobre una información

En el siguiente `get-insights` ejemplo, se recuperan los detalles de configuración de la información con lo especificadoARN.

```
aws securityhub get-insights \
  --insight-arns "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/
  custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Salida:

```
{
  "Insights": [
    {
      "Filters": {
        "ResourceType": [
          {
            "Comparison": "EQUALS",
            "Value": "AwsIamRole"
          }
        ],
        "SeverityLabel": [
          {
            "Comparison": "EQUALS",
            "Value": "CRITICAL"
          }
        ]
      }
    }
  ],
}
```

```
        "GroupByAttribute": "ResourceId",
        "InsightArn": "arn:aws:securityhub:us-
west-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111",
        "Name": "Critical role findings"
    }
]
}
```

Para obtener más información, consulte [Insights in AWS Security Hub](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [GetInsights](#) la Referencia de AWS CLI comandos.

## get-invitations-count

En el siguiente ejemplo de código, se muestra cómo usar `get-invitations-count`.

### AWS CLI

Para recuperar el número de invitaciones que no se aceptaron

En el siguiente `get-invitations-count` ejemplo, se recupera el número de invitaciones que la cuenta solicitante rechazó o a las que no respondió.

```
aws securityhub get-invitations-count
```

Salida:

```
{
  "InvitationsCount": 3
}
```

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [GetInvitationsCount](#) la Referencia de AWS CLI comandos.

## get-master-account

En el siguiente ejemplo de código, se muestra cómo usar `get-master-account`.



## AWS CLI

Para recuperar información sobre una cuenta de administrador

En el siguiente `get-master-account` ejemplo, se recupera información sobre la cuenta de administrador de la cuenta solicitante.

```
aws securityhub get-master-account
```

Salida:

```
{
  "Master": {
    "AccountId": "123456789012",
    "InvitationId": "7ab938c5d52d7904ad09f9e7c20cc4eb",
    "InvitedAt": 2020-06-01T20:21:18.042000+00:00,
    "MemberStatus": "ASSOCIATED"
  }
}
```

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [GetMasterAccount](#) la Referencia de AWS CLI comandos.

## get-members

En el siguiente ejemplo de código, se muestra cómo usar `get-members`.

## AWS CLI

Para recuperar información sobre las cuentas de los miembros seleccionados

En el siguiente `get-members` ejemplo, se recupera información sobre las cuentas de los miembros especificadas.

```
aws securityhub get-members \
  --account-ids "444455556666" "777788889999"
```

Salida:

```
{
  "Members": [
    {
      "AccountId": "123456789111",
      "AdministratorId": "123456789012",
      "InvitedAt": 2020-06-01T20:15:15.289000+00:00,
      "MasterId": "123456789012",
      "MemberStatus": "ASSOCIATED",
      "UpdatedAt": 2020-06-01T20:15:15.289000+00:00
    },
    {
      "AccountId": "123456789222",
      "AdministratorId": "123456789012",
      "InvitedAt": 2020-06-01T20:15:15.289000+00:00,
      "MasterId": "123456789012",
      "MemberStatus": "ASSOCIATED",
      "UpdatedAt": 2020-06-01T20:15:15.289000+00:00
    }
  ],
  "UnprocessedAccounts": [ ]
}
```

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [GetMembers](#) la Referencia de AWS CLI comandos.

## get-security-control-definition

En el siguiente ejemplo de código, se muestra cómo usar `get-security-control-definition`.

### AWS CLI

Para obtener los detalles de la definición del control de seguridad

En el siguiente `get-security-control-definition` ejemplo, se recuperan los detalles de la definición de un control de seguridad de Security Hub. Los detalles incluyen el título del control, la descripción, la disponibilidad regional, los parámetros y otra información.

```
aws securityhub get-security-control-definition \
  --security-control-id ACM.1
```

**Salida:**

```
{
  "SecurityControlDefinition": {
    "SecurityControlId": "ACM.1",
    "Title": "Imported and ACM-issued certificates should be renewed after a
specified time period",
    "Description": "This control checks whether an AWS Certificate Manager
(ACM) certificate is renewed within the specified time period. It checks both
imported certificates and certificates provided by ACM. The control fails if the
certificate isn't renewed within the specified time period. Unless you provide a
custom parameter value for the renewal period, Security Hub uses a default value of
30 days.",
    "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/ACM.1/
remediation",
    "SeverityRating": "MEDIUM",
    "CurrentRegionAvailability": "AVAILABLE",
    "ParameterDefinitions": {
      "daysToExpiration": {
        "Description": "Number of days within which the ACM certificate must
be renewed",
        "ConfigurationOptions": {
          "Integer": {
            "DefaultValue": 30,
            "Min": 14,
            "Max": 365
          }
        }
      }
    }
  }
}
```

Para obtener más información, consulte [Parámetros de control personalizados](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [GetSecurityControlDefinition](#) la Referencia de AWS CLI comandos.

**invite-members**

En el siguiente ejemplo de código, se muestra cómo usar `invite-members`.

## AWS CLI

Para enviar invitaciones a las cuentas de los miembros

En el siguiente `invite-members` ejemplo, se envían invitaciones a las cuentas de los miembros especificadas.

```
aws securityhub invite-members \  
  --account-ids "123456789111" "123456789222"
```

Salida:

```
{  
  "UnprocessedAccounts": []  
}
```

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [InviteMembers](#) la Referencia de AWS CLI comandos.

## `list-automation-rules`

En el siguiente ejemplo de código, se muestra cómo usar `list-automation-rules`.

## AWS CLI

Para ver una lista de reglas de automatización

En el siguiente `list-automation-rules` ejemplo, se enumeran las reglas de automatización de una AWS cuenta. Solo la cuenta de administrador de Security Hub puede ejecutar este comando.

```
aws securityhub list-automation-rules \  
  --max-results 3 \  
  --next-token NULL
```

Salida:

```
{
```

```
"AutomationRulesMetadata": [
  {
    "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "RuleStatus": "ENABLED",
    "RuleOrder": 1,
    "RuleName": "Suppress informational findings",
    "Description": "Suppress GuardDuty findings with Informational
severity",
    "IsTerminal": false,
    "CreatedAt": "2023-05-31T17:56:14.837000+00:00",
    "UpdatedAt": "2023-05-31T17:59:38.466000+00:00",
    "CreatedBy": "arn:aws:iam::123456789012:role/Admin"
  },
  {
    "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "RuleStatus": "ENABLED",
    "RuleOrder": 1,
    "RuleName": "sample rule",
    "Description": "A sample rule",
    "IsTerminal": false,
    "CreatedAt": "2023-07-15T23:37:20.223000+00:00",
    "UpdatedAt": "2023-07-15T23:37:20.223000+00:00",
    "CreatedBy": "arn:aws:iam::123456789012:role/Admin"
  },
  {
    "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "RuleStatus": "ENABLED",
    "RuleOrder": 1,
    "RuleName": "sample rule",
    "Description": "A sample rule",
    "IsTerminal": false,
    "CreatedAt": "2023-07-15T23:45:25.126000+00:00",
    "UpdatedAt": "2023-07-15T23:45:25.126000+00:00",
    "CreatedBy": "arn:aws:iam::123456789012:role/Admin"
  }
]
}
```

Para obtener más información, consulte [Visualización de las reglas de automatización](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [ListAutomationRules](#) la Referencia de AWS CLI comandos.

## list-configuration-policies

En el siguiente ejemplo de código, se muestra cómo usar `list-configuration-policies`.

AWS CLI

Para enumerar los resúmenes de las políticas de configuración

El siguiente `list-configuration-policies` ejemplo muestra un resumen de las políticas de configuración de la organización.

```
aws securityhub list-configuration-policies \  
  --max-items 3
```

Salida:

```
{  
  "ConfigurationPolicySummaries": [  
    {  
      "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-  
policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "Name": "SampleConfigurationPolicy1",  
      "Description": "SampleDescription1",  
      "UpdatedAt": "2023-09-26T21:08:36.214000+00:00",  
      "ServiceEnabled": true  
    },  
    {  
      "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-  
policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "Name": "SampleConfigurationPolicy2",  
      "Description": "SampleDescription2"  
      "UpdatedAt": "2023-11-28T19:26:25.207000+00:00",  
      "ServiceEnabled": true  
    },  
    {  
      "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-  
policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
```

```

    "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "Name": "SampleConfigurationPolicy3",
    "Description": "SampleDescription3",
    "UpdatedAt": "2023-11-28T20:28:04.494000+00:00",
    "ServiceEnabled": true
  }
}

```

Para obtener más información, consulte [Visualización de las políticas de configuración de Security Hub](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [ListConfigurationPolicies](#) la Referencia de AWS CLI comandos.

## list-configuration-policy-associations

En el siguiente ejemplo de código, se muestra cómo usar `list-configuration-policy-associations`.

### AWS CLI

Para enumerar las asociaciones de configuración

El siguiente `list-configuration-policy-associations` ejemplo muestra un resumen de las asociaciones de configuración de la organización. La respuesta incluye asociaciones con las políticas de configuración y el comportamiento autogestionado.

```

aws securityhub list-configuration-policy-associations \
  --association-type "APPLIED" \
  --max-items 4

```

Salida:

```

{
  "ConfigurationPolicyAssociationSummaries": [
    {
      "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "TargetId": "r-1ab2",
      "TargetType": "ROOT",
      "AssociationType": "APPLIED",
      "UpdatedAt": "2023-11-28T19:26:49.417000+00:00",
      "AssociationStatus": "FAILED",
    }
  ]
}

```

```

    "AssociationStatusMessage": "Policy association failed because 2
organizational units or accounts under this root failed."
  },
  {
    "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "TargetId": "ou-1ab2-c3de4f5g",
    "TargetType": "ORGANIZATIONAL_UNIT",
    "AssociationType": "APPLIED",
    "UpdatedAt": "2023-09-26T21:14:05.283000+00:00",
    "AssociationStatus": "FAILED",
    "AssociationStatusMessage": "One or more children under this target
failed association."
  },
  {
    "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "TargetId": "ou-6hi7-8j9kl2m",
    "TargetType": "ORGANIZATIONAL_UNIT",
    "AssociationType": "APPLIED",
    "UpdatedAt": "2023-09-26T21:13:01.816000+00:00",
    "AssociationStatus": "SUCCESS",
    "AssociationStatusMessage": "Association applied successfully on this
target."
  },
  {
    "ConfigurationPolicyId": "SELF_MANAGED_SECURITY_HUB",
    "TargetId": "111122223333",
    "TargetType": "ACCOUNT",
    "AssociationType": "APPLIED",
    "UpdatedAt": "2023-11-28T22:01:26.409000+00:00",
    "AssociationStatus": "SUCCESS"
  }
}

```

Para obtener más información, consulte [Visualización de las políticas de configuración de Security Hub](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [ListConfigurationPolicyAssociations](#) la Referencia de AWS CLI comandos.

## list-enabled-products-for-import

En el siguiente ejemplo de código, se muestra cómo usar `list-enabled-products-for-import`.



## AWS CLI

Para devolver la lista de integraciones de productos habilitadas

En el siguiente `list-enabled-products-for-import` ejemplo, se devuelve la lista de suscripciones de ARNS las integraciones de productos actualmente habilitadas.

```
aws securityhub list-enabled-products-for-import
```

Salida:

```
{
  "ProductSubscriptions": [ "arn:aws:securityhub:us-west-1:123456789012:product-
subscription/crowdstrike/crowdstrike-falcon", "arn:aws:securityhub:us-
west-1:123456789012:product-subscription/aws/securityhub" ]
}
```

Para obtener más información, consulte [Gestión de integraciones de productos](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [ListEnabledProductsForImport](#) la Referencia de AWS CLI comandos.

## list-finding-aggregators

En el siguiente ejemplo de código, se muestra cómo usar `list-finding-aggregators`.

## AWS CLI

Para ver una lista de los widgets disponibles

El siguiente `list-finding-aggregators` ejemplo devuelve la configuración ARN de agregación de búsquedas.

```
aws securityhub list-finding-aggregators
```

Salida:

```
{
```

```
"FindingAggregatorArn": "arn:aws:securityhub:us-east-1:222222222222:finding-
aggregator/123e4567-e89b-12d3-a456-426652340000"
}
```

Para obtener más información, consulte [Visualización de la configuración de agregación de búsquedas actual](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [ListFindingAggregators](#) la Referencia de AWS CLI comandos.

## list-invitations

En el siguiente ejemplo de código, se muestra cómo usar `list-invitations`.

### AWS CLI

Para mostrar una lista de invitaciones

En el siguiente `list-invitations` ejemplo, se recupera la lista de invitaciones enviadas a la cuenta solicitante.

```
aws securityhub list-invitations
```

Salida:

```
{
  "Invitations": [
    {
      "AccountId": "123456789012",
      "InvitationId": "7ab938c5d52d7904ad09f9e7c20cc4eb",
      "InvitedAt": 2020-06-01T20:21:18.042000+00:00,
      "MemberStatus": "ASSOCIATED"
    }
  ],
}
```

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [ListInvitations](#) la Referencia de AWS CLI comandos.

## list-members

En el siguiente ejemplo de código, se muestra cómo usar `list-members`.

### AWS CLI

Para recuperar una lista de las cuentas de los miembros

En el siguiente `list-members` ejemplo, se devuelve la lista de cuentas de miembro de la cuenta de administrador solicitante.

```
aws securityhub list-members
```

Salida:

```
{
  "Members": [
    {
      "AccountId": "123456789111",
      "AdministratorId": "123456789012",
      "InvitedAt": 2020-06-01T20:15:15.289000+00:00,
      "MasterId": "123456789012",
      "MemberStatus": "ASSOCIATED",
      "UpdatedAt": 2020-06-01T20:15:15.289000+00:00
    },
    {
      "AccountId": "123456789222",
      "AdministratorId": "123456789012",
      "InvitedAt": 2020-06-01T20:15:15.289000+00:00,
      "MasterId": "123456789012",
      "MemberStatus": "ASSOCIATED",
      "UpdatedAt": 2020-06-01T20:15:15.289000+00:00
    }
  ],
}
```

Para obtener más información, consulte [Administrar cuentas de administrador y miembro](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [ListMembers](#) la Referencia de AWS CLI comandos.

## list-organization-admin-accounts

En el siguiente ejemplo de código, se muestra cómo usar `list-organization-admin-accounts`.

### AWS CLI

Para enumerar las cuentas de administrador de Security Hub designadas

En el siguiente `list-organization-admin-accounts` ejemplo, se enumeran las cuentas de administrador de Security Hub de una organización.

```
aws securityhub list-organization-admin-accounts
```

Salida:

```
{
  AdminAccounts": [
    { "AccountId": "777788889999" },
    { "Status": "ENABLED" }
  ]
}
```

Para obtener más información, consulte [Designación de una cuenta de administrador de Security Hub](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [ListOrganizationAdminAccounts](#) la Referencia de AWS CLI comandos.

## list-security-control-definitions

En el siguiente ejemplo de código, se muestra cómo usar `list-security-control-definitions`.

### AWS CLI

Ejemplo 1: Para enumerar todos los controles de seguridad disponibles

En el siguiente `list-security-control-definitions` ejemplo, se enumeran los controles de seguridad disponibles en todos los estándares de Security Hub. Este ejemplo limita los resultados a tres controles.

```
aws securityhub list-security-control-definitions \  
--max-items 3
```

Salida:

```
{  
  "SecurityControlDefinitions": [  
    {  
      "SecurityControlId": "ACM.1",  
      "Title": "Imported and ACM-issued certificates should be renewed after a  
specified time period",  
      "Description": "This control checks whether an AWS Certificate Manager  
(ACM) certificate is renewed within the specified time period. It checks both  
imported certificates and certificates provided by ACM. The control fails if the  
certificate isn't renewed within the specified time period. Unless you provide a  
custom parameter value for the renewal period, Security Hub uses a default value of  
30 days.",  
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/  
ACM.1/remediation",  
      "SeverityRating": "MEDIUM",  
      "CurrentRegionAvailability": "AVAILABLE",  
      "CustomizableProperties": [  
        "Parameters"  
      ]  
    },  
    {  
      "SecurityControlId": "ACM.2",  
      "Title": "RSA certificates managed by ACM should use a key length of at  
least 2,048 bits",  
      "Description": "This control checks whether RSA certificates managed by  
AWS Certificate Manager use a key length of at least 2,048 bits. The control fails  
if the key length is smaller than 2,048 bits.",  
      "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/  
ACM.2/remediation",  
      "SeverityRating": "HIGH",  
      "CurrentRegionAvailability": "AVAILABLE",  
      "CustomizableProperties": []  
    },  
    {  
      "SecurityControlId": "APIGateway.1",  
      "Title": "API Gateway REST and WebSocket API execution logging should be  
enabled",
```

```

        "Description": "This control checks whether all stages of an Amazon
        API Gateway REST or WebSocket API have logging enabled. The control fails if
        the 'loggingLevel' isn't 'ERROR' or 'INFO' for all stages of the API. Unless you
        provide custom parameter values to indicate that a specific log type should be
        enabled, Security Hub produces a passed finding if the logging level is either
        'ERROR' or 'INFO'.",
        "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
        APIGateway.1/remediation",
        "SeverityRating": "MEDIUM",
        "CurrentRegionAvailability": "AVAILABLE",
        "CustomizableProperties": [
            "Parameters"
        ]
    },
    ],
    "NextToken": "U2FsdGvkX1/UprCPzxVbkDeHikDXbDxfgJZ1w2RG1XWsFPTMTIQPVE0m/
    FduIGxS70bRtAbaUt/8/RCQcg2PU0YXI20hH/Grho0Tgv+Tsm0qvQVfhkJepWmqh
    +NYawjocVBeos6xzn/8qnbF9IuwGg=="
}

```

Para obtener más información, consulte [Visualización de los detalles de un estándar](#) en la Guía del usuario de AWS Security Hub.

Ejemplo 2: Para enumerar los controles de seguridad disponibles para un estándar específico

En el siguiente `list-security-control-definitions` ejemplo, se enumeran los controles de seguridad disponibles para la versión 1.4.0 de CIS AWS Foundations Benchmark. Este ejemplo limita los resultados a tres controles.

```

aws securityhub list-security-control-definitions \
  --standards-arn "arn:aws:securityhub:us-east-1::standards/cis-aws-foundations-
  benchmark/v/1.4.0" \
  --max-items 3

```

Salida:

```

{
  "SecurityControlDefinitions": [
    {
      "SecurityControlId": "CloudTrail.1",
      "Title": "CloudTrail should be enabled and configured with at least one
      multi-Region trail that includes read and write management events",

```

```

        "Description": "This AWS control checks that there is at least one
multi-region AWS CloudTrail trail includes read and write management events.",
        "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
CloudTrail.1/remediation",
        "SeverityRating": "HIGH",
        "CurrentRegionAvailability": "AVAILABLE",
        "CustomizableProperties": []
    },
    {
        "SecurityControlId": "CloudTrail.2",
        "Title": "CloudTrail should have encryption at-rest enabled",
        "Description": "This AWS control checks whether AWS CloudTrail is
configured to use the server side encryption (SSE) AWS Key Management Service (AWS
KMS) customer master key (CMK) encryption. The check will pass if the KmsKeyId is
defined.",
        "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
CloudTrail.2/remediation",
        "SeverityRating": "MEDIUM",
        "CurrentRegionAvailability": "AVAILABLE",
        "CustomizableProperties": []
    },
    {
        "SecurityControlId": "CloudTrail.4",
        "Title": "CloudTrail log file validation should be enabled",
        "Description": "This AWS control checks whether CloudTrail log file
validation is enabled.",
        "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
CloudTrail.4/remediation",
        "SeverityRating": "MEDIUM",
        "CurrentRegionAvailability": "AVAILABLE",
        "CustomizableProperties": []
    }
],
"NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAzfQ=="
}

```

Para obtener más información, consulte [Visualización de los detalles de un estándar](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [ListSecurityControlDefinitions](#) la Referencia de AWS CLI comandos.

## list-standards-control-associations

En el siguiente ejemplo de código, se muestra cómo usar `list-standards-control-associations`.

### AWS CLI

Para obtener el estado de habilitación de un control en cada estándar habilitado

En el siguiente `list-standards-control-associations` ejemplo, se muestra el estado de activación de CloudTrail .1 en cada estándar habilitado.

```
aws securityhub list-standards-control-associations \  
--security-control-id CloudTrail.1
```

Salida:

```
{  
  "StandardsControlAssociationSummaries": [  
    {  
      "StandardsArn": "arn:aws:securityhub:us-east-2::standards/nist-800-53/  
v/5.0.0",  
      "SecurityControlId": "CloudTrail.1",  
      "SecurityControlArn": "arn:aws:securityhub:us-  
east-2:123456789012:security-control/CloudTrail.1",  
      "AssociationStatus": "ENABLED",  
      "RelatedRequirements": [  
        "NIST.800-53.r5 AC-2(4)",  
        "NIST.800-53.r5 AC-4(26)",  
        "NIST.800-53.r5 AC-6(9)",  
        "NIST.800-53.r5 AU-10",  
        "NIST.800-53.r5 AU-12",  
        "NIST.800-53.r5 AU-2",  
        "NIST.800-53.r5 AU-3",  
        "NIST.800-53.r5 AU-6(3)",  
        "NIST.800-53.r5 AU-6(4)",  
        "NIST.800-53.r5 AU-14(1)",  
        "NIST.800-53.r5 CA-7",  
        "NIST.800-53.r5 SC-7(9)",  
        "NIST.800-53.r5 SI-3(8)",  
        "NIST.800-53.r5 SI-4(20)",  
        "NIST.800-53.r5 SI-7(8)",  
        "NIST.800-53.r5 SA-8(22)"  
      ]  
    }  
  ]  
}
```



```

    ],
    "UpdatedAt": "2023-05-15T17:52:21.304000+00:00",
    "StandardsControlTitle": "CloudTrail should be enabled and configured
with at least one multi-Region trail that includes read and write management
events",
    "StandardsControlDescription": "This AWS control checks that there is
at least one multi-region AWS CloudTrail trail includes read and write management
events."
  },
  {
    "StandardsArn": "arn:aws:securityhub::ruleset/cis-aws-foundations-
benchmark/v/1.2.0",
    "SecurityControlId": "CloudTrail.1",
    "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/CloudTrail.1",
    "AssociationStatus": "ENABLED",
    "RelatedRequirements": [
      "CIS AWS Foundations 2.1"
    ],
    "UpdatedAt": "2020-02-10T21:22:53.998000+00:00",
    "StandardsControlTitle": "Ensure CloudTrail is enabled in all regions",
    "StandardsControlDescription": "AWS CloudTrail is a web service that
records AWS API calls for your account and delivers log files to you. The recorded
information includes the identity of the API caller, the time of the API call,
the source IP address of the API caller, the request parameters, and the response
elements returned by the AWS service."
  },
  {
    "StandardsArn": "arn:aws:securityhub:us-east-2::standards/aws-
foundational-security-best-practices/v/1.0.0",
    "SecurityControlId": "CloudTrail.1",
    "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/CloudTrail.1",
    "AssociationStatus": "DISABLED",
    "RelatedRequirements": [],
    "UpdatedAt": "2023-05-15T19:31:52.671000+00:00",
    "UpdatedReason": "Alternative compensating controls are in place",
    "StandardsControlTitle": "CloudTrail should be enabled and configured
with at least one multi-Region trail that includes read and write management
events",
    "StandardsControlDescription": "This AWS control checks that there is
at least one multi-region AWS CloudTrail trail includes read and write management
events."
  },
}

```

```

    {
      "StandardsArn": "arn:aws:securityhub:us-east-2::standards/cis-aws-
foundations-benchmark/v/1.4.0",
      "SecurityControlId": "CloudTrail.1",
      "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/CloudTrail.1",
      "AssociationStatus": "ENABLED",
      "RelatedRequirements": [
        "CIS AWS Foundations Benchmark v1.4.0/3.1"
      ],
      "UpdatedAt": "2022-11-10T15:40:36.021000+00:00",
      "StandardsControlTitle": "Ensure CloudTrail is enabled in all regions",
      "StandardsControlDescription": "AWS CloudTrail is a web service that
records AWS API calls for your account and delivers log files to you. The recorded
information includes the identity of the API caller, the time of the API call,
the source IP address of the API caller, the request parameters, and the response
elements returned by the AWS service. CloudTrail provides a history of AWS API
calls for an account, including API calls made via the Management Console, SDKs,
command line tools, and higher-level AWS services (such as CloudFormation)."
    }
  ]
}

```

Para obtener más información, consulte [Habilitar y deshabilitar controles en estándares específicos en](#) la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [ListStandardsControlAssociations](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para recuperar las etiquetas asignadas a un recurso

El siguiente `list-tags-for-resource` ejemplo devuelve las etiquetas asignadas al recurso central especificado.

```

aws securityhub list-tags-for-resource \
  --resource-arn "arn:aws:securityhub:us-west-1:123456789012:hub/default"

```

Salida:

```
{
  "Tags": {
    "Department" : "Operations",
    "Area" : "USMidwest"
  }
}
```

Para obtener más información, consulte [AWS:SecurityHub: :Hub](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## start-configuration-policy-association

En el siguiente ejemplo de código, se muestra cómo usar `start-configuration-policy-association`.

AWS CLI

Ejemplo 1: Para asociar una política de configuración

El siguiente `start-configuration-policy-association` ejemplo asocia la política de configuración especificada a la unidad organizativa especificada. Una configuración puede estar asociada a una cuenta de destino, a una unidad organizativa o a la raíz.

```
aws securityhub start-configuration-policy-association \
  --configuration-policy-identifier "arn:aws:securityhub:eu-
  central-1:123456789012:configuration-policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333" \
  --target '{"OrganizationalUnitId": "ou-6hi7-8j91kl2m"}'
```

Salida:

```
{
  "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
  "TargetId": "ou-6hi7-8j91kl2m",
  "TargetType": "ORGANIZATIONAL_UNIT",
  "AssociationType": "APPLIED",
  "UpdatedAt": "2023-11-29T17:40:52.468000+00:00",
```

```
"AssociationStatus": "PENDING"
}
```

Para obtener más información, consulte [Creación y asociación de políticas de configuración de Security Hub](#) en la Guía del usuario AWS de Security Hub.

Ejemplo 2: Para asociar una configuración autogestionada

El siguiente `start-configuration-policy-association` ejemplo asocia una configuración autogestionada a la cuenta especificada.

```
aws securityhub start-configuration-policy-association \
  --configuration-policy-identifier "SELF_MANAGED_SECURITY_HUB" \
  --target '{"OrganizationalUnitId": "123456789012"}
```

Salida:

```
{
  "ConfigurationPolicyId": "SELF_MANAGED_SECURITY_HUB",
  "TargetId": "123456789012",
  "TargetType": "ACCOUNT",
  "AssociationType": "APPLIED",
  "UpdatedAt": "2023-11-29T17:40:52.468000+00:00",
  "AssociationStatus": "PENDING"
}
```

Para obtener más información, consulte [Creación y asociación de políticas de configuración de Security Hub](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [StartConfigurationPolicyAssociation](#) la Referencia de AWS CLI comandos.

## **start-configuration-policy-disassociation**

En el siguiente ejemplo de código, se muestra cómo usar `start-configuration-policy-disassociation`.

AWS CLI

Ejemplo 1: Para desasociar una política de configuración

En el siguiente `start-configuration-policy-disassociation` ejemplo, se disocia una política de configuración de la unidad organizativa especificada. Una configuración puede estar disociada de una cuenta de destino, de una unidad organizativa o de la raíz.

```
aws securityhub start-configuration-policy-disassociation \  
  --configuration-policy-identifier "arn:aws:securityhub:eu-  
central-1:123456789012:configuration-policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333" \  
  --target '{"OrganizationalUnitId": "ou-6hi7-8j91k12m"}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Desasociar una configuración de las cuentas y OUs](#) en la Guía del usuario AWS de Security Hub.

Ejemplo 2: Para desasociar una configuración autogestionada

El siguiente `start-configuration-policy-disassociation` ejemplo desasocia una configuración autogestionada de la cuenta especificada.

```
aws securityhub start-configuration-policy-disassociation \  
  --configuration-policy-identifier "SELF_MANAGED_SECURITY_HUB" \  
  --target '{"AccountId": "123456789012"}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Desasociar una configuración de las cuentas y OUs](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte la Referencia [StartConfigurationPolicyDisassociation](#) de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para asignar una etiqueta a un recurso

El siguiente `tag-resource` ejemplo asigna valores para las etiquetas de departamento y área al recurso central especificado.

```
aws securityhub tag-resource \  
  --resource-arn "arn:aws:securityhub:us-west-1:123456789012:hub/default" \  
  --tags '{"Department":"Operations", "Area":"USMidwest"}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [AWS:SecurityHub: :Hub](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar untag-resource.

### AWS CLI

Para eliminar el valor de una etiqueta de un recurso

En el siguiente untag-resource ejemplo, se elimina la etiqueta Department del recurso central especificado.

```
aws securityhub untag-resource \  
  --resource-arn "arn:aws:securityhub:us-west-1:123456789012:hub/default" \  
  --tag-keys "Department"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [AWS:SecurityHub: :Hub](#) en la Guía del AWS CloudFormation usuario.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-action-target

En el siguiente ejemplo de código, se muestra cómo usar update-action-target.

### AWS CLI

Para actualizar una acción personalizada

En el siguiente `update-action-target` ejemplo, se actualiza el nombre de la acción personalizada identificada por la acción especificadaARN.

```
aws securityhub update-action-target \
  --action-target-arn "arn:aws:securityhub:us-west-1:123456789012:action/custom/Remediation" \
  --name "Send to remediation"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Crear una acción personalizada y asociarla a una regla de CloudWatch eventos](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [UpdateActionTarget](#) la Referencia de AWS CLI comandos.

## update-configuration-policy

En el siguiente ejemplo de código, se muestra cómo usar `update-configuration-policy`.

AWS CLI

Para actualizar una política de configuración

El siguiente `update-configuration-policy` ejemplo actualiza una política de configuración existente para usar los parámetros especificados.

```
aws securityhub update-configuration-policy \
  --identifier "arn:aws:securityhub:eu-central-1:508236694226:configuration-policy/09f37766-57d8-4ede-9d33-5d8b0fecf70e" \
  --name "SampleConfigurationPolicyUpdated" \
  --description "SampleDescriptionUpdated" \
  --configuration-policy '{"SecurityHub": {"ServiceEnabled": true, "EnabledStandardIdentifiers": ["arn:aws:securityhub:eu-central-1::standards/aws-foundational-security-best-practices/v/1.0.0", "arn:aws:securityhub::ruleset/cis-aws-foundations-benchmark/v/1.2.0"], "SecurityControlsConfiguration": {"DisabledSecurityControlIdentifiers": ["CloudWatch.1"], "SecurityControlCustomParameters": [{"SecurityControlId": "ACM.1", "Parameters": {"daysToExpiration": {"ValueType": "CUSTOM", "Value": {"Integer": 21}}}}]}}' \
  --updated-reason "Disabling CloudWatch.1 and changing parameter value"
```

**Salida:**

```
{
  "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Name": "SampleConfigurationPolicyUpdated",
  "Description": "SampleDescriptionUpdated",
  "UpdatedAt": "2023-11-28T20:28:04.494000+00:00",
  "CreatedAt": "2023-11-28T20:28:04.494000+00:00",
  "ConfigurationPolicy": {
    "SecurityHub": {
      "ServiceEnabled": true,
      "EnabledStandardIdentifiers": [
        "arn:aws:securityhub:eu-central-1::standards/aws-foundational-security-best-practices/v/1.0.0",
        "arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/v/1.2.0"
      ],
      "SecurityControlsConfiguration": {
        "DisabledSecurityControlIdentifiers": [
          "CloudWatch.1"
        ],
        "SecurityControlCustomParameters": [
          {
            "SecurityControlId": "ACM.1",
            "Parameters": {
              "daysToExpiration": {
                "ValueType": "CUSTOM",
                "Value": {
                  "Integer": 21
                }
              }
            }
          }
        ]
      }
    }
  }
}
```

Para obtener más información, consulte [Actualización de las políticas de configuración de Security Hub](#) en la Guía del usuario de AWS Security Hub.



- Para API obtener más información, consulte [UpdateConfigurationPolicy](#) la Referencia de AWS CLI comandos.

## update-finding-aggregator

En el siguiente ejemplo de código, se muestra cómo usar `update-finding-aggregator`.

### AWS CLI

Para actualizar la configuración de agregación de búsquedas actual

El siguiente `update-finding-aggregator` ejemplo cambia la configuración de agregación de búsqueda para vincular desde las regiones seleccionadas. Se ejecuta desde EE. UU. Este (Virginia), que es la región de agregación. Selecciona EE.UU. Oeste (Norte de California) y EE.UU. Oeste (Oregón) como regiones vinculadas.

```
aws securityhub update-finding-aggregator \
  --region us-east-1 \
  --finding-aggregator-arn arn:aws:securityhub:us-east-1:222222222222:finding-agggregator/123e4567-e89b-12d3-a456-426652340000 \
  --region-linking-mode SPECIFIED_REGIONS \
  --regions us-west-1,us-west-2
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Actualización de la configuración de agregación de búsqueda](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [UpdateFindingAggregator](#) la Referencia de AWS CLI comandos.

## update-insight

En el siguiente ejemplo de código, se muestra cómo usar `update-insight`.

### AWS CLI

Ejemplo 1: Para cambiar el filtro para obtener una información personalizada

En el siguiente `update-insight` ejemplo, se cambian los filtros de una información personalizada. La información actualizada busca hallazgos de gran gravedad relacionados con las AWS funciones.

```
aws securityhub update-insight \
  --insight-arn "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" \
  --filters '{"ResourceType": [{"Comparison": "EQUALS", "Value": "AwsIamRole"}], "SeverityLabel": [{"Comparison": "EQUALS", "Value": "HIGH"}]}' \
  --name "High severity role findings"
```

Ejemplo 2: Para cambiar el atributo de agrupación para obtener una información personalizada

El siguiente update-insight ejemplo cambia el atributo de agrupación de la información personalizada por el especificado. ARN El nuevo atributo de agrupación es el ID del recurso.

```
aws securityhub update-insight \
  --insight-arn "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" \
  --group-by-attribute "ResourceId" \
  --name "Critical role findings"
```

Salida:

```
{
  "Insights": [
    {
      "InsightArn": "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Name": "Critical role findings",
      "Filters": {
        "SeverityLabel": [
          {
            "Value": "CRITICAL",
            "Comparison": "EQUALS"
          }
        ],
        "ResourceType": [
          {
            "Value": "AwsIamRole",
            "Comparison": "EQUALS"
          }
        ]
      },
      "GroupByAttribute": "ResourceId"
    }
  ]
}
```

```
    }  
  ]  
}
```

Para obtener más información, consulte [Administrar información personalizada](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [UpdateInsight](#) la Referencia de AWS CLI comandos.

## update-organization-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-organization-configuration`.

### AWS CLI

Para actualizar la configuración de Security Hub para una organización

El siguiente `update-organization-configuration` ejemplo especifica que Security Hub debe usar una configuración central para configurar una organización. Tras ejecutar este comando, el administrador delegado de Security Hub puede crear y gestionar políticas de configuración para configurar la organización. El administrador delegado también puede usar este comando para cambiar de la configuración central a la local. Si el tipo de configuración es la configuración local, el administrador delegado puede elegir si desea habilitar automáticamente Security Hub y los estándares de seguridad predeterminados en las nuevas cuentas de la organización.

```
aws securityhub update-organization-configuration \  
  --no-auto-enable \  
  --organization-configuration '{"ConfigurationType": "CENTRAL"}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar cuentas con AWS Organizations](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [UpdateOrganizationConfiguration](#) la Referencia de AWS CLI comandos.

## update-security-control

En el siguiente ejemplo de código, se muestra cómo usar `update-security-control`.

### AWS CLI

Para actualizar las propiedades del control de seguridad

En el siguiente `update-security-control` ejemplo, se especifican valores personalizados para un parámetro de control de seguridad de Security Hub.

```
aws securityhub update-security-control \  
  --security-control-id ACM.1 \  
  --parameters '{"daysToExpiration": {"ValueType": "CUSTOM", "Value": {"Integer":  
15}}}' \  
  --last-update-reason "Internal compliance requirement"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Parámetros de control personalizados](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [UpdateSecurityControl](#) la Referencia de AWS CLI comandos.

## update-security-hub-configuration

En el siguiente ejemplo de código, se muestra cómo usar `update-security-hub-configuration`.

### AWS CLI

Para actualizar la configuración de Security Hub

El siguiente `update-security-hub-configuration` ejemplo configura Security Hub para habilitar automáticamente nuevos controles para los estándares habilitados.

```
aws securityhub update-security-hub-configuration \  
  --auto-enable-controls
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Habilitar nuevos controles automáticamente](#) en la Guía del usuario de AWS Security Hub.

- Para API obtener más información, consulte [UpdateSecurityHubConfiguration](#) la Referencia de AWS CLI comandos.

## update-standards-control

En el siguiente ejemplo de código, se muestra cómo usar `update-standards-control`.

### AWS CLI

Ejemplo 1: Para deshabilitar un control

En el `update-standards-control` ejemplo siguiente se desactiva el PCI. AutoScaling1. Control.

```
aws securityhub update-standards-control \  
  --standards-control-arn "arn:aws:securityhub:us-west-1:123456789012:control/pci-  
dss/v/3.2.1/PCI.AutoScaling.1" \  
  --control-status "DISABLED" \  
  --disabled-reason "Not applicable for my service"
```

Este comando no genera ninguna salida.

Ejemplo 2: Para habilitar un control

El siguiente `update-standards-control` ejemplo habilita el PCI. AutoScaling1. Control.

```
aws securityhub update-standards-control \  
  --standards-control-arn "arn:aws:securityhub:us-west-1:123456789012:control/pci-  
dss/v/3.2.1/PCI.AutoScaling.1" \  
  --control-status "ENABLED"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Deshabilitar y habilitar controles individuales](#) en la Guía del usuario AWS de Security Hub.

- Para API obtener más información, consulte [UpdateStandardsControl](#) la Referencia de AWS CLI comandos.

# AWS Serverless Application Repository ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Serverless Application Repository.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

Acciones

## **put-application-policy**

En el siguiente ejemplo de código, se muestra cómo usar `put-application-policy`.

AWS CLI

Ejemplo 1: Para compartir una aplicación públicamente

A continuación, se `put-application-policy` comparte una aplicación públicamente, de modo que cualquier persona puede encontrarla e implementarla en AWS Serverless Application Repository.

```
aws serverlessrepo put-application-policy \  
  --application-id arn:aws:serverlessrepo:us-east-1:123456789012:applications/my-  
test-application \  
  --statements Principals='*',Actions=Deploy
```

Salida:

```
{  
  "Statements": [  
    {  
      "Action": "serverlessrepo:PutApplicationPolicy",  
      "Principal": "*" }  
    ]  
}
```

```

    {
      "Actions": [
        "Deploy"
      ],
      "Principals": [
        ""
      ],
      "StatementId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
    }
  ]
}

```

## Ejemplo 2: compartir una aplicación de forma privada

Lo siguiente `put-application-policy` comparte una aplicación de forma privada, por lo que solo AWS cuentas específicas pueden encontrar e implementar su aplicación en AWS Serverless Application Repository.

```

aws serverlessrepo put-application-policy \
  --application-id arn:aws:serverlessrepo:us-east-1:123456789012:applications/my-test-application \
  --statements Principals=111111111111,222222222222,Actions=Deploy

```

Salida:

```

{
  "Statements": [
    {
      "Actions": [
        "Deploy"
      ],
      "Principals": [
        "111111111111",
        "222222222222"
      ],
      "StatementId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
    }
  ]
}

```

Para obtener más información, consulte [Compartir una aplicación a través de la consola en la Guía](#) para desarrolladores de AWS Serverless Application Repository

- Para API obtener más información, consulte [PutApplicationPolicy](#) la Referencia de AWS CLI comandos.

## Ejemplos de Service Catalog que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface with Service Catalog.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **accept-portfolio-share**

En el siguiente ejemplo de código, se muestra cómo usar `accept-portfolio-share`.

AWS CLI

Para aceptar una participación en la cartera

En el siguiente `accept-portfolio-share` ejemplo, se acepta una oferta, realizada por otro usuario, para compartir la cartera especificada.

```
aws servicecatalog accept-portfolio-share \  
  --portfolio-id port-2s6wuabcdefghijk
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [AcceptPortfolioShare](#) la Referencia de AWS CLI comandos.



## associate-principal-with-portfolio

En el siguiente ejemplo de código, se muestra cómo usar `associate-principal-with-portfolio`.

### AWS CLI

Para asociar un principal a una cartera

En el siguiente `associate-principal-with-portfolio` ejemplo, se asocia un usuario a la cartera especificada.

```
aws servicecatalog associate-principal-with-portfolio \  
  --portfolio-id port-2s6abcdefwdh4 \  
  --principal-arn arn:aws:iam::123456789012:user/usertest \  
  --principal-type IAM
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [AssociatePrincipalWithPortfolio](#) la Referencia de AWS CLI comandos.

## associate-product-with-portfolio

En el siguiente ejemplo de código, se muestra cómo usar `associate-product-with-portfolio`.

### AWS CLI

Para asociar un producto a una cartera

El siguiente `associate-product-with-portfolio` ejemplo asocia el producto dado a la cartera especificada.

```
aws servicecatalog associate-product-with-portfolio \  
  --product-id prod-3p5abcdef3oyk \  
  --portfolio-id port-2s6abcdef5wdh4
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [AssociateProductWithPortfolio](#) la Referencia de AWS CLI comandos.

## associate-tag-option-with-resource

En el siguiente ejemplo de código, se muestra cómo usar `associate-tag-option-with-resource`.

### AWS CLI

Para asociar a `TagOption` a un recurso

El siguiente `associate-tag-option-with-resource` ejemplo asocia lo especificado `TagOption` con el recurso especificado.

```
aws servicecatalog associate-tag-option-with-resource \  
  --resource-id port-2s6abcdq5wdh4 \  
  --tag-option-id tag-p3abc2pkpz5qc
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [AssociateTagOptionWithResource](#) la Referencia de AWS CLI comandos.

## copy-product

En el siguiente ejemplo de código, se muestra cómo usar `copy-product`.

### AWS CLI

Para copiar un producto

En el siguiente `copy-product` ejemplo, se hace una copia del producto especificado mediante un JSON archivo para pasar los parámetros.

```
aws servicecatalog copy-product --cli-input-json file://copy-product-input.json
```

Contenidos de `copy-product-input.json`:

```
{  
  "SourceProductArn": "arn:aws:catalog:us-west-2:123456789012:product/prod-  
tcabcd3syn2xy",  
  "TargetProductName": "copy-of-myproduct",  
  "CopyOptions": [  
    {  
      "Name": "CopyTags",  
      "Value": true  
    }  
  ]  
}
```

```
    "CopyTags"  
  ]  
}
```

Salida:

```
{  
  "CopyProductToken": "copyproduct-abc5defgjkdji"  
}
```

- Para API obtener más información, consulte [CopyProduct](#) la Referencia de AWS CLI comandos.

## create-portfolio-share

En el siguiente ejemplo de código, se muestra cómo usar `create-portfolio-share`.

AWS CLI

Para compartir una cartera con una cuenta

En el siguiente `create-portfolio-share` ejemplo, se comparte la cartera especificada con la cuenta especificada.

```
aws servicecatalog create-portfolio-share \  
  --portfolio-id port-2s6abcdef5wdh4 \  
  --account-id 794123456789
```

Este comando no produce ningún resultado.

- Para API obtener más información, consulte [CreatePortfolioShare](#) la Referencia de AWS CLI comandos.

## create-portfolio

En el siguiente ejemplo de código, se muestra cómo usar `create-portfolio`.

AWS CLI

Para crear una cartera

En el siguiente `create-portfolio` ejemplo se crea una cartera.

```
aws servicecatalog create-portfolio \  
  --provider-name my-provider \  
  --display-name my-portfolio
```

Salida:

```
{  
  "PortfolioDetail": {  
    "ProviderName": "my-provider",  
    "DisplayName": "my-portfolio",  
    "CreatedTime": 1571337221.555,  
    "ARN": "arn:aws:catalog:us-east-2:123456789012:portfolio/  
port-2s6xmplq5wdh4",  
    "Id": "port-2s6xmplq5wdh4"  
  }  
}
```

- Para API obtener más información, consulte [CreatePortfolio](#) la Referencia de AWS CLI comandos.

## create-product

En el siguiente ejemplo de código, se muestra cómo usar create-product.

### AWS CLI

Para crear un producto

En el siguiente create-product ejemplo, se crea un producto mediante un JSON archivo para pasar los parámetros.

```
aws servicecatalog create-product \  
  --cli-input-json file://create-product-input.json
```

Contenidos de create-product-input.json:

```
{  
  "AcceptLanguage": "en",  
  "Name": "test-product",  
  "Owner": "test-owner",
```

```

"Description": "test-description",
"Distributor": "test-distributor",
"SupportDescription": "test-support",
"SupportEmail": "test@amazon.com",
"SupportUrl": "https://aws.amazon.com",
"ProductType": "CLOUD_FORMATION_TEMPLATE",
"Tags": [
  {
    "Key": "region",
    "Value": "us-east-1"
  }
],
"ProvisioningArtifactParameters": {
  "Name": "test-version-name",
  "Description": "test-version-description",
  "Info": {
    "LoadTemplateFromURL": "https://s3-us-west-1.amazonaws.com/
cloudformation-templates-us-west-1/my-cfn-template.template"
  },
  "Type": "CLOUD_FORMATION_TEMPLATE"
}
}

```

**Salida:**

```

{
  "Tags": [
    {
      "Key": "region",
      "Value": "us-east-1"
    }
  ],
  "ProductViewDetail": {
    "CreatedTime": 1576025036.0,
    "ProductARN": "arn:aws:catalog:us-west-2:1234568542028:product/
prod-3p5abcdef3oyk",
    "Status": "CREATED",
    "ProductViewSummary": {
      "Type": "CLOUD_FORMATION_TEMPLATE",
      "Distributor": "test-distributor",
      "SupportUrl": "https://aws.amazon.com",
      "SupportEmail": "test@amazon.com",
      "Id": "prodview-abcd42wvx45um",

```

```

        "SupportDescription": "test-support",
        "ShortDescription": "test-description",
        "Owner": "test-owner",
        "Name": "test-product2",
        "HasDefaultPath": false,
        "ProductId": "prod-3p5abcdef3oyk"
    }
},
"ProvisioningArtifactDetail": {
    "CreatedTime": 1576025036.0,
    "Active": true,
    "Id": "pa-pq3p5lil12a34",
    "Description": "test-version-description",
    "Name": "test-version-name",
    "Type": "CLOUD_FORMATION_TEMPLATE"
}
}

```

- Para API obtener más información, consulte [CreateProduct](#) la Referencia de AWS CLI comandos.

## create-provisioning-artifact

En el siguiente ejemplo de código, se muestra cómo usar `create-provisioning-artifact`.

### AWS CLI

Para crear un artefacto de aprovisionamiento

En el siguiente `create-provisioning-artifact` ejemplo, se crea un artefacto de aprovisionamiento mediante un JSON archivo para pasar los parámetros.

```

aws servicecatalog create-provisioning-artifact \
  --cli-input-json file://create-provisioning-artifact-input.json

```

Contenidos de `create-provisioning-artifact-input.json`:

```

{
  "ProductId": "prod-nfi2abcdefghi",
  "Parameters": {
    "Name": "test-provisioning-artifact",
    "Description": "test description",

```

```

    "Info": {
      "LoadTemplateFromURL": "https://s3-us-west-1.amazonaws.com/
cloudformation-templates-us-west-1/my-cfn-template.template"
    },
    "Type": "CLOUD_FORMATION_TEMPLATE"
  }
}

```

Salida:

```

{
  "Info": {
    "TemplateUrl": "https://s3-us-west-1.amazonaws.com/cloudformation-templates-
us-west-1/my-cfn-template.template"
  },
  "Status": "CREATING",
  "ProvisioningArtifactDetail": {
    "Id": "pa-bb4abcdefwnaio",
    "Name": "test-provisioning-artifact",
    "Description": "test description",
    "Active": true,
    "Type": "CLOUD_FORMATION_TEMPLATE",
    "CreatedTime": 1576022545.0
  }
}

```

- Para API obtener más información, consulte la Referencia [CreateProvisioningArtifact](#) de AWS CLI comandos.

## create-tag-option

En el siguiente ejemplo de código, se muestra cómo usar `create-tag-option`.

AWS CLI

Para crear un `TagOption`

En el siguiente `create-tag-option` ejemplo se crea un `TagOption`.

```

aws servicecatalog create-tag-option
  --key 1234
  --value name

```

Salida:

```
{
  "TagOptionDetail": {
    "Id": "tag-iabcdn4fzjjms",
    "Value": "name",
    "Active": true,
    "Key": "1234"
  }
}
```

- Para API obtener más información, consulte [CreateTagOption](#) la Referencia de AWS CLI comandos.

## delete-portfolio-share

En el siguiente ejemplo de código, se muestra cómo usar `delete-portfolio-share`.

AWS CLI

Para dejar de compartir una cartera con una cuenta

En el siguiente `delete-portfolio-share` ejemplo, se deja de compartir la cartera con la cuenta especificada.

```
aws servicecatalog delete-portfolio-share \
  --portfolio-id port-2s6abcdq5wdh4 \
  --account-id 123456789012
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeletePortfolioShare](#) la Referencia de AWS CLI comandos.

## delete-portfolio

En el siguiente ejemplo de código, se muestra cómo usar `delete-portfolio`.

AWS CLI

Para eliminar una cartera



En el siguiente `delete-portfolio` ejemplo, se elimina la cartera especificada.

```
aws servicecatalog delete-portfolio \  
  --id port-abcdlx4gox4do
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeletePortfolio](#) la Referencia de AWS CLI comandos.

## **delete-product**

En el siguiente ejemplo de código, se muestra cómo usar `delete-product`.

AWS CLI

Para eliminar un producto

En el siguiente `delete-product` ejemplo, se elimina el producto especificado.

```
aws servicecatalog delete-product \  
  --id prod-abcdek6yhbxi
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteProduct](#) la Referencia de AWS CLI comandos.

## **delete-provisioning-artifact**

En el siguiente ejemplo de código, se muestra cómo usar `delete-provisioning-artifact`.

AWS CLI

Para eliminar un artefacto de aprovisionamiento

En el siguiente `delete-provisioning-artifact` ejemplo, se elimina el artefacto de aprovisionamiento especificado.

```
aws servicecatalog delete-provisioning-artifact \  
  --product-id prod-abc2uebuplcpw \  
  --provisioning-artifact-id pa-pqabddii7ouc
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte la Referencia de [DeleteProvisioningArtifact](#) comandos AWS CLI .

## delete-tag-option

En el siguiente ejemplo de código, se muestra cómo usar `delete-tag-option`.

### AWS CLI

Para eliminar un TagOption

En el siguiente `delete-tag-option` ejemplo, se elimina lo especificado TagOption.

```
aws servicecatalog delete-tag-option \  
  --id tag-iabcdn4fzjjms
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteTagOption](#) la Referencia de AWS CLI comandos.

## describe-copy-product-status

En el siguiente ejemplo de código, se muestra cómo usar `describe-copy-product-status`.

### AWS CLI

Para describir el estado de la operación de copia del producto

En el siguiente `describe-copy-product-status` ejemplo, se muestra el estado actual de la operación de copia asíncrona del producto especificada.

```
aws servicecatalog describe-copy-product-status \  
  --copy-product-token copyproduct-znn5tf5abcd3w
```

Salida:

```
{  
  "CopyProductStatus": "SUCCEEDED",  
  "TargetProductId": "prod-os6hog7abcdt2"
```

```
}
```

- Para API obtener más información, consulte la Referencia de [DescribeCopyProductStatus](#) comandos AWS CLI .

## describe-portfolio

En el siguiente ejemplo de código, se muestra cómo usar `describe-portfolio`.

### AWS CLI

Para describir una cartera

En el siguiente `describe-portfolio` ejemplo, se muestran los detalles de la cartera especificada.

```
aws servicecatalog describe-portfolio \  
  --id port-2s6abcdq5wdh4
```

Salida:

```
{  
  "TagOptions": [],  
  "PortfolioDetail": {  
    "ARN": "arn:aws:catalog:us-west-2:687558541234:portfolio/  
port-2s6abcdq5wdh4",  
    "Id": "port-2s6wuzqyq5wdh4",  
    "CreatedTime": 1571337221.555,  
    "DisplayName": "my-portfolio",  
    "ProviderName": "my-provider"  
  },  
  "Tags": []  
}
```

- Para API obtener más información, consulte [DescribePortfolio](#) la Referencia de AWS CLI comandos.

## describe-product-as-admin

En el siguiente ejemplo de código, se muestra cómo usar `describe-product-as-admin`.

## AWS CLI

Para describir un producto como administrador

En el siguiente `describe-product-as-admin` ejemplo, se muestran los detalles del producto especificado con privilegios de administrador.

```
aws servicecatalog describe-product-as-admin \  
  --id prod-abcdcek6yhbx1
```

Salida:

```
{  
  "TagOptions": [],  
  "ProductViewDetail": {  
    "ProductARN": "arn:aws:catalog:us-west-2:687558542028:product/prod-  
abcdcek6yhbx1",  
    "ProductViewSummary": {  
      "SupportEmail": "test@amazon.com",  
      "Type": "CLOUD_FORMATION_TEMPLATE",  
      "Distributor": "test-distributor",  
      "ShortDescription": "test-description",  
      "Owner": "test-owner",  
      "Id": "prodview-wi3l2j4abc6vc",  
      "SupportDescription": "test-support",  
      "ProductId": "prod-abcdcek6yhbx1",  
      "HasDefaultPath": false,  
      "Name": "test-product3",  
      "SupportUrl": "https://aws.amazon.com"  
    },  
    "CreatedTime": 1577136715.0,  
    "Status": "CREATED"  
  },  
  "ProvisioningArtifactSummaries": [  
    {  
      "CreatedTime": 1577136715.0,  
      "Description": "test-version-description",  
      "ProvisioningArtifactMetadata": {  
        "SourceProvisioningArtifactId": "pa-abcdxkkiv5fcm"  
      },  
      "Name": "test-version-name-3",  
      "Id": "pa-abcdxkkiv5fcm"  
    }  
  ]  
}
```

```
    ],
    "Tags": [
      {
        "Value": "iad",
        "Key": "region"
      }
    ]
  }
}
```

- Para API obtener más información, consulte [DescribeProductAsAdmin](#) la Referencia de AWS CLI comandos.

## describe-provisioned-product

En el siguiente ejemplo de código, se muestra cómo usar `describe-provisioned-product`.

### AWS CLI

Para describir un producto aprovisionado

En el siguiente `describe-provisioned-product` ejemplo, se muestran los detalles del producto aprovisionado especificado.

```
aws servicecatalog describe-provisioned-product \
  --id pp-dpom27bm4abcd
```

Salida:

```
{
  "ProvisionedProductDetail": {
    "Status": "ERROR",
    "CreatedTime": 1577222793.358,
    "Arn": "arn:aws:servicecatalog:us-west-2:123456789012:stack/mytestppname3/pp-dpom27bm4abcd",
    "Id": "pp-dpom27bm4abcd",
    "StatusMessage": "AmazonCloudFormationException Parameters: [KeyName] must have values (Service: AmazonCloudFormation; Status Code: 400; Error Code: ValidationError; Request ID: 5528602a-a9ef-427c-825c-f82c31b814f5)",
    "IdempotencyToken": "527c5358-2a1a-4b9e-b1b9-7293b0ddff42",
    "LastRecordId": "rec-tfuawdjovzxge",
    "Type": "CFN_STACK",
    "Name": "mytestppname3"
  }
}
```

```

    },
    "CloudWatchDashboards": []
  }

```

- Para API obtener más información, consulte [DescribeProvisionedProduct](#) la Referencia de AWS CLI comandos.

## describe-provisioning-artifact

En el siguiente ejemplo de código, se muestra cómo usar `describe-provisioning-artifact`.

### AWS CLI

Para describir un artefacto de aprovisionamiento

En el siguiente `describe-provisioning-artifact` ejemplo, se muestran los detalles del artefacto de aprovisionamiento especificado.

```

aws servicecatalog describe-provisioning-artifact \
  --provisioning-artifact-id pa-pcz347abcdcfm \
  --product-id prod-abcdefz3syn2rg

```

Salida:

```

{
  "Info": {
    "TemplateUrl": "https://awsdocs.s3.amazonaws.com/servicecatalog/myexampledevelopment-environment.template"
  },
  "ProvisioningArtifactDetail": {
    "Id": "pa-pcz347abcdcfm",
    "Active": true,
    "Type": "CLOUD_FORMATION_TEMPLATE",
    "Description": "updated description",
    "CreatedTime": 1562097906.0,
    "Name": "updated name"
  },
  "Status": "AVAILABLE"
}

```

- Para API obtener más información, consulte la Referencia [DescribeProvisioningArtifact](#) de AWS CLI comandos.

## describe-tag-option

En el siguiente ejemplo de código, se muestra cómo usar `describe-tag-option`.

### AWS CLI

Para describir un TagOption

En el siguiente `describe-tag-option` ejemplo, se muestran los detalles de lo especificado TagOption.

```
aws servicecatalog describe-tag-option \  
  --id tag-p3tej2abcd5qc
```

Salida:

```
{  
  "TagOptionDetail": {  
    "Active": true,  
    "Id": "tag-p3tej2abcd5qc",  
    "Value": "value-3",  
    "Key": "1234"  
  }  
}
```

- Para API obtener más información, consulte [DescribeTagOption](#) la Referencia de AWS CLI comandos.

## disassociate-principal-from-portfolio

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-principal-from-portfolio`.

### AWS CLI

Para desasociar un principal de una cartera

El siguiente `disassociate-principal-from-portfolio` ejemplo disocia el capital especificado de la cartera.

```
aws servicecatalog disassociate-principal-from-portfolio \  
  --portfolio-id port-2s6abcdq5wdh4 \  
  --principal-id principal-id
```

```
--principal-arn arn:aws:iam::123456789012:group/myendusers
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DisassociatePrincipalFromPortfolio](#) la Referencia de AWS CLI comandos.

## **disassociate-product-from-portfolio**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-product-from-portfolio`.

### AWS CLI

Para desasociar un producto de una cartera

El siguiente `disassociate-product-from-portfolio` ejemplo disocia el producto especificado de la cartera.

```
aws servicecatalog disassociate-product-from-portfolio \  
  --product-id prod-3p5abcdmu3oyk \  
  --portfolio-id port-2s6abcdq5wdh4
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DisassociateProductFromPortfolio](#) la Referencia de AWS CLI comandos.

## **disassociate-tag-option-from-resource**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-tag-option-from-resource`.

### AWS CLI

Para desasociar a TagOption de un recurso

En el siguiente `disassociate-tag-option-from-resource` ejemplo, se disocia lo especificado TagOption del recurso.

```
aws servicecatalog disassociate-tag-option-from-resource \  
  --resource-id port-2s6abcdq5wdh4 \  
  --tag-option-id port-2s6abcdq5wdh4
```



```
--tag-option-id tag-p3abc2pkpz5qc
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DisassociateTagOptionFromResource](#) Referencia de AWS CLI comandos.

## list-accepted-portfolio-shares

En el siguiente ejemplo de código, se muestra cómo usar `list-accepted-portfolio-shares`.

### AWS CLI

Para cotizar las acciones de cartera aceptadas

En el siguiente `list-accepted-portfolio-shares` ejemplo, se enumeran todas las carteras para las que esta cuenta ha aceptado compartir, incluidas únicamente las carteras predeterminadas de Service Catalog.

```
aws servicecatalog list-accepted-portfolio-shares \  
  --portfolio-share-type "AWS_SERVICECATALOG"
```

Salida:

```
{  
  "PortfolioDetails": [  
    {  
      "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/port-  
d2abcd5dpkuma",  
      "Description": "AWS Service Catalog Reference blueprints for often-used  
AWS services such as EC2, S3, RDS, VPC and EMR.",  
      "CreatedTime": 1574456190.687,  
      "ProviderName": "AWS Service Catalog",  
      "DisplayName": "Reference Architectures",  
      "Id": "port-d2abcd5dpkuma"  
    },  
    {  
      "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/port-  
abcdefaua7zpu",  
      "Description": "AWS well-architected blueprints for high reliability  
applications.",  
      "CreatedTime": 1574461496.092,  
      "ProviderName": "AWS Service Catalog",
```

```
        "DisplayName": "High Reliability Architectures",
        "Id": "port-abcdefaua7zpu"
    }
]
}
```

- Para API obtener más información, consulte la Referencia [ListAcceptedPortfolioShares](#) de AWS CLI comandos.

## list-portfolio-access

En el siguiente ejemplo de código, se muestra cómo usar `list-portfolio-access`.

### AWS CLI

Para enumerar las cuentas con acceso a una cartera

En el siguiente `list-portfolio-access` ejemplo, se enumeran las AWS cuentas que tienen acceso a la cartera especificada.

```
aws servicecatalog list-portfolio-access \
  --portfolio-id port-2s6abcdq5wdh4
```

Salida:

```
{
  "AccountIds": [
    "123456789012"
  ]
}
```

- Para API obtener más información, consulte [ListPortfolioAccess](#) la Referencia de AWS CLI comandos.

## list-portfolios-for-product

En el siguiente ejemplo de código, se muestra cómo usar `list-portfolios-for-product`.

### AWS CLI

Para enumerar las carteras asociadas a un producto

En el siguiente `list-portfolios-for-product` ejemplo, se muestran las carteras asociadas al producto especificado.

```
aws servicecatalog list-portfolios-for-product \  
--product-id prod-abcdefz3syn2rg
```

Salida:

```
{  
  "PortfolioDetails": [  
    {  
      "CreatedTime": 1571337221.555,  
      "Id": "port-2s6abcdq5wdh4",  
      "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/  
port-2s6abcdq5wdh4",  
      "DisplayName": "my-portfolio",  
      "ProviderName": "my-provider"  
    },  
    {  
      "CreatedTime": 1559665256.348,  
      "Id": "port-5abcd3e5st4ei",  
      "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/  
port-5abcd3e5st4ei",  
      "DisplayName": "test",  
      "ProviderName": "provider-name"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListPortfoliosForProduct](#) la Referencia de AWS CLI comandos.

## list-portfolios

En el siguiente ejemplo de código, se muestra cómo usar `list-portfolios`.

AWS CLI

Para enumerar carteras

En el siguiente `list-portfolios` ejemplo, se enumeran las carteras de Service Catalog de la región actual.

```
aws servicecatalog list-portfolios
```

Salida:

```
{
  "PortfolioDetails": [
    {
      "CreatedTime": 1559665256.348,
      "ARN": "arn:aws:catalog:us-east-2:123456789012:portfolio/
port-5pzcxmlst4ei",
      "DisplayName": "my-portfolio",
      "Id": "port-5pzcxmlst4ei",
      "ProviderName": "my-user"
    }
  ]
}
```

- Para API obtener más información, consulte [ListPortfolios](#) la Referencia de AWS CLI comandos.

## list-principals-for-portfolio

En el siguiente ejemplo de código, se muestra cómo usar `list-principals-for-portfolio`.

AWS CLI

Para enumerar todos los elementos principales de una cartera

En el siguiente `list-principals-for-portfolio` ejemplo, se enumeran todos los principales de la cartera especificada.

```
aws servicecatalog list-principals-for-portfolio \
  --portfolio-id port-2s6abcdq5wdh4
```

Salida:

```
{
  "Principals": [
    {
      "PrincipalARN": "arn:aws:iam::123456789012:user/usertest",
      "PrincipalType": "IAM"
    }
  ]
}
```

```
    }
  ]
}
```

- Para API obtener más información, consulte [ListPrincipalsForPortfolio](#) la Referencia de AWS CLI comandos.

## list-provisioning-artifacts

En el siguiente ejemplo de código, se muestra cómo usar `list-provisioning-artifacts`.

### AWS CLI

Para enumerar todos los artefactos de aprovisionamiento de un producto

En el siguiente `list-provisioning-artifacts` ejemplo, se enumeran todos los artefactos de aprovisionamiento del producto especificado.

```
aws servicecatalog list-provisioning-artifacts \
  --product-id prod-nfi2abcdefgcpw
```

Salida:

```
{
  "ProvisioningArtifactDetails": [
    {
      "Id": "pa-abcdef54ipm6z",
      "Description": "test-version-description",
      "Type": "CLOUD_FORMATION_TEMPLATE",
      "CreatedTime": 1576021147.0,
      "Active": true,
      "Name": "test-version-name"
    },
    {
      "Id": "pa-bb4zyxwwnaio",
      "Description": "test description",
      "Type": "CLOUD_FORMATION_TEMPLATE",
      "CreatedTime": 1576022545.0,
      "Active": true,
      "Name": "test-provisioning-artifact-2"
    }
  ]
}
```

```
}
```

- Para API obtener más información, consulte [ListProvisioningArtifacts](#) la Referencia de AWS CLI comandos.

## list-resources-for-tag-option

En el siguiente ejemplo de código, se muestra cómo usar `list-resources-for-tag-option`.

### AWS CLI

Para enumerar los recursos asociados a un TagOption

En el siguiente `list-resources-for-tag-option` ejemplo, se enumeran los recursos asociados a lo especificado `TagOption`.

```
aws servicecatalog list-resources-for-tag-option \
  --tag-option-id tag-p3tej2abcd5qc
```

Salida:

```
{
  "ResourceDetails": [
    {
      "ARN": "arn:aws:catalog:us-west-2:123456789012:product/prod-
abcdfz3syn2rg",
      "Name": "my product",
      "Description": "description",
      "CreatedTime": 1562097906.0,
      "Id": "prod-abcdfz3syn2rg"
    }
  ]
}
```

- Para API obtener más información, consulte [ListResourcesForTagOption](#) la Referencia de AWS CLI comandos.

## list-tag-options

En el siguiente ejemplo de código, se muestra cómo usar `list-tag-options`.

## AWS CLI

En el siguiente `list-tag-options` ejemplo se enumeran todos los valores de `TagOptions`.

```
aws servicecatalog list-tag-options
```

Salida:

```
{
  "TagOptionDetails": [
    {
      "Value": "newvalue",
      "Active": true,
      "Id": "tag-iabcdn4fzjjms",
      "Key": "1234"
    },
    {
      "Value": "value1",
      "Active": true,
      "Id": "tag-e3abcdvmwvrzy",
      "Key": "key"
    }
  ]
}
```

- Para API obtener más información, consulte [ListTagOptions](#) la Referencia de AWS CLI comandos.

## provision-product

En el siguiente ejemplo de código, se muestra cómo usar `provision-product`.

## AWS CLI

Para aprovisionar un producto

En el siguiente `provision-product` ejemplo, se aprovisiona el producto especificado mediante el artefacto de aprovisionamiento especificado.

```
aws servicecatalog provision-product \
  --product-id prod-abcdfz3syn2rg \
```

```
--provisioning-artifact-id pa-abc347pcscfm \  
--provisioned-product-name "mytestppname3"
```

Salida:

```
{  
  "RecordDetail": {  
    "RecordId": "rec-tfuawdabcdege",  
    "CreatedTime": 1577222793.362,  
    "ProvisionedProductId": "pp-abcd27bm4mldq",  
    "PathId": "lpv2-abcdg3jp6t5k6",  
    "RecordErrors": [],  
    "ProductId": "prod-abcdfz3syn2rg",  
    "UpdatedTime": 1577222793.362,  
    "RecordType": "PROVISION_PRODUCT",  
    "ProvisionedProductName": "mytestppname3",  
    "ProvisioningArtifactId": "pa-pcz347abcdcfm",  
    "RecordTags": [],  
    "Status": "CREATED",  
    "ProvisionedProductType": "CFN_STACK"  
  }  
}
```

- Para API obtener más información, consulte la Referencia [ProvisionProduct](#) de AWS CLI comandos.

## reject-portfolio-share

En el siguiente ejemplo de código, se muestra cómo usar `reject-portfolio-share`.

AWS CLI

Para rechazar una participación en una cartera

En el siguiente `reject-portfolio-share` ejemplo, se rechaza la parte de cartera de la cartera en cuestión.

```
aws servicecatalog reject-portfolio-share \  
--portfolio-id port-2s6wuabcdefghijk
```

Este comando no genera ninguna salida.



- Para API obtener más información, consulte [RejectPortfolioShare](#) la Referencia de AWS CLI comandos.

## scan-provisioned-products

En el siguiente ejemplo de código, se muestra cómo usar `scan-provisioned-products`.

### AWS CLI

Para ver una lista de todos los productos aprovisionados disponibles

En el siguiente `scan-provisioned-products` ejemplo, se enumeran los productos aprovisionados disponibles.

```
aws servicecatalog scan-provisioned-products
```

Salida:

```
{
  "ProvisionedProducts": [
    {
      "Status": "ERROR",
      "Arn": "arn:aws:servicecatalog:us-west-2:123456789012:stack/mytestppname3/pp-abcd27bm4mldq",
      "StatusMessage": "AmazonCloudFormationException Parameters: [KeyName] must have values (Service: AmazonCloudFormation; Status Code: 400; Error Code: ValidationError; Request ID: 5528602a-a9ef-427c-825c-f82c31b814f5)",
      "Id": "pp-abcd27bm4mldq",
      "Type": "CFN_STACK",
      "IdempotencyToken": "527c5358-2a1a-4b9e-b1b9-7293b0ddff42",
      "CreatedTime": 1577222793.358,
      "Name": "mytestppname3",
      "LastRecordId": "rec-tfuawdabcdxge"
    }
  ]
}
```

- Para API obtener más información, consulte [ScanProvisionedProducts](#) la Referencia de AWS CLI comandos.

## search-products-as-admin

En el siguiente ejemplo de código, se muestra cómo usar `search-products-as-admin`.

### AWS CLI

Para buscar productos con privilegios de administrador

En el siguiente `search-products-as-admin` ejemplo, se buscan productos con privilegios de administrador utilizando un identificador de cartera como filtro.

```
aws servicecatalog search-products-as-admin \  
  --portfolio-id port-5abcd3e5st4ei
```

Salida:

```
{  
  "ProductViewDetails": [  
    {  
      "ProductViewSummary": {  
        "Name": "my product",  
        "Owner": "owner name",  
        "Type": "CLOUD_FORMATION_TEMPLATE",  
        "ProductId": "prod-abcdefz3syn2rg",  
        "HasDefaultPath": false,  
        "Id": "prodview-abcdmyuzv2dlu",  
        "ShortDescription": "description"  
      },  
      "ProductARN": "arn:aws:catalog:us-west-2:123456789012:product/prod-  
abcdefz3syn2rg",  
      "CreatedTime": 1562097906.0,  
      "Status": "CREATED"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [SearchProductsAsAdmin](#) la Referencia de AWS CLI comandos.

## search-provisioned-products

En el siguiente ejemplo de código, se muestra cómo usar `search-provisioned-products`.

## AWS CLI

Para buscar productos aprovisionados

En el siguiente `search-provisioned-products` ejemplo, se buscan productos aprovisionados que coincidan con el identificador de producto especificado mediante un JSON archivo para transferir los parámetros.

```
aws servicecatalog search-provisioned-products \  
--cli-input-json file://search-provisioned-products-input.json
```

Contenidos de `search-provisioned-products-input.json`:

```
{  
  "Filters": {  
    "SearchQuery": [  
      "prod-tcjevz3syn2rg"  
    ]  
  }  
}
```

Salida:

```
{  
  "ProvisionedProducts": [  
    {  
      "ProvisioningArtifactId": "pa-pcz347abcdcfm",  
      "Name": "mytestppname3",  
      "CreatedTime": 1577222793.358,  
      "Id": "pp-abcd27bm4mldq",  
      "Status": "ERROR",  
      "UserArn": "arn:aws:iam::123456789012:user/cliuser",  
      "StatusMessage": "AmazonCloudFormationException Parameters: [KeyName]  
must have values (Service: AmazonCloudFormation; Status Code: 400; Error Code:  
ValidationError; Request ID: 5528602a-a9ef-427c-825c-f82c31b814f5)",  
      "Arn": "arn:aws:servicecatalog:us-west-2:123456789012:stack/  
mytestppname3/pp-abcd27bm4mldq",  
      "Tags": [  
        {  
          "Value": "arn:aws:catalog:us-west-2:123456789012:product/prod-  
abcdfz3syn2rg",  
          "Key": "aws:servicecatalog:productArn"  
        }  
      ],  
    }  
  ],  
}
```

```

        {
            "Value": "arn:aws:iam::123456789012:user/cliuser",
            "Key": "aws:servicecatalog:provisioningPrincipalArn"
        },
        {
            "Value": "value-3",
            "Key": "1234"
        },
        {
            "Value": "pa-pcz347abcdcfm",
            "Key": "aws:servicecatalog:provisioningArtifactIdentifier"
        },
        {
            "Value": "arn:aws:catalog:us-west-2:123456789012:portfolio/
port-2s6abcdq5wdh4",
            "Key": "aws:servicecatalog:portfolioArn"
        },
        {
            "Value": "arn:aws:servicecatalog:us-west-2:123456789012:stack/
mytestppname3/pp-abcd27bm4mldq",
            "Key": "aws:servicecatalog:provisionedProductArn"
        }
    ],
    "IdempotencyToken": "527c5358-2a1a-4b9e-b1b9-7293b0ddff42",
    "UserArnSession": "arn:aws:iam::123456789012:user/cliuser",
    "Type": "CFN_STACK",
    "LastRecordId": "rec-tfuawdabcdxge",
    "ProductId": "prod-abcdfz3syn2rg"
}
],
"TotalResultsCount": 1
}

```

- Para API obtener más información, consulte [SearchProvisionedProducts](#) la Referencia de AWS CLI comandos.

## update-portfolio

En el siguiente ejemplo de código, se muestra cómo usar `update-portfolio`.

### AWS CLI

Para actualizar una cartera

En el siguiente `update-portfolio` ejemplo, se actualiza el nombre de la cartera especificada.

```
aws servicecatalog update-portfolio \  
  --id port-5abcd3e5st4ei \  
  --display-name "New portfolio name"
```

Salida:

```
{  
  "PortfolioDetail": {  
    "DisplayName": "New portfolio name",  
    "ProviderName": "provider",  
    "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/  
port-5abcd3e5st4ei",  
    "Id": "port-5abcd3e5st4ei",  
    "CreatedTime": 1559665256.348  
  },  
  "Tags": []  
}
```

- Para API obtener más información, consulte [UpdatePortfolio](#) la Referencia de AWS CLI comandos.

## update-product

En el siguiente ejemplo de código, se muestra cómo usar `update-product`.

### AWS CLI

Para actualizar un producto

En el siguiente `update-product` ejemplo, se actualizan el nombre y el propietario del producto especificado.

```
aws servicecatalog update-product \  
  --id prod-os6abc7drqlt2 \  
  --name "New product name" \  
  --owner "Updated product owner"
```

Salida:

```
{
```

```
"Tags": [
  {
    "Value": "iad",
    "Key": "region"
  }
],
"ProductViewDetail": {
  "ProductViewSummary": {
    "Owner": "Updated product owner",
    "ProductId": "prod-os6abc7drqlt2",
    "Distributor": "test-distributor",
    "SupportUrl": "https://aws.amazon.com",
    "Name": "New product name",
    "ShortDescription": "test-description",
    "HasDefaultPath": false,
    "Id": "prodview-6abcdgrfhvidy",
    "SupportDescription": "test-support",
    "SupportEmail": "test@amazon.com",
    "Type": "CLOUD_FORMATION_TEMPLATE"
  },
  "Status": "CREATED",
  "ProductARN": "arn:aws:catalog:us-west-2:123456789012:product/prod-
os6abc7drqlt2",
  "CreatedTime": 1577136255.0
}
}
```

- Para API obtener más información, consulte [UpdateProduct](#) la Referencia de AWS CLI comandos.

## update-provisioning-artifact

En el siguiente ejemplo de código, se muestra cómo usar `update-provisioning-artifact`.

### AWS CLI

Para actualizar un artefacto de aprovisionamiento

En el siguiente `update-provisioning-artifact` ejemplo, se actualizan el nombre y la descripción del artefacto de aprovisionamiento especificado mediante un JSON archivo para transferir los parámetros.

```
aws servicecatalog update-provisioning-artifact \
```

```
--cli-input-json file://update-provisioning-artifact-input.json
```

Contenidos de `update-provisioning-artifact-input.json`:

```
{
  "ProductId": "prod-abcdefz3syn2rg",
  "ProvisioningArtifactId": "pa-pcz347abcdcfm",
  "Name": "updated name",
  "Description": "updated description"
}
```

Salida:

```
{
  "Info": {
    "TemplateUrl": "https://awsdocs.s3.amazonaws.com/servicecatalog/
myexampledevelopment-environment.template"
  },
  "Status": "AVAILABLE",
  "ProvisioningArtifactDetail": {
    "Active": true,
    "Description": "updated description",
    "Id": "pa-pcz347abcdcfm",
    "Name": "updated name",
    "Type": "CLOUD_FORMATION_TEMPLATE",
    "CreatedTime": 1562097906.0
  }
}
```

- Para API obtener más información, consulte la Referencia [UpdateProvisioningArtifact](#) de AWS CLI comandos.

## update-tag-option

En el siguiente ejemplo de código, se muestra cómo usar `update-tag-option`.

### AWS CLI

Para actualizar un `TagOption`

En el siguiente `update-tag-option` ejemplo `TagOption`, se actualiza el valor de `a` mediante el JSON archivo especificado.

```
aws servicecatalog update-tag-option --cli-input-json file://update-tag-option-input.json
```

Contenidos de `update-tag-option-input.json`:

```
{
  "Id": "tag-iabcdn4fzjjms",
  "Value": "newvalue",
  "Active": true
}
```

Salida:

```
{
  "TagOptionDetail": {
    "Value": "newvalue",
    "Key": "1234",
    "Active": true,
    "Id": "tag-iabcdn4fzjjms"
  }
}
```

- Para API obtener más información, consulte [UpdateTagOption](#) la Referencia de AWS CLI comandos.

## Ejemplos de Service Quotas que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface with Service Quotas.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)



## Acciones

### get-aws-default-service-quota

En el siguiente ejemplo de código, se muestra cómo usar `get-aws-default-service-quota`.

#### AWS CLI

Para describir una cuota de servicio predeterminada

En el siguiente `get-aws-default-service-quota` ejemplo, se muestran los detalles de la cuota especificada.

```
aws service-quotas get-aws-default-service-quota \  
  --service-code ec2 \  
  --quota-code L-1216C47A
```

#### Salida:

```
{  
  "Quota": {  
    "ServiceCode": "ec2",  
    "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",  
    "QuotaArn": "arn:aws:servicequotas:us-east-2::ec2/L-1216C47A",  
    "QuotaCode": "L-1216C47A",  
    "QuotaName": "Running On-Demand Standard (A, C, D, H, I, M, R, T, Z)  
instances",  
    "Value": 5.0,  
    "Unit": "None",  
    "Adjustable": true,  
    "GlobalQuota": false,  
    "UsageMetric": {  
      "MetricNamespace": "AWS/Usage",  
      "MetricName": "ResourceCount",  
      "MetricDimensions": {  
        "Class": "Standard/OnDemand",  
        "Resource": "vCPU",  
        "Service": "EC2",  
        "Type": "Resource"  
      },  
      "MetricStatisticRecommendation": "Maximum"  
    }  
  }  
}
```

```
}
```

- Para API obtener más información, consulte [GetAwsDefaultServiceQuota](#) la Referencia de AWS CLI comandos.

## get-requested-service-quota-change

En el siguiente ejemplo de código, se muestra cómo usar `get-requested-service-quota-change`.

### AWS CLI

Para describir una solicitud de aumento de cuota de servicio

El siguiente `get-requested-service-quota-change` ejemplo describe la solicitud de aumento de cuota especificada.

```
aws service-quotas get-requested-service-quota-change \
  --request-id d187537d15254312a9609aa51bbf7624u7W49tP0
```

Salida:

```
{
  "RequestedQuota": {
    "Id": "d187537d15254312a9609aa51bbf7624u7W49tP0",
    "CaseId": "6780195351",
    "ServiceCode": "ec2",
    "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",
    "QuotaCode": "L-20F13EBD",
    "QuotaName": "Running Dedicated c5n Hosts",
    "DesiredValue": 2.0,
    "Status": "CASE_OPENED",
    "Created": 1580446904.067,
    "LastUpdated": 1580446953.265,
    "Requester": "{\"accountId\":\"123456789012\",\"callerArn\":\
  \"arn:aws:iam::123456789012:root\"}",
    "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/L-20F13EBD",
    "GlobalQuota": false,
    "Unit": "None"
  }
}
```

- Para API obtener más información, consulte [GetRequestedServiceQuotaChange](#) la Referencia de AWS CLI comandos.

## get-service-quota

En el siguiente ejemplo de código, se muestra cómo usar `get-service-quota`.

### AWS CLI

Para describir una cuota de servicio

En el siguiente `get-service-quota` ejemplo, se muestran detalles sobre la cuota especificada.

```
aws service-quotas get-service-quota \  
  --service-code ec2 \  
  --quota-code L-1216C47A
```

Salida:

```
{  
  "Quota": {  
    "ServiceCode": "ec2",  
    "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",  
    "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/L-1216C47A",  
    "QuotaCode": "L-1216C47A",  
    "QuotaName": "Running On-Demand Standard (A, C, D, H, I, M, R, T, Z)  
instances",  
    "Value": 1920.0,  
    "Unit": "None",  
    "Adjustable": true,  
    "GlobalQuota": false,  
    "UsageMetric": {  
      "MetricNamespace": "AWS/Usage",  
      "MetricName": "ResourceCount",  
      "MetricDimensions": {  
        "Class": "Standard/OnDemand",  
        "Resource": "vCPU",  
        "Service": "EC2",  
        "Type": "Resource"  
      },  
      "MetricStatisticRecommendation": "Maximum"  
    }  
  }  
}
```

```
}  
}
```

- Para API obtener más información, consulte [GetServiceQuota](#) la Referencia de AWS CLI comandos.

## list-aws-default-service-quotas

En el siguiente ejemplo de código, se muestra cómo usar `list-aws-default-service-quotas`.

### AWS CLI

Para enumerar las cuotas predeterminadas de un servicio

El siguiente `list-aws-default-service-quotas` ejemplo muestra los valores predeterminados de las cuotas del servicio especificado.

```
aws service-quotas list-aws-default-service-quotas \  
  --service-code xray
```

Salida:

```
{  
  "Quotas": [  
    {  
      "ServiceCode": "xray",  
      "ServiceName": "AWS X-Ray",  
      "QuotaArn": "arn:aws:servicequotas:us-west-2::xray/L-C6B6F05D",  
      "QuotaCode": "L-C6B6F05D",  
      "QuotaName": "Indexed annotations per trace",  
      "Value": 50.0,  
      "Unit": "None",  
      "Adjustable": false,  
      "GlobalQuota": false  
    },  
    {  
      "ServiceCode": "xray",  
      "ServiceName": "AWS X-Ray",  
      "QuotaArn": "arn:aws:servicequotas:us-west-2::xray/L-D781C0FD",  
      "QuotaCode": "L-D781C0FD",  
      "QuotaName": "Segment document size",  
      "Value": 64.0,  
    }  
  ]  
}
```

```

        "Unit": "Kilobytes",
        "Adjustable": false,
        "GlobalQuota": false
    },
    {
        "ServiceCode": "xray",
        "ServiceName": "AWS X-Ray",
        "QuotaArn": "arn:aws:servicequotas:us-west-2::xray/L-998BFF16",
        "QuotaCode": "L-998BFF16",
        "QuotaName": "Trace and service graph retention in days",
        "Value": 30.0,
        "Unit": "None",
        "Adjustable": false,
        "GlobalQuota": false
    }
]
}

```

- Para API obtener más información, consulte [ListAwsDefaultServiceQuotas](#) la Referencia de AWS CLI comandos.

## list-requested-service-quota-change-history-by-quota

En el siguiente ejemplo de código, se muestra cómo usar `list-requested-service-quota-change-history-by-quota`.

### AWS CLI

Para enumerar sus solicitudes de aumento de cuota

El siguiente `list-requested-service-quota-change-history-by-quota` ejemplo muestra las solicitudes de aumento de cuota para la cuota especificada.

```

aws service-quotas list-requested-service-quota-change-history-by-quota \
  --service-code ec2 \
  --quota-code L-20F13EBD

```

Salida:

```

{
  "RequestedQuotas": [

```

```

    {
      "Id": "d187537d15254312a9609aa51bbf7624u7W49tP0",
      "CaseId": "6780195351",
      "ServiceCode": "ec2",
      "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",
      "QuotaCode": "L-20F13EBD",
      "QuotaName": "Running Dedicated c5n Hosts",
      "DesiredValue": 2.0,
      "Status": "CASE_OPENED",
      "Created": 1580446904.067,
      "LastUpdated": 1580446953.265,
      "Requester": "{\"accountId\":\"123456789012\",\"callerArn\":
\\\"arn:aws:iam::123456789012:root\\\"}\",
      "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/
L-20F13EBD",
      "GlobalQuota": false,
      "Unit": "None"
    }
  ]
}

```

- Para API obtener más información, consulte [ListRequestedServiceQuotaChangeHistoryByQuota](#) la Referencia de AWS CLI comandos.

## list-requested-service-quota-change-history

En el siguiente ejemplo de código, se muestra cómo usar `list-requested-service-quota-change-history`.

### AWS CLI

Para enumerar sus solicitudes de aumento de cuota

El siguiente `list-requested-service-quota-change-history` ejemplo muestra las solicitudes de aumento de cuota para el servicio especificado.

```
aws service-quotas list-requested-service-quota-change-history \
  --service-code ec2
```

Salida:

```
{
```

```

    "RequestedQuotas": [
      {
        "Id": "d187537d15254312a9609aa51bbf7624u7W49tP0",
        "CaseId": "6780195351",
        "ServiceCode": "ec2",
        "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",
        "QuotaCode": "L-20F13EBD",
        "QuotaName": "Running Dedicated c5n Hosts",
        "DesiredValue": 2.0,
        "Status": "CASE_OPENED",
        "Created": 1580446904.067,
        "LastUpdated": 1580446953.265,
        "Requester": "{\"accountId\":\"123456789012\",\"callerArn\":\
\\\"arn:aws:iam::123456789012:root\\\"}\",
        "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/
L-20F13EBD",
        "GlobalQuota": false,
        "Unit": "None"
      }
    ]
  }

```

- Para API obtener más información, consulte [ListRequestedServiceQuotaChangeHistory](#) la Referencia de AWS CLI comandos.

## list-service-quotas

En el siguiente ejemplo de código, se muestra cómo usar `list-service-quotas`.

### AWS CLI

Para enumerar las cuotas de un servicio

En el siguiente `list-service-quotas` ejemplo, se muestran detalles sobre las cuotas de AWS CloudFormation.

```

aws service-quotas list-service-quotas \
  --service-code cloudformation

```

Salida:

```

{

```

```
"Quotas": [
  {
    "ServiceCode": "cloudformation",
    "ServiceName": "AWS CloudFormation",
    "QuotaArn": "arn:aws:servicequotas:us-
east-2:123456789012:cloudformation/L-87D14FB7",
    "QuotaCode": "L-87D14FB7",
    "QuotaName": "Output count in CloudFormation template",
    "Value": 60.0,
    "Unit": "None",
    "Adjustable": false,
    "GlobalQuota": false
  },
  {
    "ServiceCode": "cloudformation",
    "ServiceName": "AWS CloudFormation",
    "QuotaArn": "arn:aws:servicequotas:us-
east-2:123456789012:cloudformation/L-0485CB21",
    "QuotaCode": "L-0485CB21",
    "QuotaName": "Stack count",
    "Value": 200.0,
    "Unit": "None",
    "Adjustable": true,
    "GlobalQuota": false
  }
]
```

- Para API obtener más información, consulte [ListServiceQuotas](#) la Referencia de AWS CLI comandos.

## list-services

En el siguiente ejemplo de código, se muestra cómo usar `list-services`.

### AWS CLI

Para ver una lista de los servicios disponibles

El siguiente comando muestra los servicios que están disponibles en Service Quotas.

```
aws service-quotas list-services
```



**Salida:**

```
{
  "Services": [
    {
      "ServiceCode": "AWSCloudMap",
      "ServiceName": "AWS Cloud Map"
    },
    {
      "ServiceCode": "access-analyzer",
      "ServiceName": "Access Analyzer"
    },
    {
      "ServiceCode": "acm",
      "ServiceName": "AWS Certificate Manager (ACM)"
    },
    ...truncated...
    {
      "ServiceCode": "xray",
      "ServiceName": "AWS X-Ray"
    }
  ]
}
```

Puede añadir el `--query` parámetro para filtrar la visualización según la información que le interese. El siguiente ejemplo muestra solo los códigos de servicio.

```
aws service-quotas list-services \  
  --query Services[*].ServiceCode
```

**Salida:**

```
[
  "AWSCloudMap",
  "access-analyzer",
  "acm",
  "acm-pca",
  "amplify",
  "apigateway",
  "application-autoscaling",
```

```

    ...truncated...
    "xray"
]

```

- Para API obtener más información, consulte [ListServices](#) la Referencia de AWS CLI comandos.

## request-service-quota-increase

En el siguiente ejemplo de código, se muestra cómo usar `request-service-quota-increase`.

### AWS CLI

Para solicitar un aumento de la cuota de servicio

El siguiente `request-service-quota-increase` ejemplo solicita un aumento de la cuota de servicio especificada.

```

aws service-quotas request-service-quota-increase \
  --service-code ec2 \
  --quota-code L-20F13EBD \
  --desired-value 2

```

Salida:

```

{
  "RequestedQuota": {
    "Id": "d187537d15254312a9609aa51bbf7624u7W49tP0",
    "ServiceCode": "ec2",
    "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",
    "QuotaCode": "L-20F13EBD",
    "QuotaName": "Running Dedicated c5n Hosts",
    "DesiredValue": 2.0,
    "Status": "PENDING",
    "Created": 1580446904.067,
    "Requester": "{\"accountId\":\"123456789012\",\"callerArn\":\
  \"arn:aws:iam::123456789012:root\"}",
    "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/L-20F13EBD",
    "GlobalQuota": false,
    "Unit": "None"
  }
}

```

- Para API obtener más información, consulte [RequestServiceQuotaIncrease](#) la Referencia de AWS CLI comandos.

## SESEjemplos de Amazon que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante AmazonSES.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **delete-identity**

En el siguiente ejemplo de código, se muestra cómo usar `delete-identity`.

AWS CLI

Para eliminar una identidad

En el siguiente ejemplo, se utiliza el `delete-identity` comando para eliminar una identidad de la lista de identidades verificadas con AmazonSES:

```
aws ses delete-identity --identity user@example.com
```

Para obtener más información sobre las identidades verificadas, consulte Verificación de direcciones de correo electrónico y dominios en Amazon SES en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [DeleteIdentity](#) la Referencia de AWS CLI comandos.

## get-identity-dkim-attributes

En el siguiente ejemplo de código, se muestra cómo usar `get-identity-dkim-attributes`.

### AWS CLI

Para obtener los DKIM atributos de Amazon SES Easy para una lista de identidades

En el siguiente ejemplo, se utiliza el `get-identity-dkim-attributes` comando para recuperar los DKIM atributos de Amazon SES Easy de una lista de identidades:

```
aws ses get-identity-dkim-attributes --identities "example.com" "user@example.com"
```

Salida:

```
{
  "DkimAttributes": {
    "example.com": {
      "DkimTokens": [
        "EXAMPLEjcs5xoyqytjsotsijas7236gr",
        "EXAMPLEjr76cvoc6mysspnioorxsn6ep",
        "EXAMPLEkbnkqkhlm2lyz77ppkulerm4k"
      ],
      "DkimEnabled": true,
      "DkimVerificationStatus": "Success"
    },
    "user@example.com": {
      "DkimEnabled": false,
      "DkimVerificationStatus": "NotStarted"
    }
  }
}
```

Si llama a este comando con una identidad que nunca ha enviado para su verificación, esa identidad no aparecerá en el resultado.

Para obtener más información sobre EasyDKIM, consulta Easy DKIM in Amazon SES en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [GetIdentityDkimAttributes](#) la Referencia de AWS CLI comandos.

## get-identity-notification-attributes

En el siguiente ejemplo de código, se muestra cómo usar `get-identity-notification-attributes`.

### AWS CLI

Para obtener los atributos de SES notificación de Amazon para una lista de identidades

En el siguiente ejemplo, se utiliza el `get-identity-notification-attributes` comando para recuperar los atributos de SES notificación de Amazon para una lista de identidades:

```
aws ses get-identity-notification-attributes --
identities "user1@example.com" "user2@example.com"
```

Salida:

```
{
  "NotificationAttributes": {
    "user1@example.com": {
      "ForwardingEnabled": false,
      "ComplaintTopic": "arn:aws:sns:us-east-1:EXAMPLE65304:MyTopic",
      "BounceTopic": "arn:aws:sns:us-east-1:EXAMPLE65304:MyTopic",
      "DeliveryTopic": "arn:aws:sns:us-east-1:EXAMPLE65304:MyTopic"
    },
    "user2@example.com": {
      "ForwardingEnabled": true
    }
  }
}
```

Este comando devuelve el estado del reenvío de comentarios por correo electrónico y, si corresponde, los nombres de los recursos de Amazon (ARNs) de los SNS temas de Amazon a los que se envían las notificaciones de rebote, queja y entrega.

Si llama a este comando con una identidad que nunca ha enviado para su verificación, esa identidad no aparecerá en el resultado.

Para obtener más información sobre las notificaciones, consulte [Uso de notificaciones con Amazon SES](#) en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [GetIdentityNotificationAttributes](#) la Referencia de AWS CLI comandos.

## get-identity-verification-attributes

En el siguiente ejemplo de código, se muestra cómo usar `get-identity-verification-attributes`.

### AWS CLI

Para obtener el estado SES de verificación de Amazon para una lista de identidades

En el siguiente ejemplo, se utiliza el `get-identity-verification-attributes` comando para recuperar el estado SES de verificación de Amazon de una lista de identidades:

```
aws ses get-identity-verification-attributes --
identities "user1@example.com" "user2@example.com"
```

Salida:

```
{
  "VerificationAttributes": {
    "user1@example.com": {
      "VerificationStatus": "Success"
    },
    "user2@example.com": {
      "VerificationStatus": "Pending"
    }
  }
}
```

Si llama a este comando con una identidad que nunca ha enviado para su verificación, esa identidad no aparecerá en el resultado.

Para obtener más información sobre las identidades verificadas, consulte Verificación de direcciones de correo electrónico y dominios en Amazon SES en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [GetIdentityVerificationAttributes](#) la Referencia de AWS CLI comandos.

## get-send-quota

En el siguiente ejemplo de código, se muestra cómo usar `get-send-quota`.

## AWS CLI

Para conocer tus límites de SES envío en Amazon

En el siguiente ejemplo, se utiliza el `get-send-quota` comando para devolver los límites de SES envío de Amazon:

```
aws ses get-send-quota
```

Salida:

```
{
  "Max24HourSend": 200.0,
  "SentLast24Hours": 1.0,
  "MaxSendRate": 1.0
}
```

`Max24 HourSend` es tu cuota de envío, que es la cantidad máxima de correos electrónicos que puedes enviar en un período de 24 horas. La cuota de envío refleja un período de tiempo acumulativo. Cada vez que intentas enviar un correo electrónico, Amazon SES comprueba cuántos correos electrónicos has enviado en las últimas 24 horas. Siempre que el número total de mensajes de correo electrónico que haya enviado sea inferior a su cuota, se aceptará su solicitud de envío y se enviará su correo electrónico.

`SentLast24 horas` es el número de correos electrónicos que has enviado en las últimas 24 horas.

`MaxSendRate` es el número máximo de correos electrónicos que puedes enviar por segundo.

Tenga en cuenta que los límites de envío se basan en los destinatarios en lugar de en los mensajes. Por ejemplo, un correo electrónico que tenga 10 destinatarios se contabiliza como 10 en la cuota de envío.

Para obtener más información, consulta [Cómo gestionar tus límites de SES envío de Amazon en la Guía para desarrolladores de Amazon Simple Email Service](#).

- Para API obtener más información, consulte [GetSendQuota](#) la Referencia de AWS CLI comandos.

### **get-send-statistics**

En el siguiente ejemplo de código, se muestra cómo usar `get-send-statistics`.

## AWS CLI

Para obtener tus estadísticas de SES envíos de Amazon

En el siguiente ejemplo, se utiliza el `get-send-statistics` comando para devolver las estadísticas de SES envíos de Amazon.

```
aws ses get-send-statistics
```

Salida:

```
{
  "SendDataPoints": [
    {
      "Complaints": 0,
      "Timestamp": "2013-06-12T19:32:00Z",
      "DeliveryAttempts": 2,
      "Bounces": 0,
      "Rejects": 0
    },
    {
      "Complaints": 0,
      "Timestamp": "2013-06-12T00:47:00Z",
      "DeliveryAttempts": 1,
      "Bounces": 0,
      "Rejects": 0
    }
  ]
}
```

El resultado es una lista de puntos de datos que representan las dos últimas semanas de actividad de envío. Cada punto de datos de la lista contiene estadísticas para un intervalo de 15 minutos.

En este ejemplo, solo hay dos puntos de datos porque los únicos correos electrónicos que el usuario envió en las últimas dos semanas se produjeron en dos intervalos de 15 minutos.

Para obtener más información, consulta [Cómo monitorizar tus estadísticas de SES uso de Amazon](#) en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [GetSendStatistics](#) la Referencia de AWS CLI comandos.



## list-identities

En el siguiente ejemplo de código, se muestra cómo usar `list-identities`.

### AWS CLI

Para enumerar todas las identidades (direcciones de correo electrónico y dominios) de una AWS cuenta específica

En el siguiente ejemplo, se utiliza el `list-identities` comando para enumerar todas las identidades que se han enviado para su verificación con AmazonSES:

```
aws ses list-identities
```

Salida:

```
{
  "Identities": [
    "user@example.com",
    "example.com"
  ]
}
```

La lista que se devuelve contiene todas las identidades, independientemente del estado de la verificación (verificada, pendiente de verificación, error, etc.).

En este ejemplo, se devuelven las direcciones de correo electrónico y los dominios porque no especificamos el parámetro de tipo de identidad.

Para obtener más información sobre la verificación, consulte [Verificación de direcciones de correo electrónico y dominios en Amazon SES](#) en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [ListIdentities](#) la Referencia de AWS CLI comandos.

## send-email

En el siguiente ejemplo de código, se muestra cómo usar `send-email`.

### AWS CLI

Para enviar un correo electrónico formateado mediante Amazon SES

En el siguiente ejemplo, se utiliza el comando `send-email` para enviar un correo electrónico formateado:

```
aws ses send-email --from sender@example.com --destination file://destination.json
--message file://message.json
```

Salida:

```
{
  "MessageId": "EXAMPLEf3a5efcd1-51adec81-d2a4-4e3f-9fe2-5d85c1b23783-000000"
}
```

El destino y el mensaje son estructuras de JSON datos guardadas en archivos.json en el directorio actual. Estos archivos son los siguientes:

`destination.json`:

```
{
  "ToAddresses": ["recipient1@example.com", "recipient2@example.com"],
  "CcAddresses": ["recipient3@example.com"],
  "BccAddresses": []
}
```

`message.json`:

```
{
  "Subject": {
    "Data": "Test email sent using the AWS CLI",
    "Charset": "UTF-8"
  },
  "Body": {
    "Text": {
      "Data": "This is the message body in text format.",
      "Charset": "UTF-8"
    },
    "Html": {
      "Data": "This message body contains HTML formatting. It can, for example,
contain links like this one: <a class=\"ulink\" href=\"http://docs.aws.amazon.com/
ses/latest/DeveloperGuide\" target=\"_blank\">Amazon SES Developer Guide</a>.",
      "Charset": "UTF-8"
    }
  }
}
```

```
}  
}
```

Sustituya las direcciones de correo electrónico del remitente y del destinatario por las que desee utilizar. Ten en cuenta que la dirección de correo electrónico del remitente debe estar verificada con AmazonSES. Hasta que se te conceda el acceso de producción a AmazonSES, también debes verificar la dirección de correo electrónico de cada destinatario, a menos que el destinatario sea el simulador de SES buzones de Amazon. Para obtener más información sobre la verificación, consulte Verificación de direcciones de correo electrónico y dominios en Amazon SES en la Guía para desarrolladores de Amazon Simple Email Service.

El ID del mensaje en el resultado indica que la llamada para enviar el correo electrónico se ha realizado correctamente.

Si no recibe el correo electrónico, marque la casilla de correo no deseado.

Para obtener más información sobre el envío de correo electrónico formateado, consulte Envío de correo electrónico formateado con Amazon SES API en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte la Referencia [SendEmail](#) de AWS CLI comandos.

## **send-raw-email**

En el siguiente ejemplo de código, se muestra cómo usar `send-raw-email`.

### AWS CLI

Para enviar un correo electrónico sin procesar mediante Amazon SES

En el siguiente ejemplo, se utiliza el `send-raw-email` comando para enviar un correo electrónico con un TXT archivo adjunto:

```
aws ses send-raw-email --raw-message file://message.json
```

Salida:

```
{  
  "MessageId": "EXAMPLEf3f73d99b-c63fb06f-d263-41f8-a0fb-d0dc67d56c07-000000"
```

```
}
```

El mensaje sin procesar es una estructura de JSON datos guardada en un archivo denominado `message.json` en el directorio actual. Contiene lo siguiente:

```
{
  "Data": "From: sender@example.com\nTo: recipient@example.com\nSubject: Test email
sent using the AWS CLI (contains an attachment)\nMIME-Version: 1.0\nContent-type:
Multipart/Mixed; boundary=\"NextPart\"\n\n--NextPart\nContent-Type: text/plain
\n\nThis is the message body.\n\n--NextPart\nContent-Type: text/plain;\nContent-
Disposition: attachment; filename=\"attachment.txt\"\n\nThis is the text in the
attachment.\n\n--NextPart--"
}
```

Como puede ver, «Datos» es una cadena larga que contiene todo el contenido sin procesar del correo electrónico en MIME formato, incluido un archivo adjunto denominado `attachment.txt`.

Sustituya `sender@example.com` y `recipient@example.com` por las direcciones que desee utilizar. Ten en cuenta que la dirección de correo electrónico del remitente debe estar verificada con AmazonSES. Hasta que se te conceda el acceso de producción a AmazonSES, también debes verificar la dirección de correo electrónico del destinatario, a menos que el destinatario sea el simulador de SES buzones de Amazon. Para obtener más información sobre la verificación, consulte [Verificación de direcciones de correo electrónico y dominios en Amazon SES](#) en la Guía para desarrolladores de Amazon Simple Email Service.

El identificador del mensaje que aparece en el resultado indica que la llamada a `send-raw-email` se ha realizado correctamente.

Si no recibe el correo electrónico, marque la casilla de correo no deseado.

Para obtener más información sobre el envío de correo electrónico sin procesar, consulte [Envío de correo electrónico sin procesar con Amazon SES API](#) en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [SendRawEmail](#) la Referencia de AWS CLI comandos.

## **set-identity-dkim-enabled**

En el siguiente ejemplo de código, se muestra cómo usar `set-identity-dkim-enabled`.

## AWS CLI

Para activar o desactivar Easy DKIM for a Amazon SES Verified Identity

En el siguiente ejemplo, se utiliza el `set-identity-dkim-enabled` comando DKIM para deshabilitar una dirección de correo electrónico verificada:

```
aws ses set-identity-dkim-enabled --identity user@example.com --no-dkim-enabled
```

Para obtener más información sobre EasyDKIM, consulta Easy DKIM in Amazon SES en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [SetIdentityDkimEnabled](#) la Referencia de AWS CLI comandos.

## `set-identity-feedback-forwarding-enabled`

En el siguiente ejemplo de código, se muestra cómo usar `set-identity-feedback-forwarding-enabled`.

## AWS CLI

Para activar o desactivar el rebote y el envío de comentarios por correo electrónico para una identidad verificada de Amazon SES

En el siguiente ejemplo, se utiliza el `set-identity-feedback-forwarding-enabled` comando para permitir que una dirección de correo electrónico verificada reciba notificaciones de devoluciones y quejas por correo electrónico:

```
aws ses set-identity-feedback-forwarding-enabled --identity user@example.com --forwarding-enabled
```

Debes recibir las notificaciones de devoluciones y quejas a través de Amazon SNS o mediante el reenvío de comentarios por correo electrónico, por lo que solo puedes deshabilitar el reenvío de comentarios por correo electrónico si seleccionas un SNS tema de Amazon para las notificaciones de rebote y queja.

Para obtener más información sobre las notificaciones, consulte Uso de notificaciones con Amazon SES en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [SetIdentityFeedbackForwardingEnabled](#) la Referencia de AWS CLI comandos.

## set-identity-notification-topic

En el siguiente ejemplo de código, se muestra cómo usar `set-identity-notification-topic`.

### AWS CLI

Para configurar el SNS tema de Amazon en el que Amazon SES publicará las notificaciones de rebote, queja o entrega de una identidad verificada

En el siguiente ejemplo, se utiliza el `set-identity-notification-topic` comando para especificar el SNS tema de Amazon en el que una dirección de correo electrónico verificada recibirá las notificaciones de rebote:

```
aws ses set-identity-notification-topic --identity user@example.com --notification-type Bounce --sns-topic arn:aws:sns:us-east-1:EXAMPLE65304:MyTopic
```

Para obtener más información sobre las notificaciones, consulte [Uso de notificaciones con Amazon SES](#) en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [SetIdentityNotificationTopic](#) la Referencia de AWS CLI comandos.

## verify-domain-dkim

En el siguiente ejemplo de código, se muestra cómo usar `verify-domain-dkim`.

### AWS CLI

Para generar los DKIM tokens de un dominio verificado para DKIM firmar con Amazon SES

En el siguiente ejemplo, se utiliza el `verify-domain-dkim` comando para generar DKIM tokens para un dominio que se ha verificado con AmazonSES:

```
aws ses verify-domain-dkim --domain example.com
```

Salida:

```
{
  "DkimTokens": [
    "EXAMPLEq76owjnks3lnluwg65scbemvw",
    "EXAMPLEi3dnsj67hstzaj673klariwx2",
    "EXAMPLEwfbtcukvimehexktdmtaz6naj"
```

```
]
}
```

Para configurarloDKIM, debes usar los DKIM tokens devueltos para actualizar la DNS configuración de tu dominio con CNAME registros que apunten a claves DKIM públicas alojadas por AmazonSES. Para obtener más información, consulta Easy DKIM in Amazon SES en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [VerifyDomainDkim](#) la Referencia de AWS CLI comandos.

## **verify-domain-identity**

En el siguiente ejemplo de código, se muestra cómo usar `verify-domain-identity`.

### AWS CLI

Para verificar un dominio con Amazon SES

En el siguiente ejemplo, se usa el comando `verify-domain-identity` para verificar un dominio:

```
aws ses verify-domain-identity --domain example.com
```

Salida:

```
{
  "VerificationToken": "eoEmxw+YaYhb3h3iVJHuXMJXqeu1q1/wmvjuEXAMPLE"
}
```

Para completar la verificación del dominio, debes añadir un TXT registro con el token de verificación devuelto a la DNS configuración de tu dominio. Para obtener más información, consulte Verificación de dominios en Amazon SES en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [VerifyDomainIdentity](#) la Referencia de AWS CLI comandos.

## **verify-email-identity**

En el siguiente ejemplo de código, se muestra cómo usar `verify-email-identity`.

## AWS CLI

Para verificar una dirección de correo electrónico con Amazon SES

En el siguiente ejemplo, se usa el comando `verify-email-identity` para verificar una dirección de correo electrónico:

```
aws ses verify-email-identity --email-address user@example.com
```

Antes de poder enviar un correo electrónico a través de AmazonSES, debes verificar la dirección o el dominio desde el que lo envías para demostrar que eres el propietario. Si aún no tienes acceso a producción, también debes verificar todas las direcciones de correo electrónico a las que envíes correos electrónicos, excepto las direcciones de correo electrónico proporcionadas por el simulador de SES buzones de Amazon.

Cuando `verify-email-identity` la llames, la dirección de correo electrónico recibirá un correo electrónico de verificación. El usuario debe hacer clic en el enlace del correo electrónico para completar el proceso de verificación.

Para obtener más información, consulte [Verificación de direcciones de correo electrónico en Amazon SES](#) en la Guía para desarrolladores de Amazon Simple Email Service.

- Para API obtener más información, consulte [VerifyEmailIdentity](#) la Referencia de AWS CLI comandos.

## Ejemplos de Shield usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante el uso de AWS Command Line Interface with Shield.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde encontrará instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)



## Acciones

### **associate-drt-log-bucket**

En el siguiente ejemplo de código, se muestra cómo usar `associate-drt-log-bucket`.

#### AWS CLI

Para autorizar el acceso DRT a un bucket de Amazon S3

En el siguiente `associate-drt-log-bucket` ejemplo, se crea una asociación entre el bucket de S3 especificado DRT y el depósito de S3 especificado. Esto le permite acceder DRT al depósito en nombre de la cuenta. :

```
aws shield associate-drt-log-bucket \  
  --log-bucket flow-logs-for-website-lb
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Autorizar al equipo de DDoS respuesta](#) en la Guía para desarrolladores de AWS Shield Advanced.

- Para API obtener más información, consulte [AssociateDrtLogBucket](#) la Referencia de AWS CLI comandos.

### **associate-drt-role**

En el siguiente ejemplo de código, se muestra cómo usar `associate-drt-role`.

#### AWS CLI

Para autorizar la DRT mitigación de posibles ataques en su nombre

En el siguiente `associate-drt-role` ejemplo, se crea una asociación entre la función especificada DRT y la función especificada. DRT Pueden usar el rol para acceder a la cuenta y administrarla.

```
aws shield associate-drt-role \  
  --role-arn arn:aws:iam::123456789012:role/service-role/DrtRole
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Autorizar al equipo de DDoS respuesta](#) en la Guía para desarrolladores de AWS Shield Advanced.

- Para API obtener más información, consulte [AssociateDrtRole](#) la Referencia de AWS CLI comandos.

## create-protection

En el siguiente ejemplo de código, se muestra cómo usar `create-protection`.

### AWS CLI

Para habilitar la protección AWS Shield Advanced para un solo AWS recurso

El siguiente `create-protection` ejemplo habilita la protección Shield Advanced para la AWS CloudFront distribución especificada.

```
aws shield create-protection \  
  --name "Protection for CloudFront distribution" \  
  --resource-arn arn:aws:cloudfront::123456789012:distribution/E198WC25FX0WY8
```

Salida:

```
{  
  "ProtectionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
}
```

Para obtener más información, consulte la Guía avanzada [para desarrolladores de Specify Your Resources to Protect](#) in the AWS Shield.

- Para API obtener más información, consulte [CreateProtection](#) la Referencia de AWS CLI comandos.

## create-subscription

En el siguiente ejemplo de código, se muestra cómo usar `create-subscription`.

### AWS CLI

Para habilitar la protección AWS Shield Advanced para una cuenta

El siguiente `create-subscription` ejemplo habilita la protección Shield Advanced para la cuenta.

```
aws shield create-subscription
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Introducción a AWS Shield Advanced](#) en la Guía para desarrolladores de AWS Shield Advanced.

- Para API obtener más información, consulte [CreateSubscription](#) la Referencia de AWS CLI comandos.

## **delete-protection**

En el siguiente ejemplo de código, se muestra cómo usar `delete-protection`.

### AWS CLI

Para eliminar la protección AWS Shield Advanced de un AWS recurso

El siguiente `delete-protection` ejemplo elimina la protección AWS Shield Advanced especificada.

```
aws shield delete-protection \  
  --protection-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Eliminar AWS Shield Advanced de un AWS recurso](#) en la Guía para desarrolladores de AWS Shield Advanced.

- Para API obtener más información, consulte [DeleteProtection](#) la Referencia de AWS CLI comandos.

## **describe-attack**

En el siguiente ejemplo de código, se muestra cómo usar `describe-attack`.

### AWS CLI

Para recuperar una descripción detallada de un ataque

En el siguiente describe-attack ejemplo, se muestran detalles sobre el DDoS ataque con el identificador de ataque especificado. Puede obtener un ataque IDs ejecutando el list-attacks comando.

```
aws shield describe-attack --attack-id a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Salida:

```
{
  "Attack": {
    "AttackId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "ResourceArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/testElb",
    "SubResources": [
      {
        "Type": "IP",
        "Id": "192.0.2.2",
        "AttackVectors": [
          {
            "VectorType": "SYN_FLOOD",
            "VectorCounters": [
              {
                "Name": "SYN_FLOOD_BPS",
                "Max": 982184.0,
                "Average": 982184.0,
                "Sum": 11786208.0,
                "N": 12,
                "Unit": "BPS"
              }
            ]
          }
        ]
      },
      {
        "Type": "IP",
        "Id": "192.0.2.3",
        "AttackVectors": [
          {
            "VectorType": "SYN_FLOOD",
            "VectorCounters": [
              {
                "Name": "SYN_FLOOD_BPS",
```

```
        "Max": 982184.0,
        "Average": 982184.0,
        "Sum": 9821840.0,
        "N": 10,
        "Unit": "BPS"
      }
    ]
  },
  "Counters": []
},
{
  "Type": "IP",
  "Id": "192.0.2.4",
  "AttackVectors": [
    {
      "VectorType": "SYN_FLOOD",
      "VectorCounters": [
        {
          "Name": "SYN_FLOOD_BPS",
          "Max": 982184.0,
          "Average": 982184.0,
          "Sum": 7857472.0,
          "N": 8,
          "Unit": "BPS"
        }
      ]
    }
  ]
},
  "Counters": []
},
{
  "Type": "IP",
  "Id": "192.0.2.5",
  "AttackVectors": [
    {
      "VectorType": "SYN_FLOOD",
      "VectorCounters": [
        {
          "Name": "SYN_FLOOD_BPS",
          "Max": 982184.0,
          "Average": 982184.0,
          "Sum": 1964368.0,
          "N": 2,
```

```
        "Unit": "BPS"
      }
    ]
  },
  "Counters": []
},
{
  "Type": "IP",
  "Id": "2001:DB8::bcde:4321:8765:0:0",
  "AttackVectors": [
    {
      "VectorType": "SYN_FLOOD",
      "VectorCounters": [
        {
          "Name": "SYN_FLOOD_BPS",
          "Max": 982184.0,
          "Average": 982184.0,
          "Sum": 1964368.0,
          "N": 2,
          "Unit": "BPS"
        }
      ]
    }
  ]
},
{
  "Type": "IP",
  "Id": "192.0.2.6",
  "AttackVectors": [
    {
      "VectorType": "SYN_FLOOD",
      "VectorCounters": [
        {
          "Name": "SYN_FLOOD_BPS",
          "Max": 982184.0,
          "Average": 982184.0,
          "Sum": 1964368.0,
          "N": 2,
          "Unit": "BPS"
        }
      ]
    }
  ]
}
```

```
    ],
    "Counters": []
  }
],
"StartTime": 1576024927.457,
"EndTime": 1576025647.457,
"AttackCounters": [],
"AttackProperties": [
  {
    "AttackLayer": "NETWORK",
    "AttackPropertyIdentifier": "SOURCE_IP_ADDRESS",
    "TopContributors": [
      {
        "Name": "198.51.100.5",
        "Value": 2024475682
      },
      {
        "Name": "198.51.100.8",
        "Value": 1311380863
      },
      {
        "Name": "203.0.113.4",
        "Value": 900599855
      },
      {
        "Name": "198.51.100.4",
        "Value": 769417366
      },
      {
        "Name": "203.1.113.13",
        "Value": 757992847
      }
    ]
  },
  {
    "AttackLayer": "NETWORK",
    "AttackPropertyIdentifier": "SOURCE_COUNTRY",
    "TopContributors": [
      {
        "Name": "United States",
        "Value": 80938161764
      }
    ]
  }
],
```

```
    {
      "Name": "Brazil",
      "Value": 9929864330
    },
    {
      "Name": "Netherlands",
      "Value": 1635009446
    },
    {
      "Name": "Mexico",
      "Value": 144832971
    },
    {
      "Name": "Japan",
      "Value": 45369000
    }
  ],
  "Unit": "BYTES",
  "Total": 92773354841
},
{
  "AttackLayer": "NETWORK",
  "AttackPropertyIdentifier": "SOURCE_ASN",
  "TopContributors": [
    {
      "Name": "12345",
      "Value": 74953625841
    },
    {
      "Name": "12346",
      "Value": 4440087595
    },
    {
      "Name": "12347",
      "Value": 1635009446
    },
    {
      "Name": "12348",
      "Value": 1221230000
    },
    {
      "Name": "12349",
      "Value": 1199425294
    }
  ]
}
```



```

        ],
        "Unit": "BYTES",
        "Total": 92755479921
    }
  ],
  "Mitigations": []
}

```

Para obtener más información, consulte [Revisión de DDoS incidentes](#) en la Guía para desarrolladores de AWS Shield Advanced.

- Para API obtener más información, consulte [DescribeAttack](#) la Referencia de AWS CLI comandos.

## **describe-drt-access**

En el siguiente ejemplo de código, se muestra cómo usar `describe-drt-access`.

### AWS CLI

Para obtener una descripción de las autorizaciones que DRT tiene para mitigar los ataques en su nombre

En el siguiente `describe-drt-access` ejemplo, se recuperan las autorizaciones de rol y bucket de S3 que DRT tiene, lo que le permite responder a posibles ataques en su nombre.

```
aws shield describe-drt-access
```

Salida:

```

{
  "RoleArn": "arn:aws:iam::123456789012:role/service-role/DrtRole",
  "LogBucketList": [
    "flow-logs-for-website-lb"
  ]
}

```

Para obtener más información, consulte [Autorizar al equipo de DDoS respuesta](#) en la Guía para desarrolladores de AWS Shield Advanced.

- Para API obtener más información, consulte [DescribeDrtAccess](#) la Referencia de AWS CLI comandos.

## **describe-emergency-contact-settings**

En el siguiente ejemplo de código, se muestra cómo usar `describe-emergency-contact-settings`.

### AWS CLI

Para recuperar las direcciones de correo electrónico de emergencia que tenga archivadas en el DRT

En el siguiente `describe-emergency-contact-settings` ejemplo, se recuperan las direcciones de correo electrónico archivadas en DRT la cuenta. Estas son las direcciones con las que DRT debe ponerse en contacto cuando se trata de responder a un presunto ataque.

```
aws shield describe-emergency-contact-settings
```

Salida:

```
{
  "EmergencyContactList": [
    {
      "EmailAddress": "ops@example.com"
    },
    {
      "EmailAddress": "ddos-notifications@example.com"
    }
  ]
}
```

Para obtener más información, consulte [Cómo funciona AWS Shield < https://docs.aws.amazon.com/waf/latest/developerguide/ddos-overview.html >](https://docs.aws.amazon.com/waf/latest/developerguide/ddos-overview.html) en la Guía avanzada para desarrolladores de Shield.AWS

- Para obtener API más información, [DescribeEmergencyContactSettings](#) consulte AWS CLI la Referencia de comandos.

## describe-protection

En el siguiente ejemplo de código, se muestra cómo usar `describe-protection`.

### AWS CLI

Para recuperar los detalles de una protección AWS Shield Advanced

El siguiente `describe-protection` ejemplo muestra detalles sobre la protección Shield Advanced con el ID especificado. Puede obtener protección IDs ejecutando el `list-protections` comando.

```
aws shield describe-protection \  
  --protection-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{  
  "Protection": {  
    "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
    "Name": "1.2.3.4",  
    "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:eip-allocation/  
eipalloc-0ac1537af40742a6d"  
  }  
}
```

Para obtener más información, consulte la Guía avanzada [para desarrolladores de Specify Your Resources to Protect](#) in the AWS Shield.

- Para API obtener más información, consulte [DescribeProtection](#) la Referencia de AWS CLI comandos.

## describe-subscription

En el siguiente ejemplo de código, se muestra cómo usar `describe-subscription`.

### AWS CLI

Para recuperar los detalles de la protección AWS Shield Advanced de la cuenta

En el siguiente `describe-subscription` ejemplo, se muestran detalles sobre la protección Shield Advanced proporcionada a la cuenta. :

**aws shield describe-subscription**

Salida:

```
{
  "Subscription": {
    "StartTime": 1534368978.0,
    "EndTime": 1597613778.0,
    "TimeCommitmentInSeconds": 63244800,
    "AutoRenew": "ENABLED",
    "Limits": [
      {
        "Type": "GLOBAL_ACCELERATOR",
        "Max": 1000
      },
      {
        "Type": "ROUTE53_HOSTED_ZONE",
        "Max": 1000
      },
      {
        "Type": "CF_DISTRIBUTION",
        "Max": 1000
      },
      {
        "Type": "ELB_LOAD_BALANCER",
        "Max": 1000
      },
      {
        "Type": "EC2_ELASTIC_IP_ALLOCATION",
        "Max": 1000
      }
    ]
  }
}
```

Para obtener más información, consulte [Cómo funciona AWS Shield](#) en la Guía avanzada para desarrolladores de AWS Shield.

- Para API obtener más información, consulte [DescribeSubscription](#) la Referencia de AWS CLI comandos.

## disassociate-drt-log-bucket

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-drt-log-bucket`.

### AWS CLI

Para eliminar la autorización para acceder DRT a un bucket de Amazon S3 en su nombre

El siguiente `disassociate-drt-log-bucket` ejemplo elimina la asociación entre el bucket de S3 DRT y el especificado. Una vez completado este comando, ya no DRT podrá acceder al depósito en nombre de la cuenta.

```
aws shield disassociate-drt-log-bucket \  
--log-bucket flow-logs-for-website-lb
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Autorizar al equipo de DDoS respuesta](#) en la Guía para desarrolladores de AWS Shield Advanced.

- Para API obtener más información, consulte [DisassociateDrtLogBucket](#) la Referencia de AWS CLI comandos.

## disassociate-drt-role

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-drt-role`.

### AWS CLI

Para eliminar la autorización DRT para mitigar posibles ataques en su nombre

El siguiente `disassociate-drt-role` ejemplo elimina la asociación entre la cuenta DRT y la cuenta. Tras esta llamada, ya no DRT podrá acceder a su cuenta ni administrarla.

```
aws shield disassociate-drt-role
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Autorizar al equipo de DDoS respuesta](#) en la Guía para desarrolladores de AWS Shield Advanced.

- Para API obtener más información, consulte [DisassociateDrtRole](#) la Referencia de AWS CLI comandos.

## get-subscription-state

En el siguiente ejemplo de código, se muestra cómo usar `get-subscription-state`.

### AWS CLI

Para recuperar el estado actual de la suscripción a AWS Shield Advanced de la cuenta

El siguiente `get-subscription-state` ejemplo recupera el estado de la protección Shield Advanced de la cuenta.

```
aws shield get-subscription-state
```

Salida:

```
{
  "SubscriptionState": "ACTIVE"
}
```

Para obtener más información, consulte [Cómo funciona AWS Shield](#) en la Guía avanzada para desarrolladores de AWS Shield.

- Para API obtener más información, consulte [GetSubscriptionState](#) la Referencia de AWS CLI comandos.

## list-attacks

En el siguiente ejemplo de código, se muestra cómo usar `list-attacks`.

### AWS CLI

Para recuperar los resúmenes de los ataques de AWS Shield Advanced

El siguiente `list-attacks` ejemplo recupera resúmenes de los ataques de la AWS CloudFront distribución especificada durante el período de tiempo especificado. La respuesta incluye un ataque IDs que puede proporcionar al `describe-attack` comando para obtener información detallada sobre un ataque.

```
aws shield list-attacks \
  --resource-arns arn:aws:cloudfront::12345678910:distribution/E1PXMP22ZVFA0R \
  --start-time FromInclusive=1529280000,ToExclusive=1529300000
```

Salida:

```
{
  "AttackSummaries": [
    {
      "AttackId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "ResourceArn": "arn:aws:cloudfront::123456789012:distribution/
E1PXM22ZVFAOR",
      "StartTime": 1529280000.0,
      "EndTime": 1529449200.0,
      "AttackVectors": [
        {
          "VectorType": "SYN_FLOOD"
        }
      ]
    }
  ]
}
```

Para obtener más información, consulte [Revisión de DDoS incidentes](#) en la Guía para desarrolladores de AWS Shield Advanced.

- Para API obtener más información, consulte [ListAttacks](#) la Referencia de AWS CLI comandos.

## list-protections

En el siguiente ejemplo de código, se muestra cómo usar `list-protections`.

AWS CLI

Para recuperar los resúmenes de protección de AWS Shield Advanced

El siguiente `list-protections` ejemplo recupera resúmenes de las protecciones que están habilitadas para la cuenta.

```
aws shield list-protections
```

Salida:

```
{
  "Protections": [
    {
```

```
    "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Name": "Protection for CloudFront distribution",
    "ResourceArn": "arn:aws:cloudfront::123456789012:distribution/
E198WC25FX0WY8"
  }
]
```

Para obtener más información, consulte la Guía avanzada [para desarrolladores de Specify Your Resources to Protect](#) in the AWS Shield.

- Para API obtener más información, consulte [ListProtections](#) la Referencia de AWS CLI comandos.

## update-emergency-contact-settings

En el siguiente ejemplo de código, se muestra cómo usar `update-emergency-contact-settings`.

### AWS CLI

Para definir las direcciones de correo electrónico de emergencia archivadas en el DRT

En el siguiente `update-emergency-contact-settings` ejemplo, se definen dos direcciones de correo electrónico con las que DRT debe ponerse en contacto cuando se trata de responder a un presunto ataque.

```
aws shield update-emergency-contact-settings \
  --emergency-contact-list EmailAddress=ops@example.com EmailAddress=ddos-
notifications@example.com
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cómo funciona AWS Shield](#) en la Guía avanzada para desarrolladores de AWS Shield.

- Para API obtener más información, consulte [UpdateEmergencyContactSettings](#) la Referencia de AWS CLI comandos.

## update-subscription

En el siguiente ejemplo de código, se muestra cómo usar `update-subscription`.



## AWS CLI

Para modificar la suscripción a AWS Shield Advanced de la cuenta

El siguiente `update-subscription` ejemplo permite la renovación automática de la suscripción a AWS Shield Advanced de la cuenta.

```
aws shield update-subscription \  
  --auto-renew ENABLED
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Cómo funciona AWS Shield](#) en la Guía avanzada para desarrolladores de AWS Shield.

- Para API obtener más información, consulte [UpdateSubscription](#) la Referencia de AWS CLI comandos.

## Ejemplos de firmantes que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface with Signer.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **cancel-signing-profile**

En el siguiente ejemplo de código, se muestra cómo usar `cancel-signing-profile`.

## AWS CLI

Para eliminar un perfil de firma

En el siguiente `cancel-signing-profile` ejemplo, se elimina un perfil de firma existente de AWS Signer.

```
aws signer cancel-signing-profile \  
  --profile-name MyProfile1
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [CancelSigningProfile](#) la Referencia de AWS CLI comandos.

## describe-signing-job

En el siguiente ejemplo de código, se muestra cómo usar `describe-signing-job`.

## AWS CLI

Para mostrar detalles sobre un trabajo de firma

En el siguiente `describe-signing-job` ejemplo, se muestran detalles sobre el trabajo de firma especificado.

```
aws signer describe-signing-job \  
  --job-id 2065c468-73e2-4385-a6c9-0123456789abc
```

Salida:

```
{  
  "status": "Succeeded",  
  "completedAt": 1568412037,  
  "platformId": "AmazonFreeRTOS-Default",  
  "signingMaterial": {  
    "certificateArn": "arn:aws:acm:us-  
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"  
  },  
  "statusReason": "Signing Succeeded",  
  "jobId": "2065c468-73e2-4385-a6c9-0123456789abc",
```

```

"source": {
  "s3": {
    "version": "PNyFaUTgsQh5ZdMCcoCe6pT1g0pgB_M4",
    "bucketName": "signer-source",
    "key": "MyCode.rb"
  }
},
"profileName": "MyProfile2",
"signedObject": {
  "s3": {
    "bucketName": "signer-destination",
    "key": "signed-2065c468-73e2-4385-a6c9-0123456789abc"
  }
},
"requestedBy": "arn:aws:iam::123456789012:user/maria",
"createdAt": 1568412036
}

```

- Para API obtener más información, consulte [DescribeSigningJob](#) la Referencia de AWS CLI comandos.

## get-signing-platform

En el siguiente ejemplo de código, se muestra cómo usar `get-signing-platform`.

### AWS CLI

Para mostrar detalles sobre una plataforma de firma

En el siguiente `get-signing-platform` ejemplo, se muestran detalles sobre la plataforma de firma especificada.

```

aws signer get-signing-platform \
  --platform-id AmazonFreeRTOS-TI-CC3220SF

```

Salida:

```

{
  "category": "AWS",
  "displayName": "Amazon FreeRTOS SHA1-RSA CC3220SF-Format",
  "target": "SHA1-RSA-TISHA1",
  "platformId": "AmazonFreeRTOS-TI-CC3220SF",

```

```

"signingConfiguration": {
  "encryptionAlgorithmOptions": {
    "defaultValue": "RSA",
    "allowedValues": [
      "RSA"
    ]
  },
  "hashAlgorithmOptions": {
    "defaultValue": "SHA1",
    "allowedValues": [
      "SHA1"
    ]
  }
},
"maxSizeInMB": 16,
"partner": "AmazonFreeRTOS",
"signingImageFormat": {
  "defaultFormat": "JSONEmbedded",
  "supportedFormats": [
    "JSONEmbedded"
  ]
}
}

```

- Para API obtener más información, consulte [GetSigningPlatform](#) la Referencia de AWS CLI comandos.

## get-signing-profile

En el siguiente ejemplo de código, se muestra cómo usar `get-signing-profile`.

### AWS CLI

Para mostrar detalles sobre un perfil de firma

En el siguiente `get-signing-profile` ejemplo, se muestran detalles sobre el perfil de firma especificado.

```

aws signer get-signing-profile \
  --profile-name MyProfile3

```

Salida:

```
{
  "platformId": "AmazonFreeRTOS-TI-CC3220SF",
  "profileName": "MyProfile3",
  "status": "Active",
  "signingMaterial": {
    "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
  }
}
```

- Para API obtener más información, consulte [GetSigningProfile](#) la Referencia de AWS CLI comandos.

## list-signing-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-signing-jobs`.

### AWS CLI

Para ver una lista de todos los trabajos de firma

En el siguiente `list-signing-jobs` ejemplo, se muestran detalles sobre todos los trabajos de firma de la cuenta.

```
aws signer list-signing-jobs
```

En este ejemplo, se devuelven dos trabajos, uno correcto y otro fallido.

```
{
  "jobs": [
    {
      "status": "Succeeded",
      "signingMaterial": {
        "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
      },
      "jobId": "2065c468-73e2-4385-a6c9-0123456789abc",
      "source": {
        "s3": {
          "version": "PNyFaUTgsQh5ZdMCcoCe6pT1g0pgB_M4",
          "bucketName": "signer-source",

```

```

        "key": "MyCode.rb"
      }
    },
    "signedObject": {
      "s3": {
        "bucketName": "signer-destination",
        "key": "signed-2065c468-73e2-4385-a6c9-0123456789abc"
      }
    },
    "createdAt": 1568412036
  },
  {
    "status": "Failed",
    "source": {
      "s3": {
        "version": "PNyFaUTgsQh5ZdMCcoCe6pT1g0pgB_M4",
        "bucketName": "signer-source",
        "key": "MyOtherCode.rb"
      }
    },
    "signingMaterial": {
      "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
    },
    "createdAt": 1568402690,
    "jobId": "74d9825e-22fc-4a0d-b962-0123456789abc"
  }
]
}

```

- Para API obtener más información, consulte [ListSigningJobs](#) la Referencia de AWS CLI comandos.

## list-signing-platforms

En el siguiente ejemplo de código, se muestra cómo usar `list-signing-platforms`.

### AWS CLI

Para ver una lista de todas las plataformas de firma

En el siguiente `list-signing-platforms` ejemplo, se muestran detalles sobre todas las plataformas de firma disponibles.

**aws signer list-signing-platforms**

Salida:

```
{
  "platforms": [
    {
      "category": "AWS",
      "displayName": "AWS IoT Device Management SHA256-ECDSA ",
      "target": "SHA256-ECDSA",
      "platformId": "AWSIoTDeviceManagement-SHA256-ECDSA",
      "signingConfiguration": {
        "encryptionAlgorithmOptions": {
          "defaultValue": "ECDSA",
          "allowedValues": [
            "ECDSA"
          ]
        },
        "hashAlgorithmOptions": {
          "defaultValue": "SHA256",
          "allowedValues": [
            "SHA256"
          ]
        }
      },
      "maxSizeInMB": 2048,
      "partner": "AWSIoTDeviceManagement",
      "signingImageFormat": {
        "defaultFormat": "JSONDetached",
        "supportedFormats": [
          "JSONDetached"
        ]
      }
    },
    {
      "category": "AWS",
      "displayName": "Amazon FreeRTOS SHA1-RSA CC3220SF-Format",
      "target": "SHA1-RSA-TISHA1",
      "platformId": "AmazonFreeRTOS-TI-CC3220SF",
      "signingConfiguration": {
        "encryptionAlgorithmOptions": {
          "defaultValue": "RSA",
          "allowedValues": [
```

```

        "RSA"
      ]
    },
    "hashAlgorithmOptions": {
      "defaultValue": "SHA1",
      "allowedValues": [
        "SHA1"
      ]
    }
  },
  "maxSizeInMB": 16,
  "partner": "AmazonFreeRTOS",
  "signingImageFormat": {
    "defaultFormat": "JSONEmbedded",
    "supportedFormats": [
      "JSONEmbedded"
    ]
  }
},
{
  "category": "AWS",
  "displayName": "Amazon FreeRTOS SHA256-ECDSA",
  "target": "SHA256-ECDSA",
  "platformId": "AmazonFreeRTOS-Default",
  "signingConfiguration": {
    "encryptionAlgorithmOptions": {
      "defaultValue": "ECDSA",
      "allowedValues": [
        "ECDSA"
      ]
    },
    "hashAlgorithmOptions": {
      "defaultValue": "SHA256",
      "allowedValues": [
        "SHA256"
      ]
    }
  },
  "maxSizeInMB": 16,
  "partner": "AmazonFreeRTOS",
  "signingImageFormat": {
    "defaultFormat": "JSONEmbedded",
    "supportedFormats": [
      "JSONEmbedded"
    ]
  }
}

```



```

    ]
  }
]
}

```

- Para API obtener más información, consulte [ListSigningPlatforms](#) la Referencia de AWS CLI comandos.

## list-signing-profiles

En el siguiente ejemplo de código, se muestra cómo usar `list-signing-profiles`.

### AWS CLI

Para ver una lista de todos los perfiles de firma

En el siguiente `list-signing-profiles` ejemplo, se muestran detalles sobre todos los perfiles de firma de la cuenta.

```
aws signer list-signing-profiles
```

Salida:

```

{
  "profiles": [
    {
      "platformId": "AmazonFreeRTOS-TI-CC3220SF",
      "profileName": "MyProfile4",
      "status": "Active",
      "signingMaterial": {
        "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
      }
    },
    {
      "platformId": "AWSIoTDeviceManagement-SHA256-ECDSA",
      "profileName": "MyProfile5",
      "status": "Active",
      "signingMaterial": {
        "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
      }
    }
  ]
}

```

```

    }
  }
]
}

```

- Para API obtener más información, consulte [ListSigningProfiles](#) la Referencia de AWS CLI comandos.

## put-signing-profile

En el siguiente ejemplo de código, se muestra cómo usar `put-signing-profile`.

### AWS CLI

Para crear un perfil de firma

En el siguiente `put-signing-profile` ejemplo, se crea un perfil de firma con el certificado y la plataforma especificados.

```

aws signer put-signing-profile \
  --profile-name MyProfile6 \
  --signing-material certificateArn=arn:aws:acm:us-west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc \
  --platform AmazonFreeRTOS-TI-CC3220SF

```

Salida:

```

{
  "arn": "arn:aws:signer:us-west-2:123456789012:/signing-profiles/MyProfile6"
}

```

- Para API obtener más información, consulte [PutSigningProfile](#) la Referencia de AWS CLI comandos.

## start-signing-job

En el siguiente ejemplo de código, se muestra cómo usar `start-signing-job`.

### AWS CLI

Para iniciar un trabajo de firma

En el siguiente `start-signing-job` ejemplo, se inicia un trabajo de firma en el código que se encuentra en la fuente especificada. Utiliza el perfil especificado para realizar la firma y coloca el código firmado en el destino especificado.

```
aws signer start-signing-job \
  --source 's3={bucketName=signer-
source,key=MyCode.rb,version=PNyFaUTgsQh5ZdMCcoCe6pT1g0pgB_M4}' \
  --destination 's3={bucketName=signer-destination,prefix=signed-}' \
  --profile-name MyProfile7
```

El resultado es el ID del trabajo de firma.

```
{
  "jobId": "2065c468-73e2-4385-a6c9-0123456789abc"
}
```

- Para API obtener más información, consulte [StartSigningJob](#) la Referencia de AWS CLI comandos.

## Ejemplos de Snowball usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Snowball.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde encontrará instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

### Acciones

#### **get-snowball-usage**

En el siguiente ejemplo de código, se muestra cómo usar `get-snowball-usage`.

## AWS CLI

Para obtener información sobre el límite del servicio Snowball de su cuenta

El siguiente `get-snowball-usage` ejemplo muestra información sobre el límite del servicio Snowball de su cuenta y también el número de Snowballs que utiliza su cuenta.

```
aws snowball get-snowball-usage
```

Salida:

```
{
  "SnowballLimit": 1,
  "SnowballsInUse": 0
}
```

Para obtener más información, consulte los [límites de AWS Snowball Edge](#) en la Guía del desarrollador de AWS Snowball.

- Para API obtener más información, consulte la Referencia [GetSnowballUsage](#) de AWS CLI comandos.

## list-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-jobs`.

## AWS CLI

Para mostrar los trabajos actuales de Snowball en su cuenta

En el siguiente `list-jobs` ejemplo, se muestra una matriz de `JobListEntry` objetos. En este ejemplo, se muestra una sola tarea.

```
aws snowball list-jobs
```

Salida:

```
{
  "JobListEntries": [
    {
```

```
    "CreationDate": 2016-09-27T14:50Z,  
    "Description": "Important Photos 2016-08-11",  
    "IsMaster": TRUE,  
    "JobId": "ABCd1e324fe-022f-488e-a98b-3b0566063db1",  
    "JobState": "Complete",  
    "JobType": "IMPORT",  
    "SnowballType": "EDGE"  
  }  
]  
}
```

Para obtener más información, consulte [Trabajos para dispositivos AWS Snowball Edge](#) en la Guía para desarrolladores de AWS Snowball.

- Para API obtener más información, consulte la Referencia [ListJobs](#) de AWS CLI comandos.

## SNSEjemplos de Amazon que utilizan AWS CLI

En los siguientes ejemplos de código, se muestra cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con AmazonSNS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Los escenarios son ejemplos de código que muestran cómo llevar a cabo una tarea específica a través de llamadas a varias funciones dentro del servicio o combinado con otros Servicios de AWS.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)
- [Escenarios](#)

## Acciones

### **add-permission**

En el siguiente ejemplo de código, se muestra cómo usar `add-permission`.

## AWS CLI

Para añadir un permiso a un tema

En el siguiente add-permission ejemplo, se agrega el permiso para 987654321098 que la AWS cuenta utilice la Publish acción con el tema especificado en la AWS cuenta123456789012.

```
aws sns add-permission \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --label Publish-Permission \  
  --aws-account-id 987654321098 \  
  --action-name Publish
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [AddPermission](#) la Referencia de AWS CLI comandos.

## check-if-phone-number-is-opted-out

En el siguiente ejemplo de código, se muestra cómo usar check-if-phone-number-is-opted-out.

## AWS CLI

Para comprobar la exclusión de un número de teléfono de los SMS mensajes

En el siguiente check-if-phone-number-is-opted-out ejemplo, se comprueba si el número de teléfono especificado está excluido de la recepción de SMS mensajes de la AWS cuenta corriente.

```
aws sns check-if-phone-number-is-opted-out \  
  --phone-number +1555550100
```

Salida:

```
{  
  "isOptedOut": false  
}
```

- Para API obtener más información, consulte [CheckIfPhoneNumberIsOptedOut](#) la Referencia de AWS CLI comandos.

## confirm-subscription

En el siguiente ejemplo de código, se muestra cómo usar `confirm-subscription`.

### AWS CLI

Para confirmar una suscripción

El siguiente `confirm-subscription` comando completa el proceso de confirmación iniciado al suscribirse a un SNS tema denominado `my-topic`. El parámetro `--token` proviene del mensaje de confirmación enviado al punto de conexión de notificación especificado en la llamada de suscripción.

```
aws sns confirm-subscription \
  --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic \
  --
token 2336412f37fb687f5d51e6e241d7700ae02f7124d8268910b858cb4db727ceeb2474bb937929d3bdd7ce5a
```

Salida:

```
{
  "SubscriptionArn": "arn:aws:sns:us-west-2:123456789012:my-
topic:8a21d249-4329-4871-acc6-7be709c6ea7f"
}
```

- Para API obtener más información, consulte [ConfirmSubscription](#) la Referencia de AWS CLI comandos.

## create-platform-application

En el siguiente ejemplo de código, se muestra cómo usar `create-platform-application`.

### AWS CLI

Para crear una aplicación de plataforma

En el siguiente `create-platform-application` ejemplo, se crea una aplicación de plataforma Google Firebase con la credencial de plataforma especificada.

```
aws sns create-platform-application \  
  --name MyApplication \  
  --platform GCM \  
  --attributes PlatformCredential=EXAMPLEabcd12345jklm67890stuv12345bcdef
```

Salida:

```
{  
  "PlatformApplicationArn": "arn:aws:sns:us-west-2:123456789012:app/GCM/  
MyApplication"  
}
```

- Para API obtener más información, consulta la Referencia [CreatePlatformApplication](#) de AWS CLI comandos.

## create-topic

En el siguiente ejemplo de código, se muestra cómo usar create-topic.

### AWS CLI

Para crear un tema de SNS

En el siguiente create-topic ejemplo se crea un SNS tema denominado my-topic.

```
aws sns create-topic \  
  --name my-topic
```

Salida:

```
{  
  "ResponseMetadata": {  
    "RequestId": "1469e8d7-1642-564e-b85d-a19b4b341f83"  
  },  
  "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic"  
}
```

Para obtener más información, consulte [Uso de la interfaz de línea de AWS comandos con Amazon SQS y Amazon SNS](#) en la Guía del usuario de la interfaz de línea de AWS comandos.



- Para API obtener más información, consulte [CreateTopic](#) la Referencia de AWS CLI comandos.

## delete-endpoint

En el siguiente ejemplo de código, se muestra cómo usar `delete-endpoint`.

### AWS CLI

Para eliminar el punto final de una aplicación de plataforma

En el siguiente `delete-endpoint` ejemplo, se elimina el punto final de la aplicación de plataforma especificado.

```
aws sns delete-endpoint \  
  --endpoint-arn arn:aws:sns:us-west-2:123456789012:endpoint/GCM/MyApplication/12345678-abcd-9012-efgh-345678901234
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteEndpoint](#) la Referencia de AWS CLI comandos.

## delete-platform-application

En el siguiente ejemplo de código, se muestra cómo usar `delete-platform-application`.

### AWS CLI

Para eliminar una aplicación de plataforma

En el siguiente `delete-platform-application` ejemplo, se elimina la aplicación de plataforma especificada.

```
aws sns delete-platform-application \  
  --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/ADM/MyApplication
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeletePlatformApplication](#) la Referencia de AWS CLI comandos.

## delete-topic

En el siguiente ejemplo de código, se muestra cómo usar `delete-topic`.

### AWS CLI

Para eliminar un SNS tema

En el siguiente `delete-topic` ejemplo, se elimina el SNS tema especificado.

```
aws sns delete-topic \  
  --topic-arn "arn:aws:sns:us-west-2:123456789012:my-topic"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteTopic](#) la Referencia de AWS CLI comandos.

## get-endpoint-attributes

En el siguiente ejemplo de código, se muestra cómo usar `get-endpoint-attributes`.

### AWS CLI

Para enumerar los atributos de punto final de la aplicación de plataforma

En el siguiente `get-endpoint-attributes` ejemplo, se enumeran los atributos del punto final de la aplicación de plataforma especificado.

```
aws sns get-endpoint-attributes \  
  --endpoint-arn arn:aws:sns:us-west-2:123456789012:endpoint/GCM/  
  MyApplication/12345678-abcd-9012-efgh-345678901234
```

Salida:

```
{  
  "Attributes": {  
    "Enabled": "true",  
    "Token": "EXAMPLE12345..."  
  }  
}
```

- Para API obtener más información, consulte [GetEndpointAttributes](#) la Referencia de AWS CLI comandos.

## get-platform-application-attributes

En el siguiente ejemplo de código, se muestra cómo usar `get-platform-application-attributes`.

### AWS CLI

Para enumerar los atributos de la aplicación de la plataforma

En el siguiente `get-platform-application-attributes` ejemplo, se enumeran los atributos de la aplicación de plataforma especificada.

```
aws sns get-platform-application-attributes \  
  --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/MPNS/  
  MyApplication
```

Salida:

```
{  
  "Attributes": {  
    "Enabled": "true",  
    "SuccessFeedbackSampleRate": "100"  
  }  
}
```

- Para API obtener más información, consulte [GetPlatformApplicationAttributes](#) la Referencia de AWS CLI comandos.

## get-sms-attributes

En el siguiente ejemplo de código, se muestra cómo usar `get-sms-attributes`.

### AWS CLI

Para ver una lista de los atributos de los SMS mensajes por defecto

En el siguiente `get-sms-attributes` ejemplo, se enumeran los atributos predeterminados para el envío de SMS mensajes.

```
aws sns get-sms-attributes
```

Salida:

```
{
  "attributes": {
    "DefaultSenderId": "MyName"
  }
}
```

- Para API obtener más información, consulte [GetSMSAttributes](#) en la Referencia de AWS CLI comandos.

## get-subscription-attributes

En el siguiente ejemplo de código, se muestra cómo usar `get-subscription-attributes`.

AWS CLI

Para recuperar los atributos de suscripción de un tema

A continuación, `get-subscription-attributes` se muestran los atributos de la suscripción especificada. Puede obtenerlos `subscription-arn` del resultado del `list-subscriptions` comando.

```
aws sns get-subscription-attributes \
  --subscription-arn "arn:aws:sns:us-west-2:123456789012:my-
  topic:8a21d249-4329-4871-acc6-7be709c6ea7f"
```

Salida:

```
{
  "Attributes": {
    "Endpoint": "my-email@example.com",
    "Protocol": "email",
    "RawMessageDelivery": "false",
    "ConfirmationWasAuthenticated": "false",
    "Owner": "123456789012",
    "SubscriptionArn": "arn:aws:sns:us-west-2:123456789012:my-
  topic:8a21d249-4329-4871-acc6-7be709c6ea7f",
    "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic"
  }
}
```

```
}

```

- Para API obtener más información, consulte [GetSubscriptionAttributes](#) la Referencia de AWS CLI comandos.

## get-topic-attributes

En el siguiente ejemplo de código, se muestra cómo usar `get-topic-attributes`.

### AWS CLI

Para recuperar los atributos de un tema

En el siguiente ejemplo de `get-topic-attributes`, se muestran los atributos del tema especificado.

```
aws sns get-topic-attributes \
  --topic-arn "arn:aws:sns:us-west-2:123456789012:my-topic"
```

Salida:

```
{
  "Attributes": {
    "SubscriptionsConfirmed": "1",
    "DisplayName": "my-topic",
    "SubscriptionsDeleted": "0",
    "EffectiveDeliveryPolicy": "{\"http\":{\"defaultHealthyRetryPolicy\":{\"minDelayTarget\":20,\"maxDelayTarget\":20,\"numRetries\":3,\"numMaxDelayRetries\":0,\"numNoDelayRetries\":0,\"numMinDelayRetries\":0,\"backoffFunction\":\"linear\"},\"disableSubscriptionOverrides\":false}}",
    "Owner": "123456789012",
    "Policy": "{\"Version\":\"2008-10-17\",\"Id\":\"__default_policy_ID\",\"Statement\":[{\"Sid\":\"__default_statement_ID\",\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"*\"},\"Action\":[\"SNS:Subscribe\",\"SNS:ListSubscriptionsByTopic\",\"SNS>DeleteTopic\",\"SNS:GetTopicAttributes\",\"SNS:Publish\",\"SNS:RemovePermission\",\"SNS:AddPermission\",\"SNS:SetTopicAttributes\"],\"Resource\":\"arn:aws:sns:us-west-2:123456789012:my-topic\",\"Condition\":{\"StringEquals\":{\"AWS:SourceOwner\":\"0123456789012\"}}}]}",
    "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic",
    "SubscriptionsPending": "0"
  }
}
```

- Para API obtener más información, consulte [GetTopicAttributes](#) la Referencia de AWS CLI comandos.

## list-endpoints-by-platform-application

En el siguiente ejemplo de código, se muestra cómo usar `list-endpoints-by-platform-application`.

AWS CLI

Para enumerar los puntos finales de una aplicación de plataforma

En el siguiente `list-endpoints-by-platform-application` ejemplo, se enumeran los puntos finales y los atributos de los puntos finales de la aplicación de plataforma especificada.

```
aws sns list-endpoints-by-platform-application \
  --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/GCM/MyApplication
```

Salida:

```
{
  "Endpoints": [
    {
      "Attributes": {
        "Token": "EXAMPLE12345...",
        "Enabled": "true"
      },
      "EndpointArn": "arn:aws:sns:us-west-2:123456789012:endpoint/GCM/MyApplication/12345678-abcd-9012-efgh-345678901234"
    }
  ]
}
```

- Para API obtener más información, consulte [ListEndpointsByPlatformApplication](#) la Referencia de AWS CLI comandos.

## list-phone-numbers-opted-out

En el siguiente ejemplo de código, se muestra cómo usar `list-phone-numbers-opted-out`.

## AWS CLI

Para ver una lista de las exclusiones de los SMS mensajes

En el siguiente `list-phone-numbers-opted-out` ejemplo, se enumeran los números de teléfono que se excluyeron de la recepción SMS de mensajes.

```
aws sns list-phone-numbers-opted-out
```

Salida:

```
{
  "phoneNumbers": [
    "+15555550100"
  ]
}
```

- Para API obtener más información, consulte [ListPhoneNumbersOptedOut](#) la Referencia de AWS CLI comandos.

## `list-platform-applications`

En el siguiente ejemplo de código, se muestra cómo usar `list-platform-applications`.

## AWS CLI

Para enumerar las aplicaciones de plataforma

En el siguiente `list-platform-applications` ejemplo se enumeran las aplicaciones de plataforma para ADM yMPNS.

```
aws sns list-platform-applications
```

Salida:

```
{
  "PlatformApplications": [
    {
      "PlatformApplicationArn": "arn:aws:sns:us-west-2:123456789012:app/ADM/MyApplication",
      "Attributes": {
        "SuccessFeedbackSampleRate": "100",
```

```

        "Enabled": "true"
      }
    },
    {
      "PlatformApplicationArn": "arn:aws:sns:us-west-2:123456789012:app/MPNS/MyOtherApplication",
      "Attributes": {
        "SuccessFeedbackSampleRate": "100",
        "Enabled": "true"
      }
    }
  ]
}

```

- Para API obtener más información, consulte [ListPlatformApplications](#) la Referencia de AWS CLI comandos.

## list-subscriptions-by-topic

En el siguiente ejemplo de código, se muestra cómo usar `list-subscriptions-by-topic`.

### AWS CLI

Para ver una lista de las suscripciones asociadas a un tema

A continuación, se `list-subscriptions-by-topic` recupera una lista de SNS las suscripciones asociadas al tema especificado.

```

aws sns list-subscriptions-by-topic \
  --topic-arn "arn:aws:sns:us-west-2:123456789012:my-topic"

```

Salida:

```

{
  "Subscriptions": [
    {
      "Owner": "123456789012",
      "Endpoint": "my-email@example.com",
      "Protocol": "email",
      "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic",
      "SubscriptionArn": "arn:aws:sns:us-west-2:123456789012:my-topic:8a21d249-4329-4871-acc6-7be709c6ea7f"
    }
  ]
}

```



```
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListSubscriptionsByTopic](#) la Referencia de AWS CLI comandos.

## list-subscriptions

En el siguiente ejemplo de código, se muestra cómo usar `list-subscriptions`.

### AWS CLI

Para enumerar sus SNS suscripciones

En el siguiente `list-subscriptions` ejemplo, se muestra una lista de las SNS suscripciones de su AWS cuenta.

```
aws sns list-subscriptions
```

Salida:

```
{  
  "Subscriptions": [  
    {  
      "Owner": "123456789012",  
      "Endpoint": "my-email@example.com",  
      "Protocol": "email",  
      "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic",  
      "SubscriptionArn": "arn:aws:sns:us-west-2:123456789012:my-  
topic:8a21d249-4329-4871-acc6-7be709c6ea7f"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListSubscriptions](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

## AWS CLI

Para enumerar las etiquetas de un tema

En el siguiente `list-tags-for-resource` ejemplo, se enumeran las etiquetas del SNS tema de Amazon especificado.

```
aws sns list-tags-for-resource \  
  --resource-arn arn:aws:sns:us-west-2:123456789012:MyTopic
```

Salida:

```
{  
  "Tags": [  
    {  
      "Key": "Team",  
      "Value": "Alpha"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## list-topics

En el siguiente ejemplo de código, se muestra cómo usar `list-topics`.

## AWS CLI

Para enumerar sus SNS temas

En el siguiente `list-topics` ejemplo, se enumeran todos SNS los temas de tu AWS cuenta.

```
aws sns list-topics
```

Salida:

```
{  
  "Topics": [  
    {  
      "TopicName": "MyTopic"  
    }  
  ]  
}
```

```
    {
      "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic"
    }
  ]
}
```

- Para API obtener más información, consulte [ListTopics](#) la Referencia de AWS CLI comandos.

## **opt-in-phone-number**

En el siguiente ejemplo de código, se muestra cómo usar `opt-in-phone-number`.

### AWS CLI

Para aceptar la recepción de mensajes SMS

En el siguiente `opt-in-phone-number` ejemplo, se opta por el número de teléfono especificado para recibir SMS mensajes.

```
aws sns opt-in-phone-number \
  --phone-number +15555550100
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [OptInPhoneNumber](#) la Referencia de AWS CLI comandos.

## **publish**

En el siguiente ejemplo de código, se muestra cómo usar `publish`.

### AWS CLI

Ejemplo 1: Para publicar un mensaje en un tema

El siguiente `publish` ejemplo publica el mensaje especificado en el SNS tema especificado. El mensaje proviene de un archivo de texto que le permite incluir saltos de línea.

```
aws sns publish \
  --topic-arn "arn:aws:sns:us-west-2:123456789012:my-topic" \
  --message file://message.txt
```

Contenidos de `message.txt`:

```
Hello World
Second Line
```

Salida:

```
{
  "MessageId": "123a45b6-7890-12c3-45d6-111122223333"
}
```

Ejemplo 2: publicar un SMS mensaje en un número de teléfono

En el siguiente ejemplo de `publish`, se publica el mensaje `Hello world!` en el número de teléfono `+1-555-555-0100`.

```
aws sns publish \
  --message "Hello world!" \
  --phone-number +1-555-555-0100
```

Salida:

```
{
  "MessageId": "123a45b6-7890-12c3-45d6-333322221111"
}
```

- Para API obtener más información, consulte [Publish](#) in AWS CLI Command Reference.

## **put-data-protection-policy**

En el siguiente ejemplo de código, se muestra cómo usar `put-data-protection-policy`.

### AWS CLI

Para establecer una política de protección de datos

Ejemplo 1: impedir que los editores publiquen mensajes con `CreditCardNumber`

El siguiente `put-data-protection-policy` ejemplo impide a los editores publicar mensajes con `CreditCardNumber`.

```
aws sns put-data-protection-policy \  
  --resource-arn arn:aws:sns:us-east-1:123456789012:mytopic \  
  --data-protection-policy '{"Name\":\"data_protection_policy\",\"Description\  
\": \"Example data protection policy\",\"Version\":\"2021-06-01\",\"Statement\  
\": [{\"DataDirection\":\"Inbound\",\"Principal\":[\"*\"],\"DataIdentifier\  
\": [\"arn:aws:dataprotection::aws:data-identifier/CreditCardNumber\"],\"Operation\  
\": {\"Deny\":{}}}]}'
```

Este comando no genera ninguna salida.

Ejemplo 2: Para cargar parámetros de un archivo

Lo siguiente `put-data-protection-policy` carga los parámetros de un archivo.

```
aws sns put-data-protection-policy \  
  --resource-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --data-protection-policy file://policy.json
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutDataProtectionPolicy](#) la Referencia de AWS CLI comandos.

## remove-permission

En el siguiente ejemplo de código, se muestra cómo usar `remove-permission`.

AWS CLI

Para eliminar un permiso de un tema

En el siguiente `remove-permission` ejemplo, se quita el permiso `Publish-Permission` del tema especificado.

```
aws sns remove-permission \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --label Publish-Permission
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [RemovePermission](#) la Referencia de AWS CLI comandos.

## set-endpoint-attributes

En el siguiente ejemplo de código, se muestra cómo usar `set-endpoint-attributes`.

### AWS CLI

Para establecer los atributos del punto final

En el siguiente `set-endpoint-attributes` ejemplo, se deshabilita el punto final de la aplicación de plataforma especificado.

```
aws sns set-endpoint-attributes \  
  --endpoint-arn arn:aws:sns:us-west-2:123456789012:endpoint/GCM/  
MyApplication/12345678-abcd-9012-efgh-345678901234 \  
  --attributes Enabled=false
```

Salida:

```
{  
  "Attributes": {  
    "Enabled": "false",  
    "Token": "EXAMPLE12345..."  
  }  
}
```

- Para API obtener más información, consulte [SetEndpointAttributes](#) la Referencia de AWS CLI comandos.

## set-platform-application-attributes

En el siguiente ejemplo de código, se muestra cómo usar `set-platform-application-attributes`.

### AWS CLI

Para configurar los atributos de la aplicación de plataforma

El siguiente `set-platform-application-attributes` ejemplo establece el `EventDeliveryFailure` atributo de la aplicación de plataforma especificada en el SNS tema ARN de Amazon especificado.

```
aws sns set-platform-application-attributes \  
  --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/GCM/MyApplication \  
  --attributes EventDeliveryFailure=arn:aws:sns:us-west-2:123456789012:AnotherTopic
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SetPlatformApplicationAttributes](#) la Referencia de AWS CLI comandos.

## set-sms-attributes

En el siguiente ejemplo de código, se muestra cómo usar `set-sms-attributes`.

### AWS CLI

Para configurar los atributos de los SMS mensajes

En el siguiente `set-sms-attributes` ejemplo, se establece el identificador de remitente predeterminado para SMS los mensajes en `MyName`.

```
aws sns set-sms-attributes \  
  --attributes DefaultSenderId=MyName
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SetSMSAttributes](#) en la referencia de AWS CLI comandos.

## set-subscription-attributes

En el siguiente ejemplo de código, se muestra cómo usar `set-subscription-attributes`.

### AWS CLI

Para establecer los atributos de suscripción

En el siguiente `set-subscription-attributes` ejemplo, se establece el `RawMessageDelivery` atributo en una SQS suscripción.

```
aws sns set-subscription-attributes \  
  --attributes RawMessageDelivery
```

```
--subscription-arn arn:aws:sns:us-east-1:123456789012:mytopic:f248de18-2cf6-578c-8592-b6f1eaa877dc \  
--attribute-name RawMessageDelivery \  
--attribute-value true
```

Este comando no genera ninguna salida.

En el siguiente `set-subscription-attributes` ejemplo, se establece un `FilterPolicy` atributo en una SQS suscripción.

```
aws sns set-subscription-attributes \  
--subscription-arn arn:aws:sns:us-east-1:123456789012:mytopic:f248de18-2cf6-578c-8592-b6f1eaa877dc \  
--attribute-name FilterPolicy \  
--attribute-value "{ \"anyMandatoryKey\": [\"any\", \"of\", \"these\"] }"
```

Este comando no genera ninguna salida.

En el siguiente `set-subscription-attributes` ejemplo, se quita el `FilterPolicy` atributo de una SQS suscripción.

```
aws sns set-subscription-attributes \  
--subscription-arn arn:aws:sns:us-east-1:123456789012:mytopic:f248de18-2cf6-578c-8592-b6f1eaa877dc \  
--attribute-name FilterPolicy \  
--attribute-value "{}"
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SetSubscriptionAttributes](#) la Referencia de AWS CLI comandos.

## set-topic-attributes

En el siguiente ejemplo de código, se muestra cómo usar `set-topic-attributes`.

### AWS CLI

Para establecer un atributo para un tema

En el ejemplo de `set-topic-attributes` siguiente, se establece el atributo `DisplayName` del tema especificado.



```
aws sns set-topic-attributes \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --attribute-name DisplayName \  
  --attribute-value MyTopicDisplayName
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [SetTopicAttributes](#) la Referencia de AWS CLI comandos.

## subscribe

En el siguiente ejemplo de código, se muestra cómo usar `subscribe`.

### AWS CLI

Para suscribirse a un tema

El siguiente comando `subscribe` suscribe una dirección de correo electrónico al tema especificado.

```
aws sns subscribe \  
  --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic \  
  --protocol email \  
  --notification-endpoint my-email@example.com
```

Salida:

```
{  
  "SubscriptionArn": "pending confirmation"  
}
```

- Para API obtener más información, consulte [Suscribirse](#) en AWS CLI Command Reference.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para agregar una etiqueta a un tema

En el siguiente `tag-resource` ejemplo, se añade una etiqueta de metadatos al SNS tema de Amazon especificado.

```
aws sns tag-resource \  
  --resource-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --tags Key=Team,Value=Alpha
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## unsubscribe

En el siguiente ejemplo de código, se muestra cómo usar `unsubscribe`.

### AWS CLI

Para cancelar la suscripción a un tema

En el siguiente ejemplo de `unsubscribe`, se elimina la suscripción especificada de un tema.

```
aws sns unsubscribe \  
  --subscription-arn arn:aws:sns:us-west-2:0123456789012:my-  
topic:8a21d249-4329-4871-acc6-7be709c6ea7f
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [Cancelar la suscripción](#) en la referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para eliminar una etiqueta de un tema

En el siguiente `untag-resource` ejemplo, se eliminan todas las etiquetas con las claves especificadas del SNS tema de Amazon especificado.

```
aws sns untag-resource \  
  --resource-arn arn:aws:sns:us-west-2:123456789012:MyTopic \  
  --tag-keys Team
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## Escenarios

Creación de un punto de enlace de la plataforma para notificaciones push

El siguiente ejemplo de código muestra cómo crear un punto final de plataforma para las notificaciones SNS push de Amazon.

### AWS CLI

Para crear un punto de conexión de aplicación de plataforma

En el siguiente ejemplo de `create-platform-endpoint`, se crea un punto de conexión para la aplicación de plataforma especificada mediante el token especificado.

```
aws sns create-platform-endpoint \  
  --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/GCM/MyApplication \  
  --token EXAMPLE12345...
```

Salida:

```
{  
  "EndpointArn": "arn:aws:sns:us-west-2:1234567890:endpoint/GCM/  
MyApplication/12345678-abcd-9012-efgh-345678901234"  
}
```

## SQSEjemplos de Amazon que utilizan AWS CLI

En los siguientes ejemplos de código, se muestra cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con AmazonSQS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **add-permission**

En el siguiente ejemplo de código, se muestra cómo usar `add-permission`.

AWS CLI

Para añadir un permiso a una cola

En este ejemplo, se permite a la AWS cuenta especificada enviar mensajes a la cola especificada.

Comando:

```
aws sqs add-permission --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --label SendMessageFromMyQueue --aws-account-ids 12345EXAMPLE --actions SendMessage
```

Salida:

```
None.
```

- Para API obtener más información, consulte [AddPermission](#) la Referencia de AWS CLI comandos.

### **cancel-message-move-task**

En el siguiente ejemplo de código, se muestra cómo usar `cancel-message-move-task`.

## AWS CLI

Para cancelar un mensaje, mueva la tarea

El siguiente `cancel-message-move-task` ejemplo cancela la tarea de traslado de mensajes especificada.

```
aws sqs cancel-message-move-task \  
  --task-handle AQEB6nR4...HzLvZQ==
```

Salida:

```
{  
  "ApproximateNumberOfMessagesMoved": 102  
}
```

Para obtener más información, consulta la [sección SQS API Permisos de Amazon: acciones y referencia de recursos](#) en la Guía para desarrolladores.

- Para API obtener más información, consulte [CancelMessageMoveTask](#) la Referencia de AWS CLI comandos.

## change-message-visibility-batch

En el siguiente ejemplo de código, se muestra cómo usar `change-message-visibility-batch`.

### AWS CLI

Para cambiar la visibilidad de los tiempos de espera de varios mensajes como un lote

En este ejemplo, se cambia la visibilidad del tiempo de espera de los dos mensajes especificados a 10 horas (10 horas \* 60 minutos \* 60 segundos).

Comando:

```
aws sqs change-message-visibility-batch --queue-url https://sqs.us-  
east-1.amazonaws.com/80398EXAMPLE/MyQueue --entries file://change-message-  
visibility-batch.json
```

Archivo de entrada (.json): `change-message-visibility-batch`

```
[
  {
    "Id": "FirstMessage",
    "ReceiptHandle": "AQEBhz2q...Jf3kaw==",
    "VisibilityTimeout": 36000
  },
  {
    "Id": "SecondMessage",
    "ReceiptHandle": "AQEBkTUH...HifSnw==",
    "VisibilityTimeout": 36000
  }
]
```

Salida:

```
{
  "Successful": [
    {
      "Id": "SecondMessage"
    },
    {
      "Id": "FirstMessage"
    }
  ]
}
```

- Para API obtener más información, consulte [ChangeMessageVisibilityBatch](#) la Referencia de AWS CLI comandos.

## change-message-visibility

En el siguiente ejemplo de código, se muestra cómo usar `change-message-visibility`.

### AWS CLI

Cambio de la visibilidad del tiempo de espera de un mensaje

En este ejemplo, se cambia la visibilidad del tiempo de espera del mensaje especificado a 10 horas (10 horas \* 60 minutos \* 60 segundos).

Comando:

```
aws sqs change-message-visibility --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --receipt-handle AQEBTpyI...t6HyQg== --visibility-timeout 36000
```

Salida:

```
None.
```

- Para API obtener más información, consulte [ChangeMessageVisibility](#) la Referencia de AWS CLI comandos.

## create-queue

En el siguiente ejemplo de código, se muestra cómo usar create-queue.

### AWS CLI

Para crear una cola

En este ejemplo, se crea una cola con el nombre especificado, se establece el período de retención de mensajes en 3 días (3 días \* 24 horas \* 60 minutos \* 60 segundos) y se establece la cola de mensajes fallidos en la cola especificada con un recuento máximo de 1000 mensajes recibidos.

Comando:

```
aws sqs create-queue --queue-name MyQueue --attributes file://create-queue.json
```

Archivo de entrada (create-queue.json):

```
{
  "RedrivePolicy": "{\"deadLetterTargetArn\":\"arn:aws:sqs:us-east-1:80398EXAMPLE:MyDeadLetterQueue\", \"maxReceiveCount\":\"1000\"}\",
  \"MessageRetentionPeriod\": \"259200\"
}
```

Salida:

```
{
```

```
"QueueUrl": "https://queue.amazonaws.com/80398EXAMPLE/MyQueue"
}
```

- Para API obtener más información, consulte [CreateQueue](#) la Referencia de AWS CLI comandos.

## delete-message-batch

En el siguiente ejemplo de código, se muestra cómo usar `delete-message-batch`.

### AWS CLI

Eliminación de varios mensajes en un lote

En este ejemplo, se eliminan los mensajes especificados.

Comando:

```
aws sqs delete-message-batch --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --entries file://delete-message-batch.json
```

Archivo de entrada (delete-message-batch.json):

```
[
  {
    "Id": "FirstMessage",
    "ReceiptHandle": "AQEB1mg1...Z4GuLw=="
  },
  {
    "Id": "SecondMessage",
    "ReceiptHandle": "AQEBLsYM...VQubAA=="
  }
]
```

Salida:

```
{
  "Successful": [
    {
      "Id": "FirstMessage"
    }
  ]
}
```



```
    },  
    {  
      "Id": "SecondMessage"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [DeleteMessageBatch](#) la Referencia de AWS CLI comandos.

## delete-message

En el siguiente ejemplo de código, se muestra cómo usar `delete-message`.

### AWS CLI

Eliminación de un mensaje

En este ejemplo, se elimina el mensaje especificado.

Comando:

```
aws sqs delete-message --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --receipt-handle AQEBRXTo...q2doVA==
```

Salida:

```
None.
```

- Para API obtener más información, consulte [DeleteMessage](#) la Referencia de AWS CLI comandos.

## delete-queue

En el siguiente ejemplo de código, se muestra cómo usar `delete-queue`.

### AWS CLI

Para eliminar una cola

En este ejemplo, se elimina la cola especificada.

Comando:

```
aws sqs delete-queue --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyNewerQueue
```

Salida:

```
None.
```

- Para API obtener más información, consulte [DeleteQueue](#) la Referencia de AWS CLI comandos.

## get-queue-attributes

En el siguiente ejemplo de código, se muestra cómo usar `get-queue-attributes`.

AWS CLI

Obtención de los atributos de una cola

En este ejemplo, se obtienen todos los atributos de la cola especificada.

Comando:

```
aws sqs get-queue-attributes --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --attribute-names All
```

Salida:

```
{
  "Attributes": {
    "ApproximateNumberOfMessagesNotVisible": "0",
    "RedrivePolicy": "{\"deadLetterTargetArn\":\"arn:aws:sqs:us-east-1:80398EXAMPLE:MyDeadLetterQueue\", \"maxReceiveCount\":1000}",
    "MessageRetentionPeriod": "345600",
    "ApproximateNumberOfMessagesDelayed": "0",
    "MaximumMessageSize": "262144",
```

```
"CreatedTimestamp": "1442426968",
"ApproximateNumberOfMessages": "0",
"ReceiveMessageWaitTimeSeconds": "0",
"DelaySeconds": "0",
"VisibilityTimeout": "30",
"LastModifiedTimestamp": "1442426968",
"QueueArn": "arn:aws:sqs:us-east-1:80398EXAMPLE:MyNewQueue"
}
}
```

En este ejemplo, se obtienen únicamente los atributos de tamaño máximo de mensaje y tiempo de espera de visibilidad de la cola especificada.

Comando:

```
aws sqs get-queue-attributes --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyNewQueue --attribute-names MaximumMessageSize VisibilityTimeout
```

Salida:

```
{
  "Attributes": {
    "VisibilityTimeout": "30",
    "MaximumMessageSize": "262144"
  }
}
```

- Para API obtener más información, consulte [GetQueueAttributes](#) la Referencia de AWS CLI comandos.

## get-queue-url

En el siguiente ejemplo de código, se muestra cómo usar `get-queue-url`.

AWS CLI

Para obtener una cola URL

En este ejemplo se obtienen las colas especificadas. URL

Comando:

```
aws sqs get-queue-url --queue-name MyQueue
```

Salida:

```
{  
  "QueueUrl": "https://queue.amazonaws.com/80398EXAMPLE/MyQueue"  
}
```

- Para API obtener más información, consulte [GetQueueUrl](#) la Referencia de AWS CLI comandos.

## list-dead-letter-source-queues

En el siguiente ejemplo de código, se muestra cómo usar `list-dead-letter-source-queues`.

AWS CLI

Para enumerar las colas de fuentes de letra muerta

En este ejemplo, se muestran las colas asociadas a la cola de origen de cartas no utilizadas especificada.

Comando:

```
aws sqs list-dead-letter-source-queues --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyDeadLetterQueue
```

Salida:

```
{  
  "queueUrls": [  
    "https://queue.amazonaws.com/80398EXAMPLE/MyQueue",  
    "https://queue.amazonaws.com/80398EXAMPLE/MyOtherQueue"  
  ]  
}
```

- Para API obtener más información, consulte la Referencia [ListDeadLetterSourceQueues](#) de AWS CLI comandos.

## list-message-move-tasks

En el siguiente ejemplo de código, se muestra cómo usar `list-message-move-tasks`.

### AWS CLI

Para ver el mensaje, mueva las tareas

En el siguiente `list-message-move-tasks` ejemplo, se enumeran las dos tareas de movimiento de mensajes más recientes de la cola especificada.

```
aws sqs list-message-move-tasks \  
  --source-arn arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue \  
  --max-results 2
```

Salida:

```
{  
  "Results": [  
    {  
      "TaskHandle": "AQEB6nR4...HzlvZQ==",  
      "Status": "RUNNING",  
      "SourceArn": "arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue1",  
      "DestinationArn": "arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue2",  
      "MaxNumberOfMessagesPerSecond": 50,  
      "ApproximateNumberOfMessagesMoved": 203,  
      "ApproximateNumberOfMessagesToMove": 30,  
      "StartedTimestamp": 1442428276921  
    },  
    {  
      "Status": "COMPLETED",  
      "SourceArn": "arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue1",  
      "DestinationArn": "arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue2",  
      "ApproximateNumberOfMessagesMoved": 29,  
      "ApproximateNumberOfMessagesToMove": 0,  
      "StartedTimestamp": 1342428272093  
    }  
  ]  
}
```

Para obtener más información, consulta la [sección SQS API Permisos de Amazon: acciones y referencia de recursos](#) en la Guía para desarrolladores.

- Para API obtener más información, consulte [ListMessageMoveTasks](#) la Referencia de AWS CLI comandos.

## list-queue-tags

En el siguiente ejemplo de código, se muestra cómo usar `list-queue-tags`.

AWS CLI

Para enumerar todas las etiquetas de asignación de costes de una cola

En el siguiente `list-queue-tags` ejemplo, se muestran todas las etiquetas de asignación de costes asociadas a la cola especificada.

```
aws sqs list-queue-tags \  
  --queue-url https://sqs.us-west-2.amazonaws.com/123456789012/MyQueue
```

Salida:

```
{  
  "Tags": {  
    "Team": "Alpha"  
  }  
}
```

Para obtener más información, consulte [Publicar etiquetas de asignación de costes](#) en la Guía para desarrolladores de Amazon Simple Queue Service.

- Para API obtener más información, consulte [ListQueueTags](#) la Referencia de AWS CLI comandos.

## list-queues

En el siguiente ejemplo de código, se muestra cómo usar `list-queues`.

AWS CLI

Creación de una lista de las colas

En este ejemplo, se enumeran todas las colas.

Comando:

```
aws sqs list-queues
```

Salida:

```
{
  "QueueUrls": [
    "https://queue.amazonaws.com/80398EXAMPLE/MyDeadLetterQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyOtherQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/TestQueue1",
    "https://queue.amazonaws.com/80398EXAMPLE/TestQueue2"
  ]
}
```

En este ejemplo, solo se muestran las colas que comienzan por "My".

Comando:

```
aws sqs list-queues --queue-name-prefix My
```

Salida:

```
{
  "QueueUrls": [
    "https://queue.amazonaws.com/80398EXAMPLE/MyDeadLetterQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyOtherQueue"
  ]
}
```

- Para API obtener más información, consulte [ListQueues](#) la Referencia de AWS CLI comandos.

## purge-queue

En el siguiente ejemplo de código, se muestra cómo usar purge-queue.

### AWS CLI

Para purgar una cola

En este ejemplo, se eliminan todos los mensajes de la cola especificada.

Comando:

```
aws sqs purge-queue --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyNewQueue
```

Salida:

```
None.
```

- Para API obtener más información, consulte la Referencia [PurgeQueue](#) de AWS CLI comandos.

## receive-message

En el siguiente ejemplo de código, se muestra cómo usar `receive-message`.

AWS CLI

Recepción de un mensaje

En este ejemplo, se reciben hasta 10 mensajes disponibles y se devuelven todos los atributos disponibles.

Comando:

```
aws sqs receive-message --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --attribute-names All --message-attribute-names All --max-number-of-messages 10
```

Salida:

```
{
  "Messages": [
    {
      "Body": "My first message.",
      "ReceiptHandle": "AQEBzbVv...fqNzFw==",
      "MD5ofBody": "1000f835...a35411fa",
      "MD5ofMessageAttributes": "9424c491...26bc3ae7",
```



```

    "MessageId": "d6790f8d-d575-4f01-bc51-40122EXAMPLE",
    "Attributes": {
      "ApproximateFirstReceiveTimestamp": "1442428276921",
      "SenderId": "AIDAIKMSNQ7TEXAMPLE",
      "ApproximateReceiveCount": "5",
      "SentTimestamp": "1442428276921"
    },
    "MessageAttributes": {
      "PostalCode": {
        "DataType": "String",
        "StringValue": "ABC123"
      },
      "City": {
        "DataType": "String",
        "StringValue": "Any City"
      }
    }
  }
]
}

```

En este ejemplo se recibe el siguiente mensaje disponible y se devuelven solo los `SentTimestamp` atributos `SenderId` y, además del atributo de `PostalCode` mensaje.

Comando:

```

aws sqs receive-message --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --attribute-names SenderId SentTimestamp
--message-attribute-names PostalCode

```

Salida:

```

{
  "Messages": [
    {
      "Body": "My first message.",
      "ReceiptHandle": "AQEB6nR4...HzlvZQ==",
      "MD5ofBody": "1000f835...a35411fa",
      "MD5ofMessageAttributes": "b8e89563...e088e74f",
      "MessageId": "d6790f8d-d575-4f01-bc51-40122EXAMPLE",
      "Attributes": {
        "SenderId": "AIDAIKMSNQ7TEXAMPLE",

```

```
    "SentTimestamp": "1442428276921"
  },
  "MessageAttributes": {
    "PostalCode": {
      "DataType": "String",
      "StringValue": "ABC123"
    }
  }
}
]
```

- Para API obtener más información, consulte [ReceiveMessage](#) la Referencia de AWS CLI comandos.

## remove-permission

En el siguiente ejemplo de código, se muestra cómo usar `remove-permission`.

### AWS CLI

Para eliminar un permiso

En este ejemplo, se quita el permiso con la etiqueta especificada de la cola especificada.

Comando:

```
aws sqs remove-permission --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --label SendMessageFromMyQueue
```

Salida:

```
None.
```

- Para API obtener más información, consulte [RemovePermission](#) la Referencia de AWS CLI comandos.

## send-message-batch

En el siguiente ejemplo de código, se muestra cómo usar `send-message-batch`.

## AWS CLI

### Envío de varios mensajes por lotes

En este ejemplo, se envían 2 mensajes con los cuerpos de los mensajes, los períodos de retraso y los atributos de los mensajes especificados a la cola especificada.

Comando:

```
aws sqs send-message-batch --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --entries file://send-message-batch.json
```

Archivo de entrada (send-message-batch.json):

```
[
  {
    "Id": "FuelReport-0001-2015-09-16T140731Z",
    "MessageBody": "Fuel report for account 0001 on 2015-09-16 at 02:07:31 PM.",
    "DelaySeconds": 10,
    "MessageAttributes": {
      "SellerName": {
        "DataType": "String",
        "StringValue": "Example Store"
      },
      "City": {
        "DataType": "String",
        "StringValue": "Any City"
      },
      "Region": {
        "DataType": "String",
        "StringValue": "WA"
      },
      "PostalCode": {
        "DataType": "String",
        "StringValue": "99065"
      },
      "PricePerGallon": {
        "DataType": "Number",
        "StringValue": "1.99"
      }
    }
  },
  {
```

```

    "Id": "FuelReport-0002-2015-09-16T140930Z",
    "MessageBody": "Fuel report for account 0002 on 2015-09-16 at 02:09:30 PM.",
    "DelaySeconds": 10,
    "MessageAttributes": {
      "SellerName": {
        "DataType": "String",
        "StringValue": "Example Fuels"
      },
      "City": {
        "DataType": "String",
        "StringValue": "North Town"
      },
      "Region": {
        "DataType": "String",
        "StringValue": "WA"
      },
      "PostalCode": {
        "DataType": "String",
        "StringValue": "99123"
      },
      "PricePerGallon": {
        "DataType": "Number",
        "StringValue": "1.87"
      }
    }
  }
]

```

**Salida:**

```

{
  "Successful": [
    {
      "MD5ofMessageBody": "203c4a38...7943237e",
      "MD5ofMessageAttributes": "10809b55...baf283ef",
      "Id": "FuelReport-0001-2015-09-16T140731Z",
      "MessageId": "d175070c-d6b8-4101-861d-adeb3EXAMPLE"
    },
    {
      "MD5ofMessageBody": "2cf0159a...c1980595",
      "MD5ofMessageAttributes": "55623928...ae354a25",
      "Id": "FuelReport-0002-2015-09-16T140930Z",
      "MessageId": "f9b7d55d-0570-413e-b9c5-a9264EXAMPLE"
    }
  ]
}

```

```
    }  
  ]  
}
```

- Para API obtener más información, consulte [SendMessageBatch](#) la Referencia de AWS CLI comandos.

## send-message

En el siguiente ejemplo de código, se muestra cómo usar send-message.

### AWS CLI

Para enviar un mensaje

En este ejemplo, se envía un mensaje con el cuerpo del mensaje, el período de retraso y los atributos del mensaje especificados a la cola especificada.

Comando:

```
aws sqs send-message --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --message-body "Information about the largest city in Any Region." --delay-seconds 10 --message-attributes file://send-message.json
```

Archivo de entrada (send-message.json):

```
{  
  "City": {  
    "DataType": "String",  
    "StringValue": "Any City"  
  },  
  "Greeting": {  
    "DataType": "Binary",  
    "BinaryValue": "Hello, World!"  
  },  
  "Population": {  
    "DataType": "Number",  
    "StringValue": "1250800"  
  }  
}
```

Salida:

```
{
  "MD5ofMessageBody": "51b0a325...39163aa0",
  "MD5ofMessageAttributes": "00484c68...59e48f06",
  "MessageId": "da68f62c-0c07-4bee-bf5f-7e856EXAMPLE"
}
```

- Para API obtener más información, consulte [SendMessage](#) la Referencia de AWS CLI comandos.

## set-queue-attributes

En el siguiente ejemplo de código, se muestra cómo usar `set-queue-attributes`.

### AWS CLI

Fijación de los atributos de una cola

Este ejemplo establece la cola especificada en un retraso de entrega de 10 segundos, un tamaño máximo de mensaje de 128 KB (128 KB \* 1024 bytes), un período de retención de mensajes de 3 días (3 días \* 24 horas \* 60 minutos \* 60 segundos), un tiempo de espera de recepción de mensajes de 20 segundos y un tiempo de espera de visibilidad predeterminado de 60 segundos. En este ejemplo, también se asocia la cola de mensajes fallidos especificada a un recuento máximo de 1000 mensajes recibidos.

Comando:

```
aws sqs set-queue-attributes --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyNewQueue --attributes file://set-queue-attributes.json
```

Archivo de entrada (set-queue-attributes.json):

```
{
  "DelaySeconds": "10",
  "MaximumMessageSize": "131072",
  "MessageRetentionPeriod": "259200",
  "ReceiveMessageWaitTimeSeconds": "20",
  "RedrivePolicy": "{\"deadLetterTargetArn\":\"arn:aws:sqs:us-east-1:80398EXAMPLE:MyDeadLetterQueue\", \"maxReceiveCount\": \"1000\"}",
  "VisibilityTimeout": "60"
}
```

Salida:

```
None.
```

- Para API obtener más información, consulte [SetQueueAttributes](#) la Referencia de AWS CLI comandos.

## start-message-move-task

En el siguiente ejemplo de código, se muestra cómo usar `start-message-move-task`.

AWS CLI

Ejemplo 1: \*Para iniciar una tarea de traslado de mensajes\*

En el siguiente `start-message-move-task` ejemplo, se inicia una tarea de traslado de mensajes para redirigir los mensajes de la cola de letras muertas especificada a la cola de origen.

```
aws sqs start-message-move-task \  
  --source-arn arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue
```

Salida:

```
{  
  "TaskHandle": "AQEB6nR4...Hz1vZQ=="  
}
```

Para obtener más información, consulte [Este es el título del tema en el](#) nombre de su guía.

Ejemplo 2: \*Para iniciar una tarea de traslado de mensajes con una velocidad máxima\*

En el siguiente `start-message-move-task` ejemplo, se inicia una tarea de traslado de mensajes para redirigir los mensajes de la cola de letras muertas especificada a la cola de destino especificada a una velocidad máxima de 50 mensajes por segundo.

```
aws sqs start-message-move-task \  
  --source-arn arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue1 \  
  --destination-arn arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue2 \  
  --max-number-of-messages-per-second 50
```

Salida:

```
{
  "TaskHandle": "AQEB6nR4...Hz1vZQ=="
}
```

Para obtener más información, consulta la [sección SQS API Permisos de Amazon: acciones y referencia de recursos](#) en la Guía para desarrolladores.

- Para API obtener más información, consulte [StartMessageMoveTask](#) la Referencia de AWS CLI comandos.

## tag-queue

En el siguiente ejemplo de código, se muestra cómo usar tag-queue.

### AWS CLI

Para añadir etiquetas de asignación de costes a una cola

En el siguiente tag-queue ejemplo, se añade una etiqueta de asignación de costes a la SQS cola de Amazon especificada.

```
aws sqs tag-queue \  
  --queue-url https://sqs.us-west-2.amazonaws.com/123456789012/MyQueue \  
  --tags Priority=Highest
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir etiquetas de asignación de costes](#) en la Guía para desarrolladores de Amazon Simple Queue Service.

- Para API obtener más información, consulte [TagQueue](#) la Referencia de AWS CLI comandos.

## untag-queue

En el siguiente ejemplo de código, se muestra cómo usar untag-queue.

### AWS CLI

Para eliminar las etiquetas de asignación de costes de una cola

En el siguiente untag-queue ejemplo, se elimina una etiqueta de asignación de costes de la SQS cola de Amazon especificada.



```
aws sqs untag-queue \  
  --queue-url https://sqs.us-west-2.amazonaws.com/123456789012/MyQueue \  
  --tag-keys "Priority"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Añadir etiquetas de asignación de costes](#) en la Guía para desarrolladores de Amazon Simple Queue Service.

- Para API obtener más información, consulte [UntagQueue](#) la Referencia de AWS CLI comandos.

## Ejemplos de Storage Gateway que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante Storage Gateway. AWS Command Line Interface

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

Acciones

### **describe-gateway-information**

En el siguiente ejemplo de código, se muestra cómo usar `describe-gateway-information`.

AWS CLI

Para describir una puerta de enlace

El siguiente `describe-gateway-information` comando devuelve metadatos sobre la puerta de enlace especificada. Para especificar qué puerta de enlace desea describir, utilice el nombre de recurso de Amazon (ARN) de la puerta de enlace en el comando.

En este ejemplo se especifica una puerta de enlace con el identificador `sgw-12A3456B` de la cuenta `123456789012`:

```
aws storagegateway describe-gateway-information --gateway-arn "arn:aws:storagegateway:us-west-2:123456789012:gateway/sgw-12A3456B"
```

Este comando genera un JSON bloque que contiene metadatos sobre la puerta de enlace, como su nombre, las interfaces de red, la zona horaria configurada y el estado (independientemente de que la puerta de enlace esté en ejecución o no).

- Para API obtener más información, consulte [DescribeGatewayInformation](#) la Referencia de AWS CLI comandos.

## list-file-shares

En el siguiente ejemplo de código, se muestra cómo usar `list-file-shares`.

### AWS CLI

Para enumerar los archivos compartidos

En el siguiente `command-name` ejemplo, se muestran los widgets disponibles en su AWS cuenta.

```
aws storagegateway list-file-shares \
  --gateway-arn arn:aws:storagegateway:us-east-1:209870788375:gateway/sgw-FB02E292
```

Salida:

```
{
  "FileShareInfoList": [
    {
      "FileShareType": "NFS",
      "FileShareARN": "arn:aws:storagegateway:us-east-1:111122223333:share/share-2FA12345",
      "FileShareId": "share-2FA12345",
      "FileShareStatus": "AVAILABLE",
      "GatewayARN": "arn:aws:storagegateway:us-east-1:111122223333:gateway/sgw-FB0AAAAA"
    }
  ],
  "Marker": null
}
```

```
}
```

Para obtener más información, consulte [ListFileShares](#) la API Referencia del servicio AWS Storage Gateway.

- Para API obtener más información, consulte [ListFileShares](#) la Referencia de AWS CLI comandos.

## **list-gateways**

En el siguiente ejemplo de código, se muestra cómo usar `list-gateways`.

### AWS CLI

Para enumerar las puertas de enlace de una cuenta

El siguiente `list-gateways` comando muestra todas las puertas de enlace definidas para una cuenta:

```
aws storagegateway list-gateways
```

Este comando genera un JSON bloque que contiene una lista de nombres de recursos de Amazon Resource Names (ARNs) de la puerta de enlace.

- Para API obtener más información, consulte [ListGateways](#) la Referencia de AWS CLI comandos.

## **list-volumes**

En el siguiente ejemplo de código, se muestra cómo usar `list-volumes`.

### AWS CLI

Para ver una lista de los volúmenes configurados para una puerta de enlace

El siguiente `list-volumes` comando devuelve una lista de los volúmenes configurados para la puerta de enlace especificada. Para especificar qué puerta de enlace desea describir, utilice el nombre de recurso de Amazon (ARN) de la puerta de enlace en el comando.

En este ejemplo se especifica una puerta de enlace con el identificador `sgw-12A3456B` de la cuenta `123456789012`:

```
aws storagegateway list-volumes --gateway-arn "arn:aws:storagegateway:us-west-2:123456789012:gateway/sgw-12A3456B"
```

Este comando genera un JSON bloque con una lista de volúmenes que incluye el tipo y ARN para cada volumen.

- Para API obtener más información, consulte [ListVolumes](#) la Referencia de AWS CLI comandos.

## refresh-cache

En el siguiente ejemplo de código, se muestra cómo usar `refresh-cache`.

### AWS CLI

Para actualizar la caché de archivos compartidos

En el siguiente `refresh-cache` ejemplo, se actualiza la caché del recurso compartido de archivos especificado.

```
aws storagegateway refresh-cache \  
  --file-share-arn arn:aws:storagegateway:us-east-1:111122223333:share/  
share-2FA12345
```

Salida:

```
{  
  "FileShareARN": "arn:aws:storagegateway:us-east-1:111122223333:share/  
share-2FA12345",  
  "NotificationId": "4954d4b1-abcd-ef01-1234-97950a7d3483"  
}
```

Para obtener más información, consulte [ListFileShares](#) la API Referencia del servicio AWS Storage Gateway.

- Para API obtener más información, consulte [RefreshCache](#) la Referencia de AWS CLI comandos.

## AWS STS ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS STS.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **assume-role-with-saml**

En el siguiente ejemplo de código, se muestra cómo usar `assume-role-with-saml`.

#### AWS CLI

Para obtener credenciales a corto plazo para un rol, autentíquese con SAML

El siguiente `assume-role-with-saml` comando recupera un conjunto de credenciales de corta duración para el IAM rol. `TestSaml` La solicitud de este ejemplo se autentica mediante la SAML afirmación proporcionada por el proveedor de identidad al autenticarse ante él.

```
aws sts assume-role-with-saml \  
  --role-arn arn:aws:iam::123456789012:role/TestSaml \  
  --principal-arn arn:aws:iam::123456789012:saml-provider/SAML-test \  
  --saml-  
  assertion "VERYLONGENCODEDASSERTIONEXAMPLExzYW1s0kF1ZG1LbmNlPmJsYW5rPC9zYW1s0kF1ZG1LbmNlPjwv  
+PHNhbWw6TmFtZULEIEZvcm1hdD0idXJu0m9hc2lz0m5hbWVz0nRj0lNBTUw6Mi4w0m5hbWVpZC1mb3JtYXQ6dHJhbnM  
+PHNhbWw6U3ViamVjdENvbmZpcm1hdGlvbiBNZXRob2Q9InVybJpvYXNpczpuYW1lc3p0YzptQU1MOjIuMDpjbTpiZWwv"
```

Salida:

```
{  
  "Issuer": "https://integ.example.com/idp/shibboleth</Issuer",  
  "AssumedRoleUser": {  
    "Arn": "arn:aws:sts::123456789012:assumed-role/TestSaml",  
    "AssumedRoleId": "AR0456EXAMPLE789:TestSaml"  
  },  
}
```

```

    "Credentials": {
      "AccessKeyId": "ASIAV3ZUEFP6EXAMPLE",
      "SecretAccessKey": "8P+SQvWIuLnKhh8d++jpw0nNmQRBZvNEXAMPLEKEY",
      "SessionToken": "IQoJb3JpZ2luX2VjE0z//////////
wEXAMPLEtMSJHMEUCIDoKK3JH9uGQE1z0sINr5M4jk
+Na8KHdCqYRVjJCZEv0AiEA30vJGtw1EcVi0leS2vhs8VdCKFJQWPQrmGdeehM4IC1NtBmUpp2wUE8phUZampKsburED
+xo0rKwT38xVqr7ZD0u0iPPkUL64lIZbqBAz
+scqKmlzm8FDrypNC9Yjc8fP0Ln9FX9KSYvKTr4rvx3iSILtJabIQwj2ICCR/oLxBA==",
      "Expiration": "2019-11-01T20:26:47Z"
    },
    "Audience": "https://signin.aws.amazon.com/saml",
    "SubjectType": "transient",
    "PackedPolicySize": "6",
    "NameQualifier": "SbdG0nUkh1i4+EXAMPLExL/jEvs=",
    "Subject": "SamlExample"
  }
}

```

Para obtener más información, consulte [Solicitud de credenciales de seguridad temporales](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [AssumeRoleWithSaml](#) la Referencia de AWS CLI comandos.

## assume-role-with-web-identity

En el siguiente ejemplo de código, se muestra cómo usar `assume-role-with-web-identity`.

### AWS CLI

Para obtener credenciales a corto plazo para un rol autenticado con Web Identity (OAuth2."0)

El siguiente `assume-role-with-web-identity` comando recupera un conjunto de credenciales a corto plazo para el rol. IAM `app1` La solicitud se autentica mediante el token de identidad web proporcionado por el proveedor de identidad web especificado. Se aplican dos políticas adicionales a la sesión para restringir aún más lo que el usuario puede hacer. Las credenciales devueltas caducan una hora después de haberse generado.

```

aws sts assume-role-with-web-identity \
  --duration-seconds 3600 \
  --role-session-name "app1" \
  --provider-id "www.amazon.com" \

```

```
--policy-arns "arn:aws:iam::123456789012:policy/
q=webidentitydemopolicy1","arn:aws:iam::123456789012:policy/webidentitydemopolicy2"
\
--role-arn arn:aws:iam::123456789012:role/FederatedWebIdentityRole \
--web-identity-token "Atza
%7CIQEBljAsAhRFiXuWpUXuRvQ9PZL3GMFcYevydwIUFAHZwXZXXXXXXXXXJnrulxKDHwy87oGKPznh0D6bEQZTSCzyoC
CrKqjG7nPBjNIL016GGvuS5gSvPRUxWES3VYfm1wL7WTI7jn-Pcb6M-
buCgHhF0zTQxod27L9Cqn0Lio7N3gZAGpsp6n1-
AJBOCJckcyXe2c6uD0sr0JeZLKUm2eTDVMf8IehDVI0r1Q0nTV6KzzAI30Y87Vd_cVMQ"
```

Salida:

```
{
  "SubjectFromWebIdentityToken": "amzn1.account.AF6RH07KZU5XRVQJGXX6HB56KR2A"
  "Audience": "client.5498841531868486423.1548@apps.example.com",
  "AssumedRoleUser": {
    "Arn": "arn:aws:sts::123456789012:assumed-role/FederatedWebIdentityRole/
app1",
    "AssumedRoleId": "AROACLKWSQRAOEXAMPLE:app1"
  }
  "Credentials": {
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "SecretAccessKey": "wJa1rXUtnFEMI/K7MDENG/bPxrFiCYzEXAMPLEKEY",
    "SessionToken": "AQoEXAMPLEH4aoAH0gNCAPyJxz4B1CFFxWNE1OPTgk5TthT
+FvwqnKwRc0IfrrRh3c/LTo6UddyJw00vEVPvLXCrrrUtdnniCEXAMPLE/
IvU1dYUg2RVAJBanLiHb4IgrmpRV3zrkuWJ0gQs8IZZaIv2BXIa2R401gkBN9bkUDNCJiBeb/
AXlzBBko7b15fjrBs2+cTQtpZ3CYWFXG8C5zqx37wn0E49mRl/+0tkIKG07fAE",
    "Expiration": "2020-05-19T18:06:10+00:00"
  },
  "Provider": "www.amazon.com"
}
```

Para obtener más información, consulte [Solicitud de credenciales de seguridad temporales](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [AssumeRoleWithWebIdentity](#) la Referencia de AWS CLI comandos.

## assume-role

En el siguiente ejemplo de código, se muestra cómo usar `assume-role`.

## AWS CLI

### Cómo asumir un rol

El siguiente `assume-role` comando recupera un conjunto de credenciales a corto plazo para el IAM rol `s3-access-example`.

```
aws sts assume-role \
  --role-arn arn:aws:iam::123456789012:role/xaccounts3access \
  --role-session-name s3-access-example
```

Salida:

```
{
  "AssumedRoleUser": {
    "AssumedRoleId": "AROA3XFRBF535PLBIFPI4:s3-access-example",
    "Arn": "arn:aws:sts::123456789012:assumed-role/xaccounts3access/s3-access-example"
  },
  "Credentials": {
    "SecretAccessKey": "9drTJvcXLB89EXAMPLEELB8923FB892xMFI",
    "SessionToken": "AQoXdzELDDY//////////
wEaoAK1wvxJY12r2IrDFT2IvAzTCn3zHoZ7YNtpiQLF0MqZye/qwjzP2iEXAMPLEbw/
m3hsj8VBTkPORGvr9jM5sgP+w9IZWZnU+LWhmg
+a5fDi2oTGUYcdg9uexQ4mtCHIhfi4citgqZTgco40Yqr4lIlo4V2b2Dyauk0eYFNebHtY1FVgAUj
+7Indz3LU0aTWk1WKIjHmMCIoTkyYp/k7kUG7moeEYKSitwQIi6Gjn+nyzM
+PtoA3685ixzv0R7i5rjQi0YE0lf1oeie3bDiNHncmzosRM6SFiPzSvp6h/32xQuZsjcypmwsPSDtTPYcs0+YN/8BRi2
IcixSpnWEXAMPLEXSDFTAQAM6Dl9zR0tXoybnlrZIwMLlMi1Kcgo50ytwU=",
    "Expiration": "2016-03-15T00:05:07Z",
    "AccessKeyId": "ASIAJEXAMPLEXEG2JICEA"
  }
}
```

El resultado del comando contiene una clave de acceso, una clave secreta y un token de sesión que puede utilizar para autenticarse con AWS.

Para AWS CLI su uso, puede configurar un perfil con nombre asociado a un rol. Cuando utilices el perfil, AWS CLI llamará a `assume-role` y gestionará las credenciales por ti. Para obtener más información, consulte [Uso de un IAM rol AWS CLI en la Guía del AWS CLI usuario](#).

- Para API obtener más información, consulte [AssumeRole](#) la Referencia de AWS CLI comandos.



## decode-authorization-message

En el siguiente ejemplo de código, se muestra cómo usar `decode-authorization-message`.

### AWS CLI

Cómo decodificar un mensaje de autorización codificado devuelto en respuesta a una solicitud

En el siguiente ejemplo de `decode-authorization-message`, se decodifica información adicional sobre el estado de autorización de una solicitud de un mensaje codificado que regresa como respuesta a una solicitud de Amazon Web Services.

```
aws sts decode-authorization-message \
  --encoded-message EXAMPLEWodyRNrtLQARDip-
eTA6i6DrLUhHhPQrLWB_LAb15pAKx19mPDLexYcGBreyIKQC1BGBIpBKr3dFDkwqe07e2NMk5j_hmzAiChJN-8oy3Ewi
Ojau7BMj0TWw0tHPHv_Zaz87yENDipr745EjQwRd5LaoL3vN8_5ZfA9UiBMKDGvh1gjqZJFUiQoubv78V1RbHNYnK44E
p0u3FZjwYStfvTb3GHs3-6rLribG09jZ0tkfE6vqx1FzLyeDr4P2ihC1wty9tArCvvGzIAUNmARQJ2VWVPxioqgoqCz
JWP5pwe_mAyqh0NLw-r1S56YC_90onj9A80sNrHLI-
tIiNd7tgNTYzDuPQYD2FMDBnp82V9eVmYGtPp5NIeSpuf3f0HanFuBZgENxZQZ2dLH3xJGMTtYayzZrRXjiq_SfX9zeB
FaoPIb8LmmKVBLpIB0iFhU9sEHPqKHVPi6jdxXqKaZaFGvYVmVOiuQdNQKuyk0p067P0FrZECLjj0tNPBOZCcuEKEXAM
```

Salida:

```
{
  "DecodedMessage": "{\"allowed\":false,\"explicitDeny\":true,\"matchedStatements\
\":{\n\"items\":[\n\"statementId\":\n\"VisualEditor0\", \"effect\":\n\"DENY\", \"principals\
\":{\n\"items\":[\n\"value\":\n\"AROA123456789EXAMPLE\" ]}], \"principalGroups\
\":{\n\"items\":[\n ]], \"actions\":{\n\"items\":[\n\"value\":\n\"ec2:RunInstances\
\" ]}], \"resources\":{\n\"items\":[\n\"value\":\n\"*\" ]}], \"conditions\":{\n\"items\
\":[\n ] ]}], \"failures\":{\n\"items\":[\n ] ], \"context\":{\n\"principal\":{\n\"id\":
\n\"AROA123456789EXAMPLE:Ana\", \"arn\":\n\"arn:aws:sts::111122223333:assumed-role/
Developer/Ana\" }, \"action\":\n\"RunInstances\", \"resource\":\n\"arn:aws:ec2:us-
east-1:111122223333:instance/*\", \"conditions\":{\n\"items\":[\n\"key\":
\n\"ec2:MetadataHttpResponseHopLimit\", \"values\":{\n\"items\":[\n\"value\":
\n\"2\" ] ]}], {\n\"key\":\n\"ec2:InstanceMarketType\", \"values\":{\n\"items\":[\n\"value\
\":\n\"on-demand\" ] ]}], {\n\"key\":\n\"aws:Resource\", \"values\":{\n\"items\":[\n\"value\
\":\n\"instance/*\" ] ]}], {\n\"key\":\n\"aws:Account\", \"values\":{\n\"items\":[\n\"value\
\":\n\"111122223333\" ] ]}], {\n\"key\":\n\"ec2:AvailabilityZone\", \"values\":{\n\"items\":
[\n\"value\":\n\"us-east-1f\" ] ]}], {\n\"key\":\n\"ec2:epsOptimized\", \"values\":{\n\"items\
\":[\n\"value\":\n\"false\" ] ]}], {\n\"key\":\n\"ec2:IsLaunchTemplateResource\", \"values\
\":{\n\"items\":[\n\"value\":\n\"false\" ] ]}], {\n\"key\":\n\"ec2:InstanceType\", \"values\":
{\n\"items\":[\n\"value\":\n\"t2.micro\" ] ]}], {\n\"key\":\n\"ec2:RootDeviceType\", \"values
```

```

\":{\\"items\\":[{\\"value\\":\\"ebs\\"}]},{\\"key\\":\\"aws:Region\\",\\"values\\":{\\"items\\":[{\\"value\\":\\"us-east-1\\"}]},{\\"key\\":\\"ec2:MetadataHttpEndpoint\\",\\"values\\":{\\"items\\":[{\\"value\\":\\"enabled\\"}]},{\\"key\\":\\"aws:Service\\",\\"values\\":{\\"items\\":[{\\"value\\":\\"ec2\\"}]},{\\"key\\":\\"ec2:InstanceID\\",\\"values\\":{\\"items\\":[{\\"value\\":\\"*\\"}]},{\\"key\\":\\"ec2:MetadataHttpTokens\\",\\"values\\":{\\"items\\":[{\\"value\\":\\"required\\"}]},{\\"key\\":\\"aws:Type\\",\\"values\\":{\\"items\\":[{\\"value\\":\\"instance\\"}]},{\\"key\\":\\"ec2:Tenancy\\",\\"values\\":{\\"items\\":[{\\"value\\":\\"default\\"}]},{\\"key\\":\\"ec2:Region\\",\\"values\\":{\\"items\\":[{\\"value\\":\\"us-east-1\\"}]},{\\"key\\":\\"aws:ARN\\",\\"values\\":{\\"items\\":[{\\"value\\":\\"arn:aws:ec2:us-east-1:111122223333:instance/*\\"}]}}]}"}
}

```

Para obtener más información, consulte la [lógica de evaluación de políticas](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [DecodeAuthorizationMessage](#) la Referencia de AWS CLI comandos.

## get-caller-identity

En el siguiente ejemplo de código, se muestra cómo usar `get-caller-identity`.

### AWS CLI

Para obtener detalles sobre la IAM identidad actual

El siguiente `get-caller-identity` comando muestra información sobre la IAM identidad utilizada para autenticar la solicitud. La persona que llama es un IAM usuario.

```
aws sts get-caller-identity
```

Salida:

```

{
  "UserId": "AIDASAMPLEUSERID",
  "Account": "123456789012",
  "Arn": "arn:aws:iam::123456789012:user/DevAdmin"
}

```

- Para API obtener más información, consulte [GetCallerIdentity](#) la Referencia de AWS CLI comandos.

## get-federation-token

En el siguiente ejemplo de código, se muestra cómo usar `get-federation-token`.

### AWS CLI

Para devolver un conjunto de credenciales de seguridad temporales utilizando las credenciales clave IAM de acceso del usuario

En el siguiente ejemplo de `get-federation-token`, se devuelve un conjunto de credenciales de seguridad temporales (que consiste en un ID de clave de acceso, una clave de acceso secreta y un token de seguridad). Debe llamar a la `GetFederationToken` operación con las credenciales de seguridad a largo plazo de un IAM usuario.

```
aws sts get-federation-token \  
  --name Bob \  
  --policy file://myfile.json \  
  --policy-arns arn=arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess \  
  --duration-seconds 900
```

Contenidos de `myfile.json`:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": "ec2:Describe*",  
      "Resource": "*"   
    },  
    {  
      "Effect": "Allow",  
      "Action": "elasticloadbalancing:Describe*",  
      "Resource": "*"   
    },  
    {  
      "Effect": "Allow",  
      "Action": [  
        "cloudwatch:ListMetrics",  
        "cloudwatch:GetMetricStatistics",  
        "cloudwatch:Describe*"   
      ],  
      "Resource": "*"   
    }   
  ]   
}
```

```

    },
    {
      "Effect": "Allow",
      "Action": "autoscaling:Describe*",
      "Resource": "*"
    }
  ]
}

```

Salida:

```

{
  "Credentials": {
    "AccessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY",
    "SessionToken": "EXAMPLEpZ21uX2VjEGoaCXVzLXd1c3QtMiJIMEYCIQC/
W9pL5ArQyDD5JwFL3/h5+WGopQ24GEXweNctwhi9sgIhAMkg
+MZE35iWM8s4r5Lr25f9rSTVPFH98G42QunWMTfKq0DCOP//////////
wEQAxoMNDUy0TI1MTcwNTA3Igxuy3A0puuoLsk3MJwqgQPg8Q0d9HuoClUxq26wnc/nm
+eZLjHdyGf2KUAHK2DuaS/nrGSEXAMPLE",
    "Expiration": "2023-12-20T02:06:07+00:00"
  },
  "FederatedUser": {
    "FederatedUserId": "111122223333:Bob",
    "Arn": "arn:aws:sts::111122223333:federated-user/Bob"
  },
  "PackedPolicySize": 36
}

```

Para obtener más información, consulte [Solicitud de credenciales de seguridad temporales](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetFederationToken](#) la Referencia de AWS CLI comandos.

## get-session-token

En el siguiente ejemplo de código, se muestra cómo usar `get-session-token`.

### AWS CLI

Para obtener un conjunto de credenciales a corto plazo para una IAM identidad

El siguiente `get-session-token` comando recupera un conjunto de credenciales de corta duración para la IAM identidad que realiza la llamada. Las credenciales resultantes se pueden usar para solicitudes en las que la política exija la autenticación multifactorial (MFA). Las credenciales caducan 15 minutos después de haberse generado.

```
aws sts get-session-token \  
  --duration-seconds 900 \  
  --serial-number "YourMFADeviceSerialNumber" \  
  --token-code 123456
```

Salida:

```
{  
  "Credentials": {  
    "AccessKeyId": "ASIAIOSFODNN7EXAMPLE",  
    "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxrFiCYzEXAMPLEKEY",  
    "SessionToken": "AQoEXAMPLEH4aoAH0gNCAPyJxz4B1CFFxWNE1OPTgk5TthT  
+FvwqnKwRc0IfrrRh3c/LTo6UDdyJw00vEVPvLXCrrrUtdnniCEXAMPLE/  
IvU1dYUg2RVAJBanLiHb4IgrmpRV3zrkuWJ0gQs8IZZaIv2BXIa2R40lgkBN9bkUDNCJiBeb/  
AX1zBBko7b15fjrBs2+cTQtpZ3CYWFXG8C5zqx37wn0E49mRl/+0tkIKG07fAE",  
    "Expiration": "2020-05-19T18:06:10+00:00"  
  }  
}
```

Para obtener más información, consulte [Solicitud de credenciales de seguridad temporales](#) en la Guía del AWS IAM usuario.

- Para API obtener más información, consulte [GetSessionToken](#) la Referencia de AWS CLI comandos.

## AWS Support ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS Support.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **add-attachments-to-set**

En el siguiente ejemplo de código, se muestra cómo usar `add-attachments-to-set`.

#### AWS CLI

Para añadir un archivo adjunto a un conjunto

En el siguiente `add-attachments-to-set` ejemplo, se agrega una imagen a un conjunto que, a continuación, puede especificar para un caso de soporte en su AWS cuenta.

```
aws support add-attachments-to-set \  
  --attachment-set-id "as-2f5a6faa2a4a1e600-mu-nk5xQ1Br70-  
G1cUos5LZkd38K0AHZa9BMDVzNEXAMPLE" \  
  --attachments fileName=troubleshoot-screenshot.png,data=base64-encoded-string
```

Salida:

```
{  
  "attachmentSetId": "as-2f5a6faa2a4a1e600-mu-nk5xQ1Br70-  
G1cUos5LZkd38K0AHZa9BMDVzNEXAMPLE",  
  "expiryTime": "2020-05-14T17:04:40.790+0000"  
}
```

Para obtener más información, consulte [Administración de casos](#) en la Guía del usuario de soporte de AWS .

- Para API obtener más información, consulta [AddAttachmentsToSet](#) la Referencia de AWS CLI comandos.

### **add-communication-to-case**

En el siguiente ejemplo de código, se muestra cómo usar `add-communication-to-case`.

## AWS CLI

Para añadir una comunicación a un caso

En el siguiente `add-communication-to-case` ejemplo, se agregan comunicaciones a un caso de soporte de tu AWS cuenta.

```
aws support add-communication-to-case \  
  --case-id "case-12345678910-2013-c4c1d2bf33c5cf47" \  
  --communication-body "I'm attaching a set of images to this case." \  
  --cc-email-addresses "myemail@example.com" \  
  --attachment-set-id "as-2f5a6faa2a4a1e600-mu-nk5xQlBr70-  
G1cUos5LZkd38K0AHZa9BMDVzNEXAMPLE"
```

Salida:

```
{  
  "result": true  
}
```

Para obtener más información, consulte [Administración de casos](#) en la Guía del usuario de soporte de AWS .

- Para API obtener más información, consulte [AddCommunicationToCase](#) la Referencia de AWS CLI comandos.

## create-case

En el siguiente ejemplo de código, se muestra cómo usar `create-case`.

## AWS CLI

Creación de un caso

En el siguiente `create-case` ejemplo, se crea un caso de soporte para su AWS cuenta.

```
aws support create-case \  
  --category-code "using-aws" \  
  --cc-email-addresses "myemail@example.com" \  
  --communication-body "I want to learn more about an AWS service." \  
  --issue-type "technical" \  
  --language "en" \  
  --attachment-set-id "as-2f5a6faa2a4a1e600-mu-nk5xQlBr70-  
G1cUos5LZkd38K0AHZa9BMDVzNEXAMPLE"
```

```
--service-code general-info \  
--severity-code low \  
--subject Question about my account
```

Salida:

```
{  
  "caseId": "case-12345678910-2013-c4c1d2bf33c5cf47"  
}
```

Para obtener más información, consulte [Administración de casos](#) en la Guía del usuario de soporte de AWS .

- Para API obtener más información, consulte [CreateCase](#) la Referencia de AWS CLI comandos.

## describe-attachment

En el siguiente ejemplo de código, se muestra cómo usar `describe-attachment`.

### AWS CLI

Descripción de un archivo adjunto

El siguiente ejemplo de `describe-attachment` devuelve información sobre el archivo adjunto con el ID especificado.

```
aws support describe-attachment \  
  --attachment-id attachment-KBnjRNrePd9D6Jx0-Mm00xZuDEaL2JAj_0-  
gJv9qqDooTipsz3V1Nb19rCfkZneeQeDPgp8X1iVJyHH7UuhZDdNeqGoduZsPrAhyMakqLc60-  
iJjL5HqyYGiT1FG8EXAMPLE
```

Salida:

```
{  
  "attachment": {  
    "fileName": "troubleshoot-screenshot.png",  
    "data": "base64-blob"  
  }  
}
```

Para obtener más información, consulte [Administración de casos](#) en la Guía del usuario de soporte de AWS .



- Para API obtener más información, consulte [DescribeAttachment](#) la Referencia de AWS CLI comandos.

## describe-cases

En el siguiente ejemplo de código, se muestra cómo usar `describe-cases`.

### AWS CLI

#### Descripción de un caso

El siguiente `describe-cases` ejemplo devuelve información sobre el caso de soporte especificado en su AWS cuenta.

```
aws support describe-cases \  
  --display-id "1234567890" \  
  --after-time "2020-03-23T21:31:47.774Z" \  
  --include-resolved-cases \  
  --language "en" \  
  --no-include-communications \  
  --max-item 1
```

#### Salida:

```
{  
  "cases": [  
    {  
      "status": "resolved",  
      "ccEmailAddresses": [],  
      "timeCreated": "2020-03-23T21:31:47.774Z",  
      "caseId": "case-12345678910-2013-c4c1d2bf33c5cf47",  
      "severityCode": "low",  
      "language": "en",  
      "categoryCode": "using-aws",  
      "serviceCode": "general-info",  
      "submittedBy": "myemail@example.com",  
      "displayId": "1234567890",  
      "subject": "Question about my account"  
    }  
  ]  
}
```

Para obtener más información, consulte [Administración de casos](#) en la Guía del usuario de soporte de AWS .

- Para API obtener más información, consulte [DescribeCases](#) la Referencia de AWS CLI comandos.

## describe-communications

En el siguiente ejemplo de código, se muestra cómo usar `describe-communications`.

### AWS CLI

Descripción de la última comunicación de un caso

En el siguiente `describe-communications` ejemplo, se devuelve la última comunicación del caso de soporte especificado en su AWS cuenta.

```
aws support describe-communications \
  --case-id "case-12345678910-2013-c4c1d2bf33c5cf47" \
  --after-time "2020-03-23T21:31:47.774Z" \
  --max-item 1
```

Salida:

```
{
  "communications": [
    {
      "body": "I want to learn more about an AWS service.",
      "attachmentSet": [],
      "caseId": "case-12345678910-2013-c4c1d2bf33c5cf47",
      "timeCreated": "2020-05-12T23:12:35.000Z",
      "submittedBy": "Amazon Web Services"
    }
  ],
  "NextToken": "eyJmZXRhZG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQEXAMPLE=="
}
```

Para obtener más información, consulte [Administración de casos](#) en la Guía del usuario de soporte de AWS .

- Para API obtener más información, consulte [DescribeCommunications](#) la Referencia de AWS CLI comandos.

## describe-services

En el siguiente ejemplo de código, se muestra cómo usar `describe-services`.

### AWS CLI

Para enumerar AWS los servicios y las categorías de servicios

En el siguiente ejemplo de `describe-services` se enumeran las categorías de servicios disponibles para solicitar información general.

```
aws support describe-services \
  --service-code-list general-info
```

Salida:

```
{
  "services": [
    {
      "code": "general-info",
      "name": "General Info and Getting Started",
      "categories": [
        {
          "code": "charges",
          "name": "How Will I Be Charged?"
        },
        {
          "code": "gdpr-queries",
          "name": "Data Privacy Query"
        },
        {
          "code": "reserved-instances",
          "name": "Reserved Instances"
        },
        {
          "code": "resource",
          "name": "Where is my Resource?"
        },
        {
          "code": "using-aws",
          "name": "Using AWS & Services"
        }
      ]
    }
  ]
}
```

```
        "code": "free-tier",
        "name": "Free Tier"
    },
    {
        "code": "security-and-compliance",
        "name": "Security & Compliance"
    },
    {
        "code": "account-structure",
        "name": "Account Structure"
    }
]
}
```

Para obtener más información, consulte [Administración de casos](#) en la Guía del usuario de soporte de AWS .

- Para API obtener más información, consulte [DescribeServices](#) la Referencia de AWS CLI comandos.

## describe-severity-levels

En el siguiente ejemplo de código, se muestra cómo usar `describe-severity-levels`.

### AWS CLI

Creación de una lista de los niveles de gravedad disponibles

En el siguiente ejemplo de `describe-severity-levels` se enumeran los niveles de gravedad disponibles para un caso de soporte.

```
aws support describe-severity-levels
```

Salida:

```
{
  "severityLevels": [
    {
      "code": "low",
```

```
    "name": "Low"
  },
  {
    "code": "normal",
    "name": "Normal"
  },
  {
    "code": "high",
    "name": "High"
  },
  {
    "code": "urgent",
    "name": "Urgent"
  },
  {
    "code": "critical",
    "name": "Critical"
  }
]
```

Para obtener más información, consulte [Elección de la gravedad](#) en la Guía del usuario de soporte de AWS .

- Para API obtener más información, consulte [DescribeSeverityLevels](#) la Referencia de AWS CLI comandos.

## **describe-trusted-advisor-check-refresh-statuses**

En el siguiente ejemplo de código, se muestra cómo usar `describe-trusted-advisor-check-refresh-statuses`.

### AWS CLI

Para enumerar los estados de actualización de las comprobaciones de AWS Trusted Advisor

El siguiente `describe-trusted-advisor-check-refresh-statuses` ejemplo muestra los estados de actualización de dos comprobaciones de Trusted Advisor: Amazon S3 Bucket Permissions y IAM Use.

```
aws support describe-trusted-advisor-check-refresh-statuses \
  --check-id "Pfx0RwqBli" "zXCkfM1nI3"
```

Salida:

```
{
  "statuses": [
    {
      "checkId": "Pfx0RwqBli",
      "status": "none",
      "millisUntilNextRefreshable": 0
    },
    {
      "checkId": "zXCkfM1nI3",
      "status": "none",
      "millisUntilNextRefreshable": 0
    }
  ]
}
```

Para obtener más información, consulte [AWS Trusted Advisor](#) en la Guía del usuario de AWS Support.

- Para API obtener más información, consulte [DescribeTrustedAdvisorCheckRefreshStatuses](#) la Referencia de AWS CLI comandos.

## **describe-trusted-advisor-check-result**

En el siguiente ejemplo de código, se muestra cómo usar `describe-trusted-advisor-check-result`.

AWS CLI

Para enumerar los resultados de una comprobación de AWS Trusted Advisor

En el siguiente `describe-trusted-advisor-check-result` ejemplo, se muestran los resultados de la comprobación de IAM uso.

```
aws support describe-trusted-advisor-check-result \
  --check-id "zXCkfM1nI3"
```

Salida:

```
{
  "result": {
```

```
"checkId": "zXCkfM1nI3",
"timestamp": "2020-05-13T21:38:05Z",
"status": "ok",
"resourcesSummary": {
  "resourcesProcessed": 1,
  "resourcesFlagged": 0,
  "resourcesIgnored": 0,
  "resourcesSuppressed": 0
},
"categorySpecificSummary": {
  "costOptimizing": {
    "estimatedMonthlySavings": 0.0,
    "estimatedPercentMonthlySavings": 0.0
  }
},
"flaggedResources": [
  {
    "status": "ok",
    "resourceId": "47DEQpj8HBSa-_TImW-5JCeuQeRkm5NMpJWZEXAMPLE",
    "isSuppressed": false
  }
]
}
```

Para obtener más información, consulte [AWS Trusted Advisor](#) en la Guía del usuario de AWS Support.

- Para API obtener más información, consulte [DescribeTrustedAdvisorCheckResult](#) la Referencia de AWS CLI comandos.

## **describe-trusted-advisor-check-summaries**

En el siguiente ejemplo de código, se muestra cómo usar `describe-trusted-advisor-check-summaries`.

### AWS CLI

Para enumerar los resúmenes de las comprobaciones de AWS Trusted Advisor

El siguiente `describe-trusted-advisor-check-summaries` ejemplo muestra los resultados de dos comprobaciones de Trusted Advisor: Amazon S3 Bucket Permissions y IAM Use.

```
aws support describe-trusted-advisor-check-summaries \  
--check-ids "Pfx0RwqBli" "zXCkfM1nI3"
```

Salida:

```
{  
  "summaries": [  
    {  
      "checkId": "Pfx0RwqBli",  
      "timestamp": "2020-05-13T21:38:12Z",  
      "status": "ok",  
      "hasFlaggedResources": true,  
      "resourcesSummary": {  
        "resourcesProcessed": 44,  
        "resourcesFlagged": 0,  
        "resourcesIgnored": 0,  
        "resourcesSuppressed": 0  
      },  
      "categorySpecificSummary": {  
        "costOptimizing": {  
          "estimatedMonthlySavings": 0.0,  
          "estimatedPercentMonthlySavings": 0.0  
        }  
      }  
    },  
    {  
      "checkId": "zXCkfM1nI3",  
      "timestamp": "2020-05-13T21:38:05Z",  
      "status": "ok",  
      "hasFlaggedResources": true,  
      "resourcesSummary": {  
        "resourcesProcessed": 1,  
        "resourcesFlagged": 0,  
        "resourcesIgnored": 0,  
        "resourcesSuppressed": 0  
      },  
      "categorySpecificSummary": {  
        "costOptimizing": {  
          "estimatedMonthlySavings": 0.0,  
          "estimatedPercentMonthlySavings": 0.0  
        }  
      }  
    }  
  ]  
}
```



```
]
}
```

Para obtener más información, consulte [AWS Trusted Advisor](#) en la Guía del usuario de AWS Support.

- Para API obtener más información, consulte [DescribeTrustedAdvisorCheckSummaries](#) la Referencia de AWS CLI comandos.

## describe-trusted-advisor-checks

En el siguiente ejemplo de código, se muestra cómo usar `describe-trusted-advisor-checks`.

### AWS CLI

Para enumerar las comprobaciones AWS de Trusted Advisor disponibles

El siguiente `describe-trusted-advisor-checks` ejemplo muestra los cheques de Trusted Advisor disponibles en su AWS cuenta. Esta información incluye el nombre, el identificador, la descripción, la categoría y los metadatos del cheque. Tenga en cuenta que el resultado está abreviado para facilitar la lectura.

```
aws support describe-trusted-advisor-checks \
  --language "en"
```

Salida:

```
{
  "checks": [
    {
      "id": "zXCkFM1nI3",
      "name": "IAM Use",
      "description": "Checks for your use of AWS Identity and Access
Management (IAM). You can use IAM to create users, groups, and roles in AWS, and
you can use permissions to control access to AWS resources. \n<br>\n<br>\n<b>Alert
Criteria</b><br>\nYellow: No IAM users have been created for this account.\n<br>
\n<br>\n<b>Recommended Action</b><br>\nCreate one or more IAM users and groups in
your account. You can then create additional users whose permissions are limited
to perform specific tasks in your AWS environment. For more information, see <a
href=\"https://docs.aws.amazon.com/IAM/latest/UserGuide/IAMGettingStarted.html\"
target=\"_blank\">Getting Started</a>. \n<br><br>\n<b>Additional Resources</b><br>
```

```
\n<a href=\"https://docs.aws.amazon.com/IAM/latest/UserGuide/IAM_Introduction.html\"  
  target=\"_blank\">What Is IAM?</a>\",  
    \"category\": \"security\",  
    \"metadata\": []  
  }  
]  
}
```

Para obtener más información, consulte [AWS Trusted Advisor](#) en la Guía del usuario de AWS Support.

- Para API obtener más información, consulte [DescribeTrustedAdvisorChecks](#) la Referencia de AWS CLI comandos.

## refresh-trusted-advisor-check

En el siguiente ejemplo de código, se muestra cómo usar `refresh-trusted-advisor-check`.

### AWS CLI

Para actualizar una comprobación AWS de Trusted Advisor

En el siguiente `refresh-trusted-advisor-check` ejemplo, se actualiza la comprobación de Amazon S3 Bucket Permissions Trusted Advisor de su AWS cuenta.

```
aws support refresh-trusted-advisor-check \  
  --check-id "Pfx0RwqBli"
```

Salida:

```
{  
  \"status\": {  
    \"checkId\": \"Pfx0RwqBli\",  
    \"status\": \"enqueued\",  
    \"millisUntilNextRefreshable\": 3599992  
  }  
}
```

Para obtener más información, consulte [AWS Trusted Advisor](#) en la Guía del usuario de AWS Support.

- Para API obtener más información, consulte [RefreshTrustedAdvisorCheck](#) la Referencia de AWS CLI comandos.

## resolve-case

En el siguiente ejemplo de código, se muestra cómo usar `resolve-case`.

### AWS CLI

Resolución de un caso de soporte

El siguiente `resolve-case` ejemplo resuelve un caso de soporte en su AWS cuenta.

```
aws support resolve-case \  
  --case-id "case-12345678910-2013-c4c1d2bf33c5cf47"
```

Salida:

```
{  
  "finalCaseStatus": "resolved",  
  "initialCaseStatus": "work-in-progress"  
}
```

Para obtener más información, consulte [Administración de casos](#) en la Guía del usuario de soporte de AWS .

- Para API obtener más información, consulte [ResolveCase](#) la Referencia de AWS CLI comandos.

## SWFEjemplos de Amazon que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante AmazonSWF.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **count-closed-workflow-executions**

En el siguiente ejemplo de código, se muestra cómo usar `count-closed-workflow-executions`.

#### AWS CLI

Recuento de ejecuciones de flujos de trabajo cerrados

Se puede utilizar `swf count-closed-workflow-executions` para recuperar el número de ejecuciones de flujos de trabajo cerrados de un dominio determinado. Puede especificar filtros para contar clases específicas de ejecuciones.

Los `--start-time-filter` argumentos `--domain` y uno `--close-time-filter` o dos son obligatorios. Todos los demás argumentos son opcionales.

```
aws swf count-closed-workflow-executions \
  --domain DataFrobtzz \
  --close-time-filter "{ \"latestDate\" : 1377129600, \"oldestDate\" :
  1370044800 }"
```

Salida:

```
{
  "count": 2,
  "truncated": false
}
```

Si «truncado» es `true`, «recuento» representa el número máximo que puede devolver Amazon SWF. Los demás resultados se truncan.

Para reducir el número de resultados devueltos, puede:

modificar los `--start-time-filter` valores `--close-time-filter` o para reducir el intervalo de tiempo en el que se busca. Cada uno de ellos se excluye mutuamente: solo puede

especificar uno de ellos en una solicitud. Utilice los `--type-filter` argumentos `--close-status-filter` `--execution-filter`, `--tag-filter` o para filtrar aún más los resultados. Sin embargo, estos argumentos también se excluyen mutuamente.

Consulte también [CountClosedWorkflowExecutions](#) en la API referencia de Amazon Simple Workflow Service

- Para API obtener más información, consulte [CountClosedWorkflowExecutions](#) la Referencia de AWS CLI comandos.

## count-open-workflow-executions

En el siguiente ejemplo de código, se muestra cómo usar `count-open-workflow-executions`.

### AWS CLI

Recuento de ejecuciones de flujos de trabajo abiertos

Se puede utilizar `swf count-open-workflow-executions` para recuperar el número de ejecuciones de flujos de trabajo abiertos de un dominio determinado. Puede especificar filtros para contar clases específicas de ejecuciones.

Los `--start-time-filter` argumentos `--domain` y son obligatorios. Todos los demás argumentos son opcionales.

```
aws swf count-open-workflow-executions \  
  --domain DataFrobtzz \  
  --start-time-filter "{ \"latestDate\" : 1377129600, \"oldestDate\" :  
  1370044800 }"
```

Salida:

```
{  
  "count": 4,  
  "truncated": false  
}
```

Si «truncado» es `true`, «recuento» representa el número máximo que puede devolver Amazon SWF. Los demás resultados se truncan.

Para reducir el número de resultados devueltos, puede:

modifique los `--start-time-filter` valores para reducir el intervalo de tiempo en el que se busca. Utilice los `--type-filter` argumentos `--close-status-filter` `--execution-filter`, `--tag-filter` o para filtrar aún más los resultados. Cada uno de ellos se excluye mutuamente: solo puede especificar uno de ellos en una solicitud.

Para obtener más información, consulte `CountOpenWorkflowExecutions` la API referencia de Amazon Simple Workflow Service

- Para API obtener más información, consulte [CountOpenWorkflowExecutions](#) la Referencia de AWS CLI comandos.

## `deprecate-domain`

En el siguiente ejemplo de código, se muestra cómo usar `deprecate-domain`.

### AWS CLI

Cómo dejar en desuso un dominio

Para dar de baja un dominio (aunque todavía podrá verlo, no podrá crear nuevas ejecuciones de flujo de trabajo ni tipos de registro en él), use `swf deprecate-domain`. Solo tiene un parámetro necesario, `--name`, que toma el nombre del dominio que se va a descartar.

```
aws swf deprecate-domain \  
  --name MyNeatNewDomain ""
```

Al igual que ocurre con `register-domain`, no se devuelve ningún resultado. Sin embargo, si `list-domains` solía ver los dominios registrados, verá que el dominio ha quedado obsoleto y ya no aparece en los datos devueltos.

```
aws swf list-domains \  
  --registration-status REGISTERED  
  {  
    "domainInfos": [  
      {  
        "status": "REGISTERED",  
        "name": "DataFrobotz"  
      },  
      {  
        "status": "REGISTERED",  
        "name": "erontest"  
      }  
    ]  
  }
```

```
    ]
  }
```

Si lo usas `--registration-status DEPRECATED` con `list-domains`, verás tu dominio obsoleto.

```
aws swf list-domains \
  --registration-status DEPRECATED
  {
    "domainInfos": [
      {
        "status": "DEPRECATED",
        "name": "MyNeatNewDomain"
      }
    ]
  }
```

Puedes seguir utilizándolo `describe-domain` para obtener información sobre un dominio obsoleto.

```
aws swf describe-domain \
  --name MyNeatNewDomain
  {
    "domainInfo": {
      "status": "DEPRECATED",
      "name": "MyNeatNewDomain"
    },
    "configuration": {
      "workflowExecutionRetentionPeriodInDays": "0"
    }
  }
```

Consulte también [DeprecateDomain](#) en la API referencia de Amazon Simple Workflow Service

- Para API obtener más información, consulte [DeprecateDomain](#) la Referencia de AWS CLI comandos.

## describe-domain

En el siguiente ejemplo de código, se muestra cómo usar `describe-domain`.

## AWS CLI

### Obtener información sobre un dominio

Para obtener información detallada sobre un dominio en particular, utilice el `swf describe-domain` comando. Hay un parámetro necesario: `--name`, que lleva el nombre del dominio sobre el que desee información.

```
aws swf describe-domain \  
  --name DataFrobotz \  
  {  
    "domainInfo": {  
      "status": "REGISTERED",  
      "name": "DataFrobotz"  
    },  
    "configuration": {  
      "workflowExecutionRetentionPeriodInDays": "1"  
    }  
  }
```

También se puede utilizar `describe-domain` para obtener información sobre los dominios obsoletos.

```
aws swf describe-domain \  
  --name MyNeatNewDomain \  
  {  
    "domainInfo": {  
      "status": "DEPRECATED",  
      "name": "MyNeatNewDomain"  
    },  
    "configuration": {  
      "workflowExecutionRetentionPeriodInDays": "0"  
    }  
  }
```

Consulte también [DescribeDomain](#) en la API referencia de Amazon Simple Workflow Service

- Para API obtener más información, consulte [DescribeDomain](#) la Referencia de AWS CLI comandos.



## list-activity-types

En el siguiente ejemplo de código, se muestra cómo usar `list-activity-types`.

### AWS CLI

Listado de tipos de actividades

Para obtener una lista de los tipos de actividad de un dominio, utiliceswf `list-activity-types`. Los `--registration-status` argumentos `--domain` y son obligatorios.

```
aws swf list-activity-types \
  --domain DataFrobtzz \
  --registration-status REGISTERED
```

Salida:

```
{
  "typeInfos": [
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.451,
      "activityType": {
        "version": "1",
        "name": "confirm-user-email"
      },
      "description": "subscribe confirm-user-email activity"
    },
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.709,
      "activityType": {
        "version": "1",
        "name": "confirm-user-phone"
      },
      "description": "subscribe confirm-user-phone activity"
    },
    {
      "status": "REGISTERED",
      "creationDate": 1371454149.871,
      "activityType": {
        "version": "1",
        "name": "get-subscription-info"
      }
    }
  ]
}
```

```

    },
    "description": "subscribe get-subscription-info activity"
  },
  {
    "status": "REGISTERED",
    "creationDate": 1371454150.909,
    "activityType": {
      "version": "1",
      "name": "send-subscription-success"
    },
    "description": "subscribe send-subscription-success activity"
  },
  {
    "status": "REGISTERED",
    "creationDate": 1371454150.085,
    "activityType": {
      "version": "1",
      "name": "subscribe-user-sns"
    },
    "description": "subscribe subscribe-user-sns activity"
  }
]
}

```

Puede usar el `--name` argumento para seleccionar solo los tipos de actividad con un nombre determinado:

```

aws swf list-activity-types \
  --domain DataFrobtzz \
  --registration-status REGISTERED \
  --name "send-subscription-success"

```

Salida:

```

{
  "typeInfos": [
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.909,
      "activityType": {
        "version": "1",
        "name": "send-subscription-success"
      },
    },
  ],
}

```

```

        "description": "subscribe send-subscription-success activity"
      }
    ]
  }

```

Para recuperar los resultados en las páginas, puede establecer el `--maximum-page-size` argumento. Si se devuelven más resultados de los que caben en una página de resultados, se devolverá un `nextPageToken` «» en el conjunto de resultados:

```

aws swf list-activity-types \
  --domain DataFrobtzz \
  --registration-status REGISTERED \
  --maximum-page-size 2

```

Salida:

```

{
  "nextPageToken": "AAAAKgAAAAEAAAAAAAAA1Gp1BelJq
+PmHvAnDxJYbup8+0R4LVtbXLD17QNY7C30pHo9Sz06D/GuFz10yC73umBQ1t0PJ/gC/
aYpzDMqUIWIA1T9W0s2DrryZX40C/6Lhk9/
o5kdsuWMSBkHhgaZjgwp3WJINIFJFdaSMxY2vYAX7AtRtpcqJuBDDRE9RaRqDGyqIYUMltarkiqpSY1ZVveBasBvlvyU
WGAaqehiDz7/JzLT/wNNUM0d+Nhe",
  "typeInfos": [
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.451,
      "activityType": {
        "version": "1",
        "name": "confirm-user-email"
      },
      "description": "subscribe confirm-user-email activity"
    },
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.709,
      "activityType": {
        "version": "1",
        "name": "confirm-user-phone"
      },
      "description": "subscribe confirm-user-phone activity"
    }
  ]
}

```

```
}

```

Puede pasar el nextPageToken valor a la siguiente llamada del `--next-page-token` argumento y recuperar la siguiente página de resultados: `list-activity-types`

```
aws swf list-activity-types \
  --domain DataFrobtzz \
  --registration-status REGISTERED \
  --maximum-page-size 2 \
  --next-page-token "AAAAKgAAAAEAAAAAAAAAA1Gp1Be1Jq
+PmHvAnDxJYbup8+0R4LVtbXLD17QNY7C30pHo9Ssz06D/GuFz10yC73umBQ1t0PJ/gC/
aYpzDMqUIWIA1T9W0s2DryyZX40C/6Lhk9/
o5kdsuWMSBkHhgaZjgwp3WJINIFJFdaSMxY2vYAX7AtRtpcqJuBDDRE9RaRqDGYqIYUM1tarki qpSY1ZVveBasBvLvyU
WGAaqehiDz7/JzLT/wWNNUM0d+Nhe"
```

Salida:

```
{
  "nextPageToken": "AAAAKgAAAAEAAAAAAAAAAAw+7LZ4GRZPzTqBHsp2wBxWB8m1sgLCc1gCuq3J+h/
m3+v0fFqtkcjLwV5cc40jNAzTCuq/
XcylPumGwkjbjatqzpbq0cVnFjFxGoi0LB201bv0krbUISBv1pFPmSWpDSZJsxg5UxCcweteS1Fn1PNSZ/
MoinBZo80TkjMuzcsTuK0zH9wCaR8ITcALJ3SaqHU3pyIRS5hPmFA30LIc8zaAepj1aujo6hntNSCruB4"
  "typeInfos": [
    {
      "status": "REGISTERED",
      "creationDate": 1371454149.871,
      "activityType": {
        "version": "1",
        "name": "get-subscription-info"
      },
      "description": "subscribe get-subscription-info activity"
    },
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.909,
      "activityType": {
        "version": "1",
        "name": "send-subscription-success"
      },
      "description": "subscribe send-subscription-success activity"
    }
  ]
}
```

```
}

```

Si aún quedan más resultados por devolver, se devolverá `nextPageToken` «» con los resultados. Si no hay más páginas de resultados que devolver, `nextPageToken` no aparecerá en el conjunto de resultados.

Puede utilizar el `--reverse-order` argumento para invertir el orden de los resultados devueltos. Esto también afecta a los resultados paginados.

```
aws swf list-activity-types \
  --domain DataFrobtzz \
  --registration-status REGISTERED \
  --maximum-page-size 2 \
  --reverse-order

```

Salida:

```
{
  "nextPageToken": "AAAAKgAAAAEAAAAAAAAAAwXcpu5ePSyQkrC
+8WMbmSrenuZC2ZkIXQYBPB/b9xIOVkj+bMEFhGj0KmmJ4rF7iddhjf7UMYCsfGkEn7mk
+yMCgVc1JxDWmB0EH46bhcmclmYNQihMDmUwocpr7To6/R7CLu0St1gkFayx0idJXErQW0zdNfQaIWAnF/
cwioBbXlkz1fQzmDeU3M5oYGMPQIrUqkPq7pMEW0q0lK5eDN97NzFYdZZ/r1cLDWPZhUjY",
  "typeInfos": [
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.085,
      "activityType": {
        "version": "1",
        "name": "subscribe-user-sns"
      },
      "description": "subscribe subscribe-user-sns activity"
    },
    {
      "status": "REGISTERED",
      "creationDate": 1371454150.909,
      "activityType": {
        "version": "1",
        "name": "send-subscription-success"
      },
      "description": "subscribe send-subscription-success activity"
    }
  ]
}
```

```
}
```

Consulte también [ListActivityTypes](#) en la API referencia de Amazon Simple Workflow Service

- Para API obtener más información, consulte [ListActivityTypes](#) la Referencia de AWS CLI comandos.

## list-domains

En el siguiente ejemplo de código, se muestra cómo usar `list-domains`.

### AWS CLI

Ejemplo 1: Para enumerar sus dominios registrados

El siguiente ejemplo de `list-domains` comando muestra los REGISTERED SWF dominios que ha registrado para su cuenta.

```
aws swf list-domains \  
  --registration-status REGISTERED
```

Salida:

```
{  
  "domainInfos": [  
    {  
      "status": "REGISTERED",  
      "name": "DataFrobotz"  
    },  
    {  
      "status": "REGISTERED",  
      "name": "erontest"  
    }  
  ]  
}
```

Para obtener más información, consulte [ListDomains](#) la API referencia de Amazon Simple Workflow Service

Ejemplo 2: Para enumerar sus dominios obsoletos

El siguiente ejemplo de `list-domains` comando muestra los DEPRECATED SWF dominios que ha registrado para su cuenta. Los dominios obsoletos son dominios que no pueden registrar nuevos flujos de trabajo o actividades, pero que sí pueden consultarse.

```
aws swf list-domains \  
  --registration-status DEPRECATED
```

Salida:

```
{  
  "domainInfos": [  
    {  
      "status": "DEPRECATED",  
      "name": "MyNeatNewDomain"  
    }  
  ]  
}
```

Para obtener más información, consulte [ListDomains](#) la API referencia de Amazon Simple Workflow Service

Ejemplo 3: Para mostrar la primera página de dominios registrados

El siguiente ejemplo de `list-domains` comando muestra los REGISTERED SWF dominios de la primera página que ha registrado para su cuenta mediante la `--maximum-page-size` opción.

```
aws swf list-domains \  
  --registration-status REGISTERED \  
  --maximum-page-size 1
```

Salida:

```
{  
  "domainInfos": [  
    {  
      "status": "REGISTERED",  
      "name": "DataFrobotz"  
    }  
  ],  
  "nextPageToken": "AAAAKgAAAAEAAAAAAAAAA2QJKNtidVgd49TTeNwYcpD  
+QKT2ynuEbibcQWe2QKrs1MGe63gpS0MgZGpcpoKttL40CXRFn98Xif557it  
+wSZUsvUDtImjDLvguyuyyFdIZtvIxIKE0Pm3k2r40jAGaFsG0uVbrK1jv1a7wduU7FYH301kNCP8b7PBj9SBkUyGoiAg
```

```
}

```

Para obtener más información, consulte [ListDomains](#) la API referencia de Amazon Simple Workflow Service

Ejemplo 4: Para enumerar la página única especificada de dominios registrados

El siguiente ejemplo de `list-domains` comando muestra los REGISTERED SWF dominios de la primera página que ha registrado para su cuenta mediante la `--maximum-page-size` opción.

Cuando vuelvas a realizar la llamada, esta vez introduciendo el valor de `nextPageToken` en el `--next-page-token` argumento, obtendrás otra página de resultados.

```
aws swf list-domains \
  --registration-status REGISTERED \
  --maximum-page-size 1 \
  --next-page-token "AAAAKgAAAAEAAAAAAAAA2QJKNtidVgd49TTeNwYcpD
+QKT2ynuEbIBCQWe2QKrsLMGe63gpS0MgZGpcpoKttL40CXRFn98Xif557it
+wSZUsvUDtImjDLvguyuyyFdzIZtvIxIKE0Pm3k2r40jAGaFsG0uVbrKljv1a7wdU7FYH301kNCP8b7PBj9SBkUyGoiAg
```

Salida:

```
{
  "domainInfos": [
    {
      "status": "REGISTERED",
      "name": "erontest"
    }
  ]
}
```

Cuando no haya más páginas de resultados que recuperar, `nextPageToken` no aparecerá en los resultados.

Para obtener más información, consulte [ListDomains](#) la API referencia de Amazon Simple Workflow Service

- Para API obtener más información, consulte [ListDomains](#) la Referencia de AWS CLI comandos.

## list-workflow-types

En el siguiente ejemplo de código, se muestra cómo usar `list-workflow-types`.



## AWS CLI

### Listado de tipos de flujo de trabajo

Para obtener una lista de los tipos de flujo de trabajo de un dominio, utilice `swf list-workflow-types`. Los `--registration-status` argumentos `--domain` y son obligatorios. He aquí un ejemplo sencillo.

```
aws swf list-workflow-types \  
  --domain DataFrobtzz \  
  --registration-status REGISTERED
```

Salida:

```
{  
  "typeInfos": [  
    {  
      "status": "REGISTERED",  
      "creationDate": 1371454149.598,  
      "description": "DataFrobtzz subscribe workflow",  
      "workflowType": {  
        "version": "v3",  
        "name": "subscribe"  
      }  
    }  
  ]  
}
```

Del mismo `list-activity-types` modo, puedes usar el `--name` argumento para seleccionar solo los tipos de flujo de trabajo con un nombre concreto y usar el `--maximum-page-size` argumento en coordinación con `--next-page-token` los resultados de la página. Para invertir el orden en el que se devuelven los resultados, utilice `--reverse-order`.

Consulte también [ListWorkflowTypes](#) en la API referencia de Amazon Simple Workflow Service

- Para API obtener más información, consulte [ListWorkflowTypes](#) la Referencia de AWS CLI comandos.

## register-domain

En el siguiente ejemplo de código, se muestra cómo usar `register-domain`.

## AWS CLI

### Registrar un dominio

Puede utilizar el AWS CLI para registrar nuevos dominios. Utilice el comando `swf register-domain`. Hay dos parámetros obligatorios: el que toma el nombre de dominio y `--workflow-execution-retention-period-in-days` el que utiliza un número entero para especificar el número de días que se deben conservar los datos de ejecución del flujo de trabajo en este dominio, hasta un período máximo de 90 días (para obtener más información, consulte SWF FAQ < [https://aws.amazon.com/swf/faqs/#retain\\_limit](https://aws.amazon.com/swf/faqs/#retain_limit) >). `--name` Los datos de ejecución del flujo de trabajo no se conservarán una vez transcurrido el número de días especificado.

```
aws swf register-domain \  
  --name MyNeatNewDomain \  
  --workflow-execution-retention-period-in-days 0  
  ""
```

Al registrar un dominio, no se devuelve nada (""), pero puede utilizar `swf list-domains` o `swf describe-domain` para ver el nuevo dominio.

```
aws swf list-domains \  
  --registration-status REGISTERED  
  {  
    "domainInfos": [  
      {  
        "status": "REGISTERED",  
        "name": "DataFrobotz"  
      },  
      {  
        "status": "REGISTERED",  
        "name": "MyNeatNewDomain"  
      },  
      {  
        "status": "REGISTERED",  
        "name": "erontest"  
      }  
    ]  
  }
```

Uso de `swf describe-domain`:

```
aws swf describe-domain --
name MyNeatNewDomain
{
  "domainInfo": {
    "status": "REGISTERED",
    "name": "MyNeatNewDomain"
  },
  "configuration": {
    "workflowExecutionRetentionPeriodInDays": "0"
  }
}
```

Consulte también [RegisterDomain](#) en la API referencia de Amazon Simple Workflow Service

- Para API obtener más información, consulte [RegisterDomain](#) la Referencia de AWS CLI comandos.

## register-workflow-type

En el siguiente ejemplo de código, se muestra cómo usar `register-workflow-type`.

### AWS CLI

Registro de un tipo de flujo de trabajo

Para registrar un tipo de flujo de trabajo con el AWS CLI, utilice el `swf register-workflow-type` comando.

```
aws swf register-workflow-type \
--domain DataFrobtzz \
--name "MySimpleWorkflow" \
--workflow-version "v1"
```

Si se ejecuta correctamente, el comando no produce ningún resultado.

En caso de error (por ejemplo, si intenta registrar el mismo tipo de flujo de trabajo dos veces o especifica un dominio que no existe), recibirá una respuesta. JSON

```
{
  "message": "WorkflowType=[name=MySimpleWorkflow, version=v1]",
}
```

```
"__type": "com.amazonaws.swf.base.model#TypeAlreadyExistsFault"  
}
```

Las letras `--domain`, `--name` y `--workflow-version` son obligatorias. También puede establecer la descripción del flujo de trabajo, los tiempos de espera y la política del flujo de trabajo secundario.

Para obtener más información, consulte [RegisterWorkflowType](#) la API referencia de Amazon Simple Workflow Service

- Para API obtener más información, consulte [RegisterWorkflowType](#) la Referencia de AWS CLI comandos.

## Ejemplos de Systems Manager que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Systems Manager.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **add-tags-to-resource**

En el siguiente ejemplo de código, se muestra cómo usar `add-tags-to-resource`.

AWS CLI

Ejemplo 1: adición de etiquetas a un periodo de mantenimiento

En el siguiente ejemplo de `add-tags-to-resource` se agrega una etiqueta al periodo de mantenimiento especificado.

```
aws ssm add-tags-to-resource \  
  --resource-type "MaintenanceWindow" \  
  --resource-id "mw-03eb9db428EXAMPLE" \  
  --tags "Key=Stack,Value=Production"
```

Este comando no genera ninguna salida.

Ejemplo 2: adición de etiquetas a un parámetro

En el siguiente ejemplo de `add-tags-to-resource` se agregan dos etiquetas al parámetro especificado.

```
aws ssm add-tags-to-resource \  
  --resource-type "Parameter" \  
  --resource-id "My-Parameter" \  
  --tags '[{"Key": "Region", "Value": "East"}, {"Key": "Environment",  
  "Value": "Production"}]'
```

Este comando no genera ninguna salida.

Ejemplo 3: Para añadir etiquetas a un SSM documento

En el siguiente ejemplo de `add-tags-to-resource` se agrega una etiqueta al documento especificado.

```
aws ssm add-tags-to-resource \  
  --resource-type "Document" \  
  --resource-id "My-Documents" \  
  --tags "Key=Quarter,Value=Q322"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de recursos de Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [AddTagsToResource](#) la Referencia de AWS CLI comandos.

## **associate-ops-item-related-item**

En el siguiente ejemplo de código, se muestra cómo usar `associate-ops-item-related-item`.

## AWS CLI

Para asociar un elemento relacionado

El siguiente `associate-ops-item-related-item` ejemplo asocia un elemento relacionado al OpsItem.

```
aws ssm associate-ops-item-related-item \  
  --ops-item-id "oi-649fExample" \  
  --association-type "RelatesTo" \  
  --resource-type "AWS::SSMIncidents::IncidentRecord" \  
  --resource-uri "arn:aws:ssm-incidents::111122223333:incident-record/Example-  
Response-Plan/c2bde883-f7d5-343a-b13a-bf5fe9ea689f"
```

Salida:

```
{  
  "AssociationId": "61d7178d-a30d-4bc5-9b4e-a9e74EXAMPLE"  
}
```

Para obtener más información, consulte [Trabajar con incidentes de Incident Manager OpsCenter en](#) la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [AssociateOpsItemRelatedItem](#) la Referencia de AWS CLI comandos.

## cancel-command

En el siguiente ejemplo de código, se muestra cómo usar `cancel-command`.

### AWS CLI

Ejemplo 1: cancelación de un comando para todas las instancias

En el siguiente ejemplo de `cancel-command` se intenta cancelar el comando especificado que ya se está ejecutando en todas las instancias.

```
aws ssm cancel-command \  
  --command-id "662add3d-5831-4a10-b64a-f2ff3EXAMPLE"
```

Este comando no genera ninguna salida.

## Ejemplo 2: cancelación de un comando para instancias específicas

En el siguiente ejemplo de `cancel-command` se intenta cancelar un comando únicamente en la instancia especificada.

```
aws ssm cancel-command \  
  --command-id "662add3d-5831-4a10-b64a-f2ff3EXAMPLE" \  
  --instance-ids "i-02573cafcfEXAMPLE"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de parámetros de Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [CancelCommand](#) la Referencia de AWS CLI comandos.

## cancel-maintenance-window-execution

En el siguiente ejemplo de código, se muestra cómo usar `cancel-maintenance-window-execution`.

### AWS CLI

Para cancelar la ejecución de una ventana de mantenimiento

Este `cancel-maintenance-window-execution` ejemplo detiene la ejecución de la ventana de mantenimiento especificada que ya está en curso.

```
aws ssm cancel-maintenance-window-execution \  
  --window-execution-id j218d5b5c-mw66-tk4d-r3g9-1d4d1EXAMPLE
```

Salida:

```
{  
  "WindowExecutionId": "j218d5b5c-mw66-tk4d-r3g9-1d4d1EXAMPLE"  
}
```

Para obtener más información, consulte [los tutoriales de Windows sobre mantenimiento de Systems Manager \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [CancelMaintenanceWindowExecution](#) la Referencia de AWS CLI comandos.

## create-activation

En el siguiente ejemplo de código, se muestra cómo usar `create-activation`.

### AWS CLI

Creación de la activación de una instancia administrada

En el siguiente ejemplo de `create-activation` se crea la activación de una instancia administrada.

```
aws ssm create-activation \  
  --default-instance-name "HybridWebServers" \  
  --iam-role "HybridWebServersRole" \  
  --registration-limit 5
```

Salida:

```
{  
  "ActivationId": "5743558d-563b-4457-8682-d16c3EXAMPLE",  
  "ActivationCode": "dRmgnYaFv567vEXAMPLE"  
}
```

Para obtener más información, consulte [Paso 4: crear una activación híbrida para un entorno híbrido y multinube](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [CreateActivation](#) la Referencia de AWS CLI comandos.

## create-association-batch

En el siguiente ejemplo de código, se muestra cómo usar `create-association-batch`.

### AWS CLI

Creación de varias asociaciones

En este ejemplo se asocia un documento de configuración a varias instancias. El resultado devuelve una lista de operaciones correctas y con errores, si corresponde.



## Comando:

```
aws ssm create-association-batch --entries "Name=AWS-UpdateSSMAgent,InstanceId=i-1234567890abcdef0" "Name=AWS-UpdateSSMAgent,InstanceId=i-9876543210abcdef0"
```

## Salida:

```
{
  "Successful": [
    {
      "Name": "AWS-UpdateSSMAgent",
      "InstanceId": "i-1234567890abcdef0",
      "AssociationVersion": "1",
      "Date": 1550504725.007,
      "LastUpdateAssociationDate": 1550504725.007,
      "Status": {
        "Date": 1550504725.007,
        "Name": "Associated",
        "Message": "Associated with AWS-UpdateSSMAgent"
      },
      "Overview": {
        "Status": "Pending",
        "DetailedStatus": "Creating"
      },
      "DocumentVersion": "$DEFAULT",
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "Targets": [
        {
          "Key": "InstanceIds",
          "Values": [
            "i-1234567890abcdef0"
          ]
        }
      ]
    },
    {
      "Name": "AWS-UpdateSSMAgent",
      "InstanceId": "i-9876543210abcdef0",
      "AssociationVersion": "1",
      "Date": 1550504725.057,
      "LastUpdateAssociationDate": 1550504725.057,
      "Status": {
```

```

        "Date": 1550504725.057,
        "Name": "Associated",
        "Message": "Associated with AWS-UpdateSSMAgent"
    },
    "Overview": {
        "Status": "Pending",
        "DetailedStatus": "Creating"
    },
    "DocumentVersion": "$DEFAULT",
    "AssociationId": "9c9f7f20-5154-4fed-a83e-0123456789ab",
    "Targets": [
        {
            "Key": "InstanceIds",
            "Values": [
                "i-9876543210abcdef0"
            ]
        }
    ]
}
],
"Failed": []
}

```

- Para API obtener más información, consulte [CreateAssociationBatch](#) la Referencia de AWS CLI comandos.

## create-association

En el siguiente ejemplo de código, se muestra cómo usar `create-association`.

### AWS CLI

Ejemplo 1: Para asociar un documento mediante una instancia IDs

En este ejemplo, se asocia un documento de configuración a una instancia, utilizando `instanceIds`.

```

aws ssm create-association \
  --instance-id "i-0cb2b964d3e14fd9f" \
  --name "AWS-UpdateSSMAgent"

```

Salida:

```
{
  "AssociationDescription": {
    "Status": {
      "Date": 1487875500.33,
      "Message": "Associated with AWS-UpdateSSMAgent",
      "Name": "Associated"
    },
    "Name": "AWS-UpdateSSMAgent",
    "InstanceId": "i-0cb2b964d3e14fd9f",
    "Overview": {
      "Status": "Pending",
      "DetailedStatus": "Creating"
    },
    "AssociationId": "b7c3266e-a544-44db-877e-b20d3a108189",
    "DocumentVersion": "$DEFAULT",
    "LastUpdateAssociationDate": 1487875500.33,
    "Date": 1487875500.33,
    "Targets": [
      {
        "Values": [
          "i-0cb2b964d3e14fd9f"
        ],
        "Key": "InstanceIds"
      }
    ]
  }
}
```

Para obtener más información, consulte [CreateAssociation](#) la API Referencia AWS de Systems Manager.

Ejemplo 2: asociación de un documento mediante destinos

En este ejemplo, se asocia un documento de configuración a una instancia mediante destinos.

```
aws ssm create-association \
  --name "AWS-UpdateSSMAgent" \
  --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f"
```

Salida:

```
{
```

```

"AssociationDescription": {
  "Status": {
    "Date": 1487875500.33,
    "Message": "Associated with AWS-UpdateSSMAgent",
    "Name": "Associated"
  },
  "Name": "AWS-UpdateSSMAgent",
  "InstanceId": "i-0cb2b964d3e14fd9f",
  "Overview": {
    "Status": "Pending",
    "DetailedStatus": "Creating"
  },
  "AssociationId": "b7c3266e-a544-44db-877e-b20d3a108189",
  "DocumentVersion": "$DEFAULT",
  "LastUpdateAssociationDate": 1487875500.33,
  "Date": 1487875500.33,
  "Targets": [
    {
      "Values": [
        "i-0cb2b964d3e14fd9f"
      ],
      "Key": "InstanceIds"
    }
  ]
}
}

```

Para obtener más información, consulte [CreateAssociation](#) la API Referencia AWS de Systems Manager.

Ejemplo 3: creación de una asociación que solo se ejecuta una vez

En este ejemplo se crea una nueva asociación que solo se ejecuta una vez en la fecha y hora especificadas. Las asociaciones creadas con una fecha en el pasado o en el presente (cuando se procesan, la fecha queda en el pasado) se ejecutan inmediatamente.

```

aws ssm create-association \
  --name "AWS-UpdateSSMAgent" \
  --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" \
  --schedule-expression "at(2020-05-14T15:55:00)" \
  --apply-only-at-cron-interval

```

Salida:

```
{
  "AssociationDescription": {
    "Status": {
      "Date": 1487875500.33,
      "Message": "Associated with AWS-UpdateSSMAgent",
      "Name": "Associated"
    },
    "Name": "AWS-UpdateSSMAgent",
    "InstanceId": "i-0cb2b964d3e14fd9f",
    "Overview": {
      "Status": "Pending",
      "DetailedStatus": "Creating"
    },
    "AssociationId": "b7c3266e-a544-44db-877e-b20d3a108189",
    "DocumentVersion": "$DEFAULT",
    "LastUpdateAssociationDate": 1487875500.33,
    "Date": 1487875500.33,
    "Targets": [
      {
        "Values": [
          "i-0cb2b964d3e14fd9f"
        ],
        "Key": "InstanceIds"
      }
    ]
  }
}
```

Para obtener más información, consulte [CreateAssociation](#) la Referencia de AWS Systems Manager o API Referencia: [expresiones de cron y velocidad para Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [CreateAssociation](#) la Referencia de AWS CLI comandos.

## create-document

En el siguiente ejemplo de código, se muestra cómo usar create-document.

### AWS CLI

#### Creación de un documento

En el siguiente ejemplo de `create-document` se crea un documento de Systems Manager.

```
aws ssm create-document \  
  --content file://exampleDocument.yml \  
  --name "Example" \  
  --document-type "Automation" \  
  --document-format YAML
```

Salida:

```
{  
  "DocumentDescription": {  
    "Hash": "fc2410281f40779e694a8b95975d0f9f316da8a153daa94e3d9921102EXAMPLE",  
    "HashType": "Sha256",  
    "Name": "Example",  
    "Owner": "29884EXAMPLE",  
    "CreateDate": 1583256349.452,  
    "Status": "Creating",  
    "DocumentVersion": "1",  
    "Description": "Document Example",  
    "Parameters": [  
      {  
        "Name": "AutomationAssumeRole",  
        "Type": "String",  
        "Description": "(Required) The ARN of the role that allows  
Automation to perform the actions on your behalf. If no role is specified, Systems  
Manager Automation uses your IAM permissions to execute this document.",  
        "DefaultValue": ""  
      },  
      {  
        "Name": "InstanceId",  
        "Type": "String",  
        "Description": "(Required) The ID of the Amazon EC2 instance.",  
        "DefaultValue": ""  
      }  
    ],  
    "PlatformTypes": [  
      "Windows",  
      "Linux"  
    ],  
    "DocumentType": "Automation",  
    "SchemaVersion": "0.3",  
    "LatestVersion": "1",
```

```
    "DefaultVersion": "1",
    "DocumentFormat": "YAML",
    "Tags": []
  }
}
```

Para obtener más información, consulte [Crear contenido en el documento de SSM](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [CreateDocument](#) la Referencia de AWS CLI comandos.

## create-maintenance-window

En el siguiente ejemplo de código, se muestra cómo usar `create-maintenance-window`.

### AWS CLI

Ejemplo 1: creación de un periodo de mantenimiento

En el siguiente ejemplo de `create-maintenance-window` se crea un nuevo periodo de mantenimiento que, cada cinco minutos durante un máximo de dos horas (según sea necesario), impide que se inicien nuevas tareas una hora después de que finalice el periodo de mantenimiento, permite los destinos no asociados (instancias que no ha registrado con el periodo de mantenimiento) e indica, mediante el uso de etiquetas personalizadas, que su creador tiene la intención de utilizarlo en un tutorial.

```
aws ssm create-maintenance-window \
  --name "My-Tutorial-Maintenance-Window" \
  --schedule "rate(5 minutes)" \
  --duration 2 --cutoff 1 \
  --allow-unassociated-targets \
  --tags "Key=Purpose, Value=Tutorial"
```

Salida:

```
{
  "WindowId": "mw-0c50858d01EXAMPLE"
}
```

Ejemplo 2: creación de un periodo de mantenimiento que se ejecuta solo una vez

En el siguiente ejemplo de `create-maintenance-window` se crea un nuevo periodo de mantenimiento que solo se ejecuta una vez en la fecha y hora especificadas.

```
aws ssm create-maintenance-window \
  --name My-One-Time-Maintenance-Window \
  --schedule "at(2020-05-14T15:55:00)" \
  --duration 5 \
  --cutoff 2 \
  --allow-unassociated-targets \
  --tags "Key=Environment,Value=Production"
```

Salida:

```
{
  "WindowId": "mw-01234567890abcdef"
}
```

Para obtener más información, consulte [Maintenance Windows](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [CreateMaintenanceWindow](#) la Referencia de AWS CLI comandos.

## create-ops-item

En el siguiente ejemplo de código, se muestra cómo usar `create-ops-item`.

### AWS CLI

Para crear un OpsItems

En el siguiente `create-ops-item` ejemplo, se utiliza la clave `/aws/resources OperationalData` para crear un recurso relacionado OpsItem con Amazon DynamoDB.

```
aws ssm create-ops-item \
  --title "EC2 instance disk full" \
  --description "Log clean up may have failed which caused the disk to be full" \
  --priority 2 \
  --source ec2 \
  --operational-data '{"aws/resources":{"Value":"[{"arn": "arn:aws:dynamodb:us-west-2:12345678:table/OpsItems"}]","Type":"SearchableString"}' \
```



```
--notifications Arn="arn:aws:sns:us-west-2:12345678:TestUser"
```

Salida:

```
{
  "OpsItemId": "oi-1a2b3c4d5e6f"
}
```

Para obtener más información, consulte [Creación OpsItems](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [CreateOpsItem](#) la Referencia de AWS CLI comandos.

## create-patch-baseline

En el siguiente ejemplo de código, se muestra cómo usar create-patch-baseline.

AWS CLI

Ejemplo 1: creación de una línea de base de revisiones con aprobación automática

En el siguiente ejemplo de create-patch-baseline se crea una línea de base de revisiones para Windows Server que aprueba las revisiones de un entorno de producción siete días después de que Microsoft las publique.

```
aws ssm create-patch-baseline \
  --name "Windows-Production-Baseline-AutoApproval" \
  --operating-system "WINDOWS" \
  --approval-
rules "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,Impo
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,UpdateRollups,CriticalUpdates]}]},Approv
\
  --description "Baseline containing all updates approved for Windows Server
production systems"
```

Salida:

```
{
  "BaselineId": "pb-045f10b4f3EXAMPLE"
}
```

## Ejemplo 2: creación de una línea de base de revisiones con una fecha límite de aprobación

En el siguiente ejemplo de `create-patch-baseline` se crea una línea de base de revisiones para Windows Server que aprueba todas las revisiones de un entorno de producción que se publicaron el 7 de julio del 2020 o antes.

```
aws ssm create-patch-baseline \
  --name "Windows-Production-Baseline-AutoApproval" \
  --operating-system "WINDOWS" \
  --approval-
rules "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,Impo
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,UpdateRollups,CriticalUpdates]}]},Approv
\
  --description "Baseline containing all updates approved for Windows Server
production systems"
```

Salida:

```
{
  "BaselineId": "pb-045f10b4f3EXAMPLE"
}
```

Ejemplo 3: Para crear una línea base de parches con las reglas de aprobación almacenadas en un JSON archivo

En el siguiente ejemplo de `create-patch-baseline` se crea una línea de base de revisiones para Amazon Linux 2017.09 que aprueba las revisiones de un entorno de producción siete días después de su publicación, especifica las reglas de aprobación para la línea de base de revisiones y especifica un repositorio personalizado para las revisiones.

```
aws ssm create-patch-baseline \
  --cli-input-json file://my-amazon-linux-approval-rules-and-repo.json
```

Contenidos de `my-amazon-linux-approval-rules-and-repo.json`:

```
{
  "Name": "Amazon-Linux-2017.09-Production-Baseline",
  "Description": "My approval rules patch baseline for Amazon Linux 2017.09
instances",
  "OperatingSystem": "AMAZON_LINUX",
  "Tags": [
```

```
    {
      "Key": "Environment",
      "Value": "Production"
    }
  ],
  "ApprovalRules": {
    "PatchRules": [
      {
        "ApproveAfterDays": 7,
        "EnableNonSecurity": true,
        "PatchFilterGroup": {
          "PatchFilters": [
            {
              "Key": "SEVERITY",
              "Values": [
                "Important",
                "Critical"
              ]
            },
            {
              "Key": "CLASSIFICATION",
              "Values": [
                "Security",
                "Bugfix"
              ]
            },
            {
              "Key": "PRODUCT",
              "Values": [
                "AmazonLinux2017.09"
              ]
            }
          ]
        }
      }
    ]
  },
  "Sources": [
    {
      "Name": "My-AL2017.09",
      "Products": [
        "AmazonLinux2017.09"
      ],
    }
  ],
}
```

```

    "Configuration": "[amzn-main] \nname=amzn-main-Base
\nmirrorlist=http://repo./$awsregion./$awsdomain//$releasever/main/mirror.list //
\nmirrorlist_expire=300//\nmetadata_expire=300 \npriority=10 \nfailovermethod=priority
\nfastestmirror_enabled=0 \ngpgcheck=1 \ngpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-
KEY-amazon-ga \nenabled=1 \nretries=3 \ntimeout=5\nreport_instanceid=yes"
    }
  ]
}

```

Ejemplo 4: creación de una línea de base de revisiones que especifica las revisiones aprobadas y rechazadas

En el siguiente ejemplo de `create-patch-baseline` se especifican de forma explícita las revisiones que se deben aprobar y rechazar como excepción a las reglas de aprobación predeterminadas.

```

aws ssm create-patch-baseline \
  --name "Amazon-Linux-2017.09-Alpha-Baseline" \
  --description "My custom approve/reject patch baseline for Amazon Linux 2017.09
instances" \
  --operating-system "AMAZON_LINUX" \
  --approved-patches "CVE-2018-1234567,example-pkg-EE-2018*.amzn1.noarch" \
  --approved-patches-compliance-level "HIGH" \
  --approved-patches-enable-non-security \
  --tags "Key=Environment,Value=Alpha"

```

Para obtener más información, consulte [Creación de una línea de base de revisiones personalizada](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [CreatePatchBaseline](#) la Referencia de AWS CLI comandos.

## create-resource-data-sync

En el siguiente ejemplo de código, se muestra cómo usar `create-resource-data-sync`.

### AWS CLI

Para crear una sincronización de datos de recursos

En este ejemplo, se crea una sincronización de datos de recursos. No se obtienen resultados si el comando se ejecuta correctamente.

Comando:

```
aws ssm create-resource-data-sync --sync-name "ssm-resource-data-sync" --s3-destination "BucketName=ssm-bucket,Prefix=inventory,SyncFormat=JsonSerDe,Region=us-east-1"
```

- Para API obtener más información, consulte [CreateResourceDataSync](#) la Referencia de AWS CLI comandos.

## delete-activation

En el siguiente ejemplo de código, se muestra cómo usar `delete-activation`.

AWS CLI

Eliminación de la activación de una instancia administrada

En el siguiente ejemplo de `delete-activation` se elimina la activación de una instancia administrada.

```
aws ssm delete-activation \
  --activation-id "aa673477-d926-42c1-8757-1358cEXAMPLE"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de AWS Systems Manager para entornos híbridos](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DeleteActivation](#) la Referencia de AWS CLI comandos.

## delete-association

En el siguiente ejemplo de código, se muestra cómo usar `delete-association`.

AWS CLI

Ejemplo 1: eliminación de una asociación mediante el ID de asociación

En el siguiente ejemplo de `delete-association` se elimina la asociación del ID de asociación especificado. No se obtienen resultados si el comando se ejecuta correctamente.

```
aws ssm delete-association \  
  --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Edición y creación de una nueva versión de una asociación](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: eliminación de una asociación

En el siguiente ejemplo de `delete-association` se elimina la asociación entre una instancia y un documento. No se obtienen resultados si el comando se ejecuta correctamente.

```
aws ssm delete-association \  
  --instance-id "i-1234567890abcdef0" \  
  --name "AWS-UpdateSSMAgent"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajo con asociaciones en Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DeleteAssociation](#) la Referencia de AWS CLI comandos.

## delete-document

En el siguiente ejemplo de código, se muestra cómo usar `delete-document`.

AWS CLI

Eliminación de un documento

En el siguiente ejemplo de `delete-document` se elimina un documento de Systems Manager.

```
aws ssm delete-document \  
  --name "Example"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Crear contenido en el documento de SSM](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DeleteDocument](#) la Referencia de AWS CLI comandos.

## delete-inventory

En el siguiente ejemplo de código, se muestra cómo usar `delete-inventory`.

### AWS CLI

Para eliminar un tipo de inventario personalizado

En este ejemplo, se elimina un esquema de inventario personalizado.

Comando:

```
aws ssm delete-inventory --type-name "Custom:RackInfo" --schema-delete-option "DeLeteSchema"
```

Salida:

```
{
  "DeletionId": "d72ac9e8-1f60-4d40-b1c6-bf8c78c68c4d",
  "TypeName": "Custom:RackInfo",
  "DeletionSummary": {
    "TotalCount": 1,
    "RemainingCount": 1,
    "SummaryItems": [
      {
        "Version": "1.0",
        "Count": 1,
        "RemainingCount": 1
      }
    ]
  }
}
```

Para deshabilitar un tipo de inventario personalizado

En este ejemplo, se inhabilita un esquema de inventario personalizado.

Comando:

```
aws ssm delete-inventory --type-name "Custom:RackInfo" --schema-delete-  
option "DisableSchema"
```

Salida:

```
{  
  "DeletionId": "6961492a-8163-44ec-aa1e-923364dd0850",  
  "TypeName": "Custom:RackInformation",  
  "DeletionSummary": {  
    "TotalCount": 0,  
    "RemainingCount": 0,  
    "SummaryItems": []  
  }  
}
```

- Para API obtener más información, consulte [DeleteInventory](#) la Referencia de AWS CLI comandos.

## delete-maintenance-window

En el siguiente ejemplo de código, se muestra cómo usar `delete-maintenance-window`.

AWS CLI

Eliminación de un periodo de mantenimiento

En este ejemplo de `delete-maintenance-window` se elimina el periodo de mantenimiento especificado.

```
aws ssm delete-maintenance-window \  
  --window-id "mw-1a2b3c4d5e6f7g8h9"
```

Salida:

```
{  
  "WindowId": "mw-1a2b3c4d5e6f7g8h9"  
}
```

Para obtener más información, consulte [Eliminar una ventana de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.



- Para API obtener más información, consulte [DeleteMaintenanceWindow](#) la Referencia de AWS CLI comandos.

## delete-parameter

En el siguiente ejemplo de código, se muestra cómo usar `delete-parameter`.

### AWS CLI

Eliminación de un parámetro

En el siguiente ejemplo de `delete-parameter` se elimina el parámetro único especificado.

```
aws ssm delete-parameter \  
  --name "MyParameter"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Uso de Parameter Store](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DeleteParameter](#) la Referencia de AWS CLI comandos.

## delete-parameters

En el siguiente ejemplo de código, se muestra cómo usar `delete-parameters`.

### AWS CLI

Para eliminar una lista de parámetros

En el siguiente `delete-parameters` ejemplo, se eliminan los parámetros especificados.

```
aws ssm delete-parameters \  
  --names "MyFirstParameter" "MySecondParameter" "MyInvalidParameterName"
```

Salida:

```
{  
  "DeletedParameters": [  
    "MyFirstParameter",
```

```
    "MySecondParameter"  
  ],  
  "InvalidParameters": [  
    "MyInvalidParameterName"  
  ]  
}
```

Para obtener más información, consulte [Uso de Parameter Store](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DeleteParameters](#) la Referencia de AWS CLI comandos.

## delete-patch-baseline

En el siguiente ejemplo de código, se muestra cómo usar delete-patch-baseline.

### AWS CLI

Eliminación de una línea de base de revisiones

En el siguiente ejemplo de delete-patch-baseline se elimina la línea de base de revisiones especificada.

```
aws ssm delete-patch-baseline \  
  --baseline-id "pb-045f10b4f382baeda"
```

Salida:

```
{  
  "BaselineId": "pb-045f10b4f382baeda"  
}
```

Para obtener más información, consulte [Actualización o eliminación de una línea de base de revisiones \(consola\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DeletePatchBaseline](#) la Referencia de AWS CLI comandos.

## delete-resource-data-sync

En el siguiente ejemplo de código, se muestra cómo usar delete-resource-data-sync.

## AWS CLI

Para eliminar la sincronización de datos de un recurso

En este ejemplo, se elimina la sincronización de datos de un recurso. No se obtienen resultados si el comando se ejecuta correctamente.

Comando:

```
aws ssm delete-resource-data-sync --sync-name "ssm-resource-data-sync"
```

- Para API obtener más información, consulte [DeleteResourceDataSync](#) la Referencia de AWS CLI comandos.

## deregister-managed-instance

En el siguiente ejemplo de código, se muestra cómo usar `deregister-managed-instance`.

### AWS CLI

Anulación del registro de una instancia administrada

En el siguiente ejemplo de `deregister-managed-instance` se anula el registro de la instancia administrada especificada.

```
aws ssm deregister-managed-instance  
--instance-id "mi-08ab247cdfEXAMPLE"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Anulación del registro de nodos administrados en un entorno híbrido y multinube](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DeregisterManagedInstance](#) la Referencia de AWS CLI comandos.

## deregister-patch-baseline-for-patch-group

En el siguiente ejemplo de código, se muestra cómo usar `deregister-patch-baseline-for-patch-group`.

## AWS CLI

Anulación del registro de un grupo de revisiones de una línea de base de revisiones

En el siguiente ejemplo de `deregister-patch-baseline-for-patch-group` se anula el registro del grupo de revisiones especificado de la línea de base de revisiones.

```
aws ssm deregister-patch-baseline-for-patch-group \  
  --patch-group "Production" \  
  --baseline-id "pb-0ca44a362fEXAMPLE"
```

Salida:

```
{  
  "PatchGroup": "Production",  
  "BaselineId": "pb-0ca44a362fEXAMPLE"  
}
```

Para obtener más información, consulte [Añadir un grupo de revisiones a una línea de base de revisiones](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DeregisterPatchBaselineForPatchGroup](#) la Referencia de AWS CLI comandos.

## `deregister-target-from-maintenance-window`

En el siguiente ejemplo de código, se muestra cómo usar `deregister-target-from-maintenance-window`.

## AWS CLI

Eliminación de un destino de un periodo de mantenimiento

En el siguiente ejemplo de `deregister-target-from-maintenance-window`, se elimina el destino especificado del periodo de mantenimiento especificado.

```
aws ssm deregister-target-from-maintenance-window \  
  --window-id "mw-ab12cd34ef56gh78" \  
  --window-target-id "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
```

Salida:

```
{
  "WindowId": "mw-ab12cd34ef56gh78",
  "WindowTargetId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Para obtener más información, consulte [Actualizar una ventana de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DeregisterTargetFromMaintenanceWindow](#) la Referencia de AWS CLI comandos.

## deregister-task-from-maintenance-window

En el siguiente ejemplo de código, se muestra cómo usar `deregister-task-from-maintenance-window`.

AWS CLI

Eliminación de una tarea de un periodo de mantenimiento

En el siguiente ejemplo de `deregister-task-from-maintenance-window` se elimina la tarea especificada del periodo de mantenimiento especificado.

```
aws ssm deregister-task-from-maintenance-window \
  --window-id "mw-ab12cd34ef56gh78" \
  --window-task-id "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d5e6c"
```

Salida:

```
{
  "WindowTaskId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d5e6c",
  "WindowId": "mw-ab12cd34ef56gh78"
}
```

Para obtener más información, consulte [los tutoriales de Windows sobre mantenimiento de Systems Manager \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DeregisterTaskFromMaintenanceWindow](#) la Referencia de AWS CLI comandos.

## describe-activations

En el siguiente ejemplo de código, se muestra cómo usar `describe-activations`.

### AWS CLI

Descripción de las activaciones

En el siguiente `describe-activations` ejemplo, se muestran detalles sobre las activaciones de su AWS cuenta.

```
aws ssm describe-activations
```

Salida:

```
{
  "ActivationList": [
    {
      "ActivationId": "5743558d-563b-4457-8682-d16c3EXAMPLE",
      "Description": "Example1",
      "IamRole": "HybridWebServersRole",
      "RegistrationLimit": 5,
      "RegistrationsCount": 5,
      "ExpirationDate": 1584316800.0,
      "Expired": false,
      "CreateDate": 1581954699.792
    },
    {
      "ActivationId": "3ee0322b-f62d-40eb-b672-13ebfEXAMPLE",
      "Description": "Example2",
      "IamRole": "HybridDatabaseServersRole",
      "RegistrationLimit": 5,
      "RegistrationsCount": 5,
      "ExpirationDate": 1580515200.0,
      "Expired": true,
      "CreateDate": 1578064132.002
    }
  ]
}
```

Para obtener más información, consulte [Paso 4: crear una activación híbrida para un entorno híbrido y multinube](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeActivations](#) la Referencia de AWS CLI comandos.

## describe-association-execution-targets

En el siguiente ejemplo de código, se muestra cómo usar `describe-association-execution-targets`.

### AWS CLI

Obtención de detalles de la ejecución de una asociación

En el siguiente ejemplo de `describe-association-execution-targets` se describe la ejecución de la asociación especificada.

```
aws ssm describe-association-execution-targets \  
  --association-id "8dfe3659-4309-493a-8755-0123456789ab" \  
  --execution-id "7abb6378-a4a5-4f10-8312-0123456789ab"
```

Salida:

```
{  
  "AssociationExecutionTargets": [  
    {  
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",  
      "AssociationVersion": "1",  
      "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",  
      "ResourceId": "i-1234567890abcdef0",  
      "ResourceType": "ManagedInstance",  
      "Status": "Success",  
      "DetailedStatus": "Success",  
      "LastExecutionDate": 1550505538.497,  
      "OutputSource": {  
        "OutputSourceId": "97fff367-fc5a-4299-aed8-0123456789ab",  
        "OutputSourceType": "RunCommand"  
      }  
    }  
  ]  
}
```

Para obtener más información, consulte [Visualización de los historiales de asociación](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeAssociationExecutionTargets](#) Referencia de AWS CLI comandos.

## describe-association-executions

En el siguiente ejemplo de código, se muestra cómo usar `describe-association-executions`.

### AWS CLI

Ejemplo 1: obtención de detalles de todas las ejecuciones de una asociación

En el siguiente ejemplo de `describe-association-executions` se describen todas las ejecuciones de la asociación especificada.

```
aws ssm describe-association-executions \  
  --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Salida:

```
{  
  "AssociationExecutions": [  
    {  
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",  
      "AssociationVersion": "1",  
      "ExecutionId": "474925ef-1249-45a2-b93d-0123456789ab",  
      "Status": "Success",  
      "DetailedStatus": "Success",  
      "CreatedTime": 1550505827.119,  
      "ResourceCountByStatus": "{Success=1}"  
    },  
    {  
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",  
      "AssociationVersion": "1",  
      "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",  
      "Status": "Success",  
      "DetailedStatus": "Success",  
      "CreatedTime": 1550505536.843,  
      "ResourceCountByStatus": "{Success=1}"  
    },  
    ...  
  ]  
}
```



Para obtener más información, consulte [Visualización de los historiales de asociación](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: obtención de detalles de todas las ejecuciones de una asociación después de una fecha y hora específicas

En el siguiente ejemplo de `describe-association-executions` se describen todas las ejecuciones de una asociación después de la fecha y hora especificadas.

```
aws ssm describe-association-executions \
  --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
  --filters "Key=CreatedTime,Value=2019-02-18T16:00:00Z,Type=GREATER_THAN"
```

Salida:

```
{
  "AssociationExecutions": [
    {
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "AssociationVersion": "1",
      "ExecutionId": "474925ef-1249-45a2-b93d-0123456789ab",
      "Status": "Success",
      "DetailedStatus": "Success",
      "CreatedTime": 1550505827.119,
      "ResourceCountByStatus": "{Success=1}"
    },
    {
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "AssociationVersion": "1",
      "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
      "Status": "Success",
      "DetailedStatus": "Success",
      "CreatedTime": 1550505536.843,
      "ResourceCountByStatus": "{Success=1}"
    },
    ...
  ]
}
```

Para obtener más información, consulte [Visualización de los historiales de asociación](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeAssociationExecutions](#) la Referencia de AWS CLI comandos.

## describe-association

En el siguiente ejemplo de código, se muestra cómo usar describe-association.

### AWS CLI

Ejemplo 1: obtención de los detalles de una asociación

En el siguiente ejemplo de describe-association se describe la asociación del ID de asociación especificado.

```
aws ssm describe-association \
  --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Salida:

```
{
  "AssociationDescription": {
    "Name": "AWS-GatherSoftwareInventory",
    "AssociationVersion": "1",
    "Date": 1534864780.995,
    "LastUpdateAssociationDate": 1543235759.81,
    "Overview": {
      "Status": "Success",
      "AssociationStatusAggregatedCount": {
        "Success": 2
      }
    }
  },
  "DocumentVersion": "$DEFAULT",
  "Parameters": {
    "applications": [
      "Enabled"
    ],
    "awsComponents": [
      "Enabled"
    ],
    "customInventory": [
      "Enabled"
    ]
  }
}
```

```
    "files": [
      ""
    ],
    "instanceDetailedInformation": [
      "Enabled"
    ],
    "networkConfig": [
      "Enabled"
    ],
    "services": [
      "Enabled"
    ],
    "windowsRegistry": [
      ""
    ],
    "windowsRoles": [
      "Enabled"
    ],
    "windowsUpdates": [
      "Enabled"
    ]
  },
  "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
  "Targets": [
    {
      "Key": "InstanceIds",
      "Values": [
        "*"
      ]
    }
  ],
  "ScheduleExpression": "rate(24 hours)",
  "LastExecutionDate": 1550501886.0,
  "LastSuccessfulExecutionDate": 1550501886.0,
  "AssociationName": "Inventory-Association"
}
}
```

Para obtener más información, consulte [Edición y creación de una nueva versión de una asociación](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: obtención de los detalles de una asociación para una instancia y un documento específicos

En el siguiente ejemplo de `describe-association` se describe la asociación entre una instancia y un documento.

```
aws ssm describe-association \  
  --instance-id "i-1234567890abcdef0" \  
  --name "AWS-UpdateSSMAgent"
```

Salida:

```
{  
  "AssociationDescription": {  
    "Status": {  
      "Date": 1487876122.564,  
      "Message": "Associated with AWS-UpdateSSMAgent",  
      "Name": "Associated"  
    },  
    "Name": "AWS-UpdateSSMAgent",  
    "InstanceId": "i-1234567890abcdef0",  
    "Overview": {  
      "Status": "Pending",  
      "DetailedStatus": "Associated",  
      "AssociationStatusAggregatedCount": {  
        "Pending": 1  
      }  
    },  
    "AssociationId": "d8617c07-2079-4c18-9847-1234567890ab",  
    "DocumentVersion": "$DEFAULT",  
    "LastUpdateAssociationDate": 1487876122.564,  
    "Date": 1487876122.564,  
    "Targets": [  
      {  
        "Values": [  
          "i-1234567890abcdef0"  
        ],  
        "Key": "InstanceIds"  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [Edición y creación de una nueva versión de una asociación](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeAssociation](#) la Referencia de AWS CLI comandos.

## describe-automation-executions

En el siguiente ejemplo de código, se muestra cómo usar `describe-automation-executions`.

### AWS CLI

Descripción de una ejecución de automatización

En el siguiente ejemplo de `describe-automation-executions` se muestran detalles sobre una ejecución de Automatización.

```
aws ssm describe-automation-executions \  
  --filters Key=ExecutionId,Values=73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Salida:

```
{  
  "AutomationExecutionMetadataList": [  
    {  
      "AutomationExecutionId": "73c8eef8-f4ee-4a05-820c-e354fEXAMPLE",  
      "DocumentName": "AWS-StartEC2Instance",  
      "DocumentVersion": "1",  
      "AutomationExecutionStatus": "Success",  
      "ExecutionStartTime": 1583737233.748,  
      "ExecutionEndTime": 1583737234.719,  
      "ExecutedBy": "arn:aws:sts::29884EXAMPLE:assumed-role/mw_service_role/  
OrchestrationService",  
      "LogFile": "",  
      "Outputs": {},  
      "Mode": "Auto",  
      "Targets": [],  
      "ResolvedTargets": {  
        "ParameterValues": [],  
        "Truncated": false  
      },  
      "AutomationType": "Local"  
    }  
  ]  
}
```

Para obtener más información, consulte [Ejecución de una automatización sencilla](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeAutomationExecutions](#) la Referencia de AWS CLI comandos.

## describe-automation-step-executions

En el siguiente ejemplo de código, se muestra cómo usar `describe-automation-step-executions`.

### AWS CLI

Ejemplo 1: descripción de todos los pasos de una ejecución de automatización

En el siguiente ejemplo de `describe-automation-step-executions` se muestran detalles sobre los pasos de una ejecución de Automatización.

```
aws ssm describe-automation-step-executions \
  --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Salida:

```
{
  "StepExecutions": [
    {
      "StepName": "startInstances",
      "Action": "aws:changeInstanceState",
      "ExecutionStartTime": 1583737234.134,
      "ExecutionEndTime": 1583737234.672,
      "StepStatus": "Success",
      "Inputs": {
        "DesiredState": "\"running\"",
        "InstanceIds": "[\"i-0cb99161f6EXAMPLE\"]"
      },
      "Outputs": {
        "InstanceStates": [
          "running"
        ]
      },
      "StepExecutionId": "95e70479-cf20-4d80-8018-7e4e2EXAMPLE",
      "OverriddenParameters": {}
    }
  ]
}
```

```
]
}
```

Ejemplo 2: descripción de un paso concreto de una ejecución de automatización

En el siguiente ejemplo de `describe-automation-step-executions` se muestran detalles sobre un paso específico de una ejecución de Automatización.

```
aws ssm describe-automation-step-executions \
  --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE \
  --filters Key=StepExecutionId,Values=95e70479-cf20-4d80-8018-7e4e2EXAMPLE
```

Para obtener más información, consulte [Running an Automation Workflow Step by Step \(Command Line\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeAutomationStepExecutions](#) la Referencia de AWS CLI comandos.

## `describe-available-patches`

En el siguiente ejemplo de código, se muestra cómo usar `describe-available-patches`.

### AWS CLI

Obtención de las revisiones disponibles

En el siguiente `describe-available-patches` ejemplo, se obtienen detalles sobre todos los parches disponibles para Windows Server 2019 que tienen una MSRC gravedad crítica.

```
aws ssm describe-available-patches \
  --
  filters "Key=PRODUCT,Values=WindowsServer2019" "Key=MSRC_SEVERITY,Values=Critical"
```

Salida:

```
{
  "Patches": [
    {
      "Id": "fe6bd8c2-3752-4c8b-ab3e-1a7ed08767ba",
      "ReleaseDate": 1544047205.0,
      "Title": "2018-11 Update for Windows Server 2019 for x64-based Systems (KB4470788)",
```

```

        "Description": "Install this update to resolve issues in Windows. For a
complete listing of the issues that are included in this update, see the associated
Microsoft Knowledge Base article for more information. After you install this item,
you may have to restart your computer.",
        "ContentUrl": "https://support.microsoft.com/en-us/kb/4470788",
        "Vendor": "Microsoft",
        "ProductFamily": "Windows",
        "Product": "WindowsServer2019",
        "Classification": "SecurityUpdates",
        "MsrcSeverity": "Critical",
        "KbNumber": "KB4470788",
        "MsrcNumber": "",
        "Language": "All"
    },
    {
        "Id": "c96115e1-5587-4115-b851-22baa46a3f11",
        "ReleaseDate": 1549994410.0,
        "Title": "2019-02 Security Update for Adobe Flash Player for Windows
Server 2019 for x64-based Systems (KB4487038)",
        "Description": "A security issue has been identified in a Microsoft
software product that could affect your system. You can help protect your system
by installing this update from Microsoft. For a complete listing of the issues that
are included in this update, see the associated Microsoft Knowledge Base article.
After you install this update, you may have to restart your system.",
        "ContentUrl": "https://support.microsoft.com/en-us/kb/4487038",
        "Vendor": "Microsoft",
        "ProductFamily": "Windows",
        "Product": "WindowsServer2019",
        "Classification": "SecurityUpdates",
        "MsrcSeverity": "Critical",
        "KbNumber": "KB4487038",
        "MsrcNumber": "",
        "Language": "All"
    },
    ...
]
}

```

## Obtención de detalles de una revisión específica

En el siguiente ejemplo de `describe-available-patches` se recuperan los detalles sobre la revisión especificada.



```
aws ssm describe-available-patches \  
  --filters "Key=PATCH_ID,Values=KB4480979"
```

Salida:

```
{  
  "Patches": [  
    {  
      "Id": "680861e3-fb75-432e-818e-d72e5f2be719",  
      "ReleaseDate": 1546970408.0,  
      "Title": "2019-01 Security Update for Adobe Flash Player for Windows  
Server 2016 for x64-based Systems (KB4480979)",  
      "Description": "A security issue has been identified in a Microsoft  
software product that could affect your system. You can help protect your system  
by installing this update from Microsoft. For a complete listing of the issues that  
are included in this update, see the associated Microsoft Knowledge Base article.  
After you install this update, you may have to restart your system.",  
      "ContentUrl": "https://support.microsoft.com/en-us/kb/4480979",  
      "Vendor": "Microsoft",  
      "ProductFamily": "Windows",  
      "Product": "WindowsServer2016",  
      "Classification": "SecurityUpdates",  
      "MsrcSeverity": "Critical",  
      "KbNumber": "KB4480979",  
      "MsrcNumber": "",  
      "Language": "All"  
    }  
  ]  
}
```

Para obtener más información, consulte [Cómo funcionan las operaciones de Patch Manager](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeAvailablePatches](#) la Referencia de AWS CLI comandos.

## describe-document-permission

En el siguiente ejemplo de código, se muestra cómo usar `describe-document-permission`.

## AWS CLI

### Descripción de los permisos de los documentos

En el siguiente ejemplo de `describe-document-permission` se muestran los detalles de los permisos sobre un documento de Systems Manager que se comparte públicamente.

```
aws ssm describe-document-permission \  
  --name "Example" \  
  --permission-type "Share"
```

Salida:

```
{  
  "AccountIds": [  
    "all"  
  ],  
  "AccountSharingInfoList": [  
    {  
      "AccountId": "all",  
      "SharedDocumentVersion": "$DEFAULT"  
    }  
  ]  
}
```

Para obtener más información, consulte [Compartir un documento de Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeDocumentPermission](#) la Referencia de AWS CLI comandos.

## **describe-document**

En el siguiente ejemplo de código, se muestra cómo usar `describe-document`.

## AWS CLI

### Visualización de los detalles de un documento

En el siguiente `describe-document` ejemplo, se muestran detalles sobre un documento de Systems Manager de su AWS cuenta.

```
aws ssm describe-document \  
  --name "Example"
```

Salida:

```
{  
  "Document": {  
    "Hash": "fc2410281f40779e694a8b95975d0f9f316da8a153daa94e3d9921102EXAMPLE",  
    "HashType": "Sha256",  
    "Name": "Example",  
    "Owner": "29884EXAMPLE",  
    "CreateDate": 1583257938.266,  
    "Status": "Active",  
    "DocumentVersion": "1",  
    "Description": "Document Example",  
    "Parameters": [  
      {  
        "Name": "AutomationAssumeRole",  
        "Type": "String",  
        "Description": "(Required) The ARN of the role that allows  
Automation to perform the actions on your behalf. If no role is specified, Systems  
Manager Automation uses your IAM permissions to execute this document.",  
        "DefaultValue": ""  
      },  
      {  
        "Name": "InstanceId",  
        "Type": "String",  
        "Description": "(Required) The ID of the Amazon EC2 instance.",  
        "DefaultValue": ""  
      }  
    ],  
    "PlatformTypes": [  
      "Windows",  
      "Linux"  
    ],  
    "DocumentType": "Automation",  
    "SchemaVersion": "0.3",  
    "LatestVersion": "1",  
    "DefaultVersion": "1",  
    "DocumentFormat": "YAML",  
    "Tags": []  
  }  
}
```

```
}
```

Para obtener más información, consulte [Crear contenido en el documento de SSM](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeDocument](#) la Referencia de AWS CLI comandos.

## describe-effective-instance-associations

En el siguiente ejemplo de código, se muestra cómo usar `describe-effective-instance-associations`.

### AWS CLI

Obtención de los detalles de las asociaciones efectivas de una instancia

En el siguiente ejemplo de `describe-effective-instance-associations` se recuperan los detalles sobre las asociaciones efectivas de una instancia.

Comando:

```
aws ssm describe-effective-instance-associations --instance-id "i-1234567890abcdef0"
```

Salida:

```
{
  "Associations": [
    {
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "InstanceId": "i-1234567890abcdef0",
      "Content": "{\n  \"schemaVersion\": \"1.2\",\n  \"description\":\n  \"Update the Amazon SSM Agent to the latest version or specified version.\",\n  \"parameters\": {\n    \"version\": {\n      \"default\": \"\",\n      \"description\": \"(Optional) A specific version of the Amazon SSM Agent\n  to install. If not specified, the agent will be updated to the latest version.\",\n      \"type\": \"String\",\n    },\n    \"allowDowngrade\": {\n      \"default\": \"false\",\n      \"description\": \"(Optional)\n  Allow the Amazon SSM Agent service to be downgraded to an earlier version. If\n  set to false, the service can be upgraded to newer versions only (default). If\n  set to true, specify the earlier version.\",\n      \"type\": \"String\",\n      \"allowedValues\": [\n        \"true\",\n
```

```

    \"false\"
  ],
  \"runtimeConfig\": {
    \"aws:updateSsmAgent\": {
      \"properties\": [
        {
          \"agentName\": \"amazon-ssm-agent\",
          \"source\":
            \"https://s3.{Region}.amazonaws.com/amazon-ssm-{Region}/ssm-agent-manifest.json\",
          \"allowDowngrade\": \"{{ allowDowngrade }}\",
          \"targetVersion\": \"{{ version }}\"
        }
      ]
    }
  }
}

```

- Para API obtener más información, consulte [DescribeEffectiveInstanceAssociations](#) la Referencia de AWS CLI comandos.

## describe-effective-patches-for-patch-baseline

En el siguiente ejemplo de código, se muestra cómo usar `describe-effective-patches-for-patch-baseline`.

### AWS CLI

Ejemplo 1: obtención de todas las revisiones definidas por una línea de base de revisiones personalizada

El siguiente `describe-effective-patches-for-patch-baseline` ejemplo devuelve los parches definidos por una línea base de parches personalizada en la AWS cuenta corriente. Tenga en cuenta que, para una línea de base personalizada, solo se requiere el ID para `--baseline-id`.

```

aws ssm describe-effective-patches-for-patch-baseline \
  --baseline-id "pb-08b654cf9b9681f04"

```

Salida:

```

{
  "EffectivePatches": [
    {
      "Patch": {
        "Id": "fe6bd8c2-3752-4c8b-ab3e-1a7ed08767ba",
        "ReleaseDate": 1544047205.0,

```

```
        "Title": "2018-11 Update for Windows Server 2019 for x64-based
Systems (KB4470788)",
        "Description": "Install this update to resolve issues in Windows.
For a complete listing of the issues that are included in this update, see the
associated Microsoft Knowledge Base article for more information. After you install
this item, you may have to restart your computer.",
        "ContentUrl": "https://support.microsoft.com/en-us/kb/4470788",
        "Vendor": "Microsoft",
        "ProductFamily": "Windows",
        "Product": "WindowsServer2019",
        "Classification": "SecurityUpdates",
        "MsrcSeverity": "Critical",
        "KbNumber": "KB4470788",
        "MsrcNumber": "",
        "Language": "All"
    },
    "PatchStatus": {
        "DeploymentStatus": "APPROVED",
        "ComplianceLevel": "CRITICAL",
        "ApprovalDate": 1544047205.0
    }
},
{
    "Patch": {
        "Id": "915a6b1a-f556-4d83-8f50-b2e75a9a7e58",
        "ReleaseDate": 1549994400.0,
        "Title": "2019-02 Cumulative Update for .NET Framework 3.5 and 4.7.2
for Windows Server 2019 for x64 (KB4483452)",
        "Description": "A security issue has been identified in a Microsoft
software product that could affect your system. You can help protect your system by
installing this update from Microsoft. For a complete listing of the issues that
are included in this update, see the associated Microsoft Knowledge Base article.
After you install this update, you may have to restart your system.",
        "ContentUrl": "https://support.microsoft.com/en-us/kb/4483452",
        "Vendor": "Microsoft",
        "ProductFamily": "Windows",
        "Product": "WindowsServer2019",
        "Classification": "SecurityUpdates",
        "MsrcSeverity": "Important",
        "KbNumber": "KB4483452",
        "MsrcNumber": "",
        "Language": "All"
    },
    "PatchStatus": {
```

```

        "DeploymentStatus": "APPROVED",
        "ComplianceLevel": "CRITICAL",
        "ApprovalDate": 1549994400.0
    },
    ...
],
"NextToken": "--token string truncated--"
}

```

Ejemplo 2: Para definir todos los parches mediante una línea base de parches AWS gestionada

El siguiente `describe-effective-patches-for-patch-baseline` ejemplo devuelve los parches definidos por una línea base de parches AWS gestionada. Tenga en cuenta que, en el caso de una línea base AWS gestionada, ARN es necesaria la línea base completa para `--baseline-id`

```

aws ssm describe-effective-patches-for-patch-baseline \
  --baseline-id "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-020d361a05defe4ed"

```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte [Cómo se seleccionan las revisiones de seguridad](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeEffectivePatchesForPatchBaseline](#) la Referencia de AWS CLI comandos.

## **describe-instance-associations-status**

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-associations-status`.

### AWS CLI

Descripción del estado de las asociaciones de una instancia

En este ejemplo se muestran los detalles de las asociaciones de una instancia.

Comando:

```
aws ssm describe-instance-associations-status --instance-id "i-1234567890abcdef0"
```

Salida:

```
{
  "InstanceAssociationStatusInfos": [
    {
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "Name": "AWS-GatherSoftwareInventory",
      "DocumentVersion": "1",
      "AssociationVersion": "1",
      "InstanceId": "i-1234567890abcdef0",
      "ExecutionDate": 1550501886.0,
      "Status": "Success",
      "ExecutionSummary": "1 out of 1 plugin processed, 1 success, 0 failed, 0
timedout, 0 skipped. ",
      "AssociationName": "Inventory-Association"
    },
    {
      "AssociationId": "5c5a31f6-6dae-46f9-944c-0123456789ab",
      "Name": "AWS-UpdateSSMAgent",
      "DocumentVersion": "1",
      "AssociationVersion": "1",
      "InstanceId": "i-1234567890abcdef0",
      "ExecutionDate": 1550505828.548,
      "Status": "Success",
      "DetailedStatus": "Success",
      "AssociationName": "UpdateSSMAgent"
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeInstanceAssociationsStatus](#) la Referencia de AWS CLI comandos.

## describe-instance-information

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-information`.

AWS CLI

Ejemplo 1: descripción de la información de las instancias administradas



En el siguiente ejemplo de `describe-instance-information` se recuperan los detalles de cada una de las instancias administradas.

```
aws ssm describe-instance-information
```

Ejemplo 2: descripción de la información sobre una instancia administrada específica

En el siguiente ejemplo de `describe-instance-information` se muestran los detalles de la instancia administrada `i-028ea792daEXAMPLE`.

```
aws ssm describe-instance-information \  
  --filters "Key=InstanceIds,Values=i-028ea792daEXAMPLE"
```

Ejemplo 3: descripción de la información sobre las instancias administradas con una clave de etiqueta específica

En el siguiente ejemplo de `describe-instance-information` se muestran los detalles de las instancias administradas que tienen la clave de etiqueta `DEV`.

```
aws ssm describe-instance-information \  
  --filters "Key=tag-key,Values=DEV"
```

Salida:

```
{  
  "InstanceInformationList": [  
    {  
      "InstanceId": "i-028ea792daEXAMPLE",  
      "PingStatus": "Online",  
      "LastPingDateTime": 1582221233.421,  
      "AgentVersion": "2.3.842.0",  
      "IsLatestVersion": true,  
      "PlatformType": "Linux",  
      "PlatformName": "SLES",  
      "PlatformVersion": "15.1",  
      "ResourceType": "EC2Instance",  
      "IPAddress": "192.0.2.0",  
      "ComputerName": "ip-198.51.100.0.us-east-2.compute.internal",  
      "AssociationStatus": "Success",  
      "LastAssociationExecutionDate": 1582220806.0,  
      "LastSuccessfulAssociationExecutionDate": 1582220806.0,  
    }  
  ]  
}
```

```

    "AssociationOverview": {
      "DetailedStatus": "Success",
      "InstanceAssociationStatusAggregatedCount": {
        "Success": 2
      }
    }
  ]
}

```

Para obtener más información, consulte [Managed Instances](#) en la Guía del usuario de AWS .

- Para API obtener más información, consulte [DescribeInstanceInformation](#) la Referencia de AWS CLI comandos.

## describe-instance-patch-states-for-patch-group

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-patch-states-for-patch-group`.

### AWS CLI

Ejemplo 1: obtención de los estados de las instancias de un grupo de revisiones

En el siguiente ejemplo de `describe-instance-patch-states-for-patch-group` se obtienen detalles sobre los estados resumidos de las revisiones por instancia en el grupo de revisiones especificado.

```
aws ssm describe-instance-patch-states-for-patch-group \
  --patch-group "Production"
```

Salida:

```

{
  "InstancePatchStates": [
    {
      "InstanceId": "i-02573cafcfEXAMPLE",
      "PatchGroup": "Production",
      "BaselineId": "pb-0c10e65780EXAMPLE",
      "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",
      "OwnerInformation": "",

```

```

    "InstalledCount": 32,
    "InstalledOtherCount": 1,
    "InstalledPendingRebootCount": 0,
    "InstalledRejectedCount": 0,
    "MissingCount": 2,
    "FailedCount": 0,
    "UnreportedNotApplicableCount": 2671,
    "NotApplicableCount": 400,
    "OperationStartTime": "2021-08-04T11:03:50.590000-07:00",
    "OperationEndTime": "2021-08-04T11:04:21.555000-07:00",
    "Operation": "Scan",
    "RebootOption": "NoReboot",
    "CriticalNonCompliantCount": 0,
    "SecurityNonCompliantCount": 1,
    "OtherNonCompliantCount": 0
  },
  {
    "InstanceId": "i-0471e04240EXAMPLE",
    "PatchGroup": "Production",
    "BaselineId": "pb-09ca3fb51fEXAMPLE",
    "SnapshotId": "05d8ffb0-1bbe-4812-ba2d-d9b7bEXAMPLE",
    "OwnerInformation": "",
    "InstalledCount": 32,
    "InstalledOtherCount": 1,
    "InstalledPendingRebootCount": 0,
    "InstalledRejectedCount": 0,
    "MissingCount": 2,
    "FailedCount": 0,
    "UnreportedNotApplicableCount": 2671,
    "NotApplicableCount": 400,
    "OperationStartTime": "2021-08-04T22:06:20.340000-07:00",
    "OperationEndTime": "2021-08-04T22:07:11.220000-07:00",
    "Operation": "Scan",
    "RebootOption": "NoReboot",
    "CriticalNonCompliantCount": 0,
    "SecurityNonCompliantCount": 1,
    "OtherNonCompliantCount": 0
  }
]
}

```

Ejemplo 2: obtención de los estados de las instancias de un grupo de revisiones al que le faltan más de cinco revisiones

En el siguiente ejemplo de `describe-instance-patch-states-for-patch-group` se obtienen detalles sobre los estados resumidos de las revisiones del grupo de revisiones especificado en las instancias a las que les faltan más de cinco revisiones.

```
aws ssm describe-instance-patch-states-for-patch-group \  
  --filters Key=MissingCount,Type=GreaterThan,Values=5 \  
  --patch-group "Production"
```

Salida:

```
{  
  "InstancePatchStates": [  
    {  
      "InstanceId": "i-02573cafcfEXAMPLE",  
      "PatchGroup": "Production",  
      "BaselineId": "pb-0c10e65780EXAMPLE",  
      "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",  
      "OwnerInformation": "",  
      "InstalledCount": 46,  
      "InstalledOtherCount": 4,  
      "InstalledPendingRebootCount": 1,  
      "InstalledRejectedCount": 1,  
      "MissingCount": 7,  
      "FailedCount": 0,  
      "UnreportedNotApplicableCount": 232,  
      "NotApplicableCount": 654,  
      "OperationStartTime": "2021-08-04T11:03:50.590000-07:00",  
      "OperationEndTime": "2021-08-04T11:04:21.555000-07:00",  
      "Operation": "Scan",  
      "RebootOption": "NoReboot",  
      "CriticalNonCompliantCount": 0,  
      "SecurityNonCompliantCount": 1,  
      "OtherNonCompliantCount": 1  
    }  
  ]  
}
```

Ejemplo 3: obtención de los estados de las instancias de un grupo de revisiones con menos de diez instancias que requieren un reinicio

En el siguiente ejemplo de `describe-instance-patch-states-for-patch-group` se obtienen detalles sobre los estados resumidos de las revisiones del grupo de revisiones especificado en las instancias con menos de diez instancias que requieren un reinicio.

```
aws ssm describe-instance-patch-states-for-patch-group \
  --filters Key=InstalledPendingRebootCount,Type=LessThan,Values=10 \
  --patch-group "Production"
```

Salida:

```
{
  "InstancePatchStates": [
    {
      "InstanceId": "i-02573cafcfEXAMPLE",
      "BaselineId": "pb-0c10e65780EXAMPLE",
      "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",
      "PatchGroup": "Production",
      "OwnerInformation": "",
      "InstalledCount": 32,
      "InstalledOtherCount": 1,
      "InstalledPendingRebootCount": 4,
      "InstalledRejectedCount": 0,
      "MissingCount": 2,
      "FailedCount": 0,
      "UnreportedNotApplicableCount": 846,
      "NotApplicableCount": 212,
      "OperationStartTime": "2021-08-04T11:03:50.590000-07:00",
      "OperationEndTime": "2021-08-06T11:04:21.555000-07:00",
      "Operation": "Scan",
      "RebootOption": "NoReboot",
      "CriticalNonCompliantCount": 0,
      "SecurityNonCompliantCount": 1,
      "OtherNonCompliantCount": 0
    }
  ]
}
```

Para obtener más información, consulte [Conocimiento de los valores del estado de conformidad de parches](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeInstancePatchStatesForPatchGroup](#) la Referencia de AWS CLI comandos.

## describe-instance-patch-states

En el siguiente ejemplo de código, se muestra cómo usar `describe-instance-patch-states`.

### AWS CLI

Obtención de los estados resumidos de las revisiones en instancias

En este ejemplo de `describe-instance-patch-states` se obtienen los estados resumidos de las revisiones en una instancia.

```
aws ssm describe-instance-patch-states \  
  --instance-ids "i-1234567890abcdef0"
```

Salida:

```
{  
  "InstancePatchStates": [  
    {  
      "InstanceId": "i-1234567890abcdef0",  
      "PatchGroup": "my-patch-group",  
      "BaselineId": "pb-0713accee01234567",  
      "SnapshotId": "521c3536-930c-4aa9-950e-01234567abcd",  
      "CriticalNonCompliantCount": 2,  
      "SecurityNonCompliantCount": 2,  
      "OtherNonCompliantCount": 1,  
      "InstalledCount": 123,  
      "InstalledOtherCount": 334,  
      "InstalledPendingRebootCount": 0,  
      "InstalledRejectedCount": 0,  
      "MissingCount": 1,  
      "FailedCount": 2,  
      "UnreportedNotApplicableCount": 11,  
      "NotApplicableCount": 2063,  
      "OperationStartTime": "2021-05-03T11:00:56-07:00",  
      "OperationEndTime": "2021-05-03T11:01:09-07:00",  
      "Operation": "Scan",  
      "LastNoRebootInstallOperationTime": "2020-06-14T12:17:41-07:00",  
      "RebootOption": "RebootIfNeeded"  
    }  
  ]  
}
```

Para obtener más información, consulte [Conocimiento de los valores del estado de conformidad de parches](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeInstancePatchStates](#) la Referencia de AWS CLI comandos.

## describe-instance-patches

En el siguiente ejemplo de código, se muestra cómo usar describe-instance-patches.

### AWS CLI

Ejemplo 1: obtención de los detalles del estado de una revisión en una instancia

En el siguiente ejemplo de describe-instance-patches se recuperan los detalles sobre la instancia especificada.

```
aws ssm describe-instance-patches \  
  --instance-id "i-1234567890abcdef0"
```

Salida:

```
{  
  "Patches": [  
    {  
      "Title": "2019-01 Security Update for Adobe Flash Player for Windows  
Server 2016 for x64-based Systems (KB4480979)",  
      "KBId": "KB4480979",  
      "Classification": "SecurityUpdates",  
      "Severity": "Critical",  
      "State": "Installed",  
      "InstalledTime": "2019-01-09T00:00:00+00:00"  
    },  
    {  
      "Title": "",  
      "KBId": "KB4481031",  
      "Classification": "",  
      "Severity": "",  
      "State": "InstalledOther",  
      "InstalledTime": "2019-02-08T00:00:00+00:00"  
    },  
    ...  
  ]  
}
```

```

    ],
    "NextToken": "--token string truncated--"
}

```

Ejemplo 2: obtención de una lista de las revisiones que se encuentran en el estado Faltante en una instancia

En el siguiente ejemplo de `describe-instance-patches` se recupera información sobre las revisiones que se encuentran en el estado Falta en la instancia especificada.

```

aws ssm describe-instance-patches \
  --instance-id "i-1234567890abcdef0" \
  --filters Key=State,Values=Missing

```

Salida:

```

{
  "Patches": [
    {
      "Title": "Windows Malicious Software Removal Tool x64 - February 2019 (KB890830)",
      "KBId": "KB890830",
      "Classification": "UpdateRollups",
      "Severity": "Unspecified",
      "State": "Missing",
      "InstalledTime": "1970-01-01T00:00:00+00:00"
    },
    ...
  ],
  "NextToken": "--token string truncated--"
}

```

Para obtener más información, consulte [Conocimiento de los valores del estado de conformidad de parches](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 3: Para obtener una lista de los parches instalados desde una instancia especificada `InstalledTime`

En el siguiente ejemplo de `describe-instance-patches` se recupera información sobre las revisiones instaladas desde un tiempo específico en la instancia especificada combinando el uso de `--filters` y `--query`.



```
aws ssm describe-instance-patches \  
  --instance-id "i-1234567890abcdef0" \  
  --filters Key=State,Values=Installed \  
  --query "Patches[?InstalledTime >= `2023-01-01T16:00:00`]"
```

Salida:

```
{  
  "Patches": [  
    {  
      "Title": "2023-03 Cumulative Update for Windows Server 2019 (1809) for  
x64-based Systems (KB5023702)",  
      "KBId": "KB5023702",  
      "Classification": "SecurityUpdates",  
      "Severity": "Critical",  
      "State": "Installed",  
      "InstalledTime": "2023-03-16T11:00:00+00:00"  
    },  
    ...  
  ],  
  "NextToken": "--token string truncated--"  
}
```

- Para API obtener más información, consulte [DescribeInstancePatches](#) la Referencia de AWS CLI comandos.

## describe-inventory-deletions

En el siguiente ejemplo de código, se muestra cómo usar `describe-inventory-deletions`.

AWS CLI

Para obtener información sobre las eliminaciones de inventario

En este ejemplo, se recuperan los detalles de las operaciones de eliminación de inventario.

Comando:

```
aws ssm describe-inventory-deletions
```

Salida:

```
{
  "InventoryDeletions": [
    {
      "DeletionId": "6961492a-8163-44ec-aa1e-01234567850",
      "TypeName": "Custom:RackInformation",
      "DeletionStartTime": 1550254911.0,
      "LastStatus": "InProgress",
      "LastStatusMessage": "The Delete is in progress",
      "DeletionSummary": {
        "TotalCount": 0,
        "RemainingCount": 0,
        "SummaryItems": []
      },
      "LastStatusUpdateTime": 1550254911.0
    },
    {
      "DeletionId": "d72ac9e8-1f60-4d40-b1c6-987654321c4d",
      "TypeName": "Custom:RackInfo",
      "DeletionStartTime": 1550254859.0,
      "LastStatus": "InProgress",
      "LastStatusMessage": "The Delete is in progress",
      "DeletionSummary": {
        "TotalCount": 1,
        "RemainingCount": 1,
        "SummaryItems": [
          {
            "Version": "1.0",
            "Count": 1,
            "RemainingCount": 1
          }
        ]
      },
      "LastStatusUpdateTime": 1550254859.0
    }
  ]
}
```

Para obtener detalles de una eliminación de inventario específica

En este ejemplo, se recuperan los detalles de una operación de eliminación de inventario específica.

Comando:

```
aws ssm describe-inventory-deletions --deletion-id "d72ac9e8-1f60-4d40-b1c6-987654321c4d"
```

Salida:

```
{
  "InventoryDeletions": [
    {
      "DeletionId": "d72ac9e8-1f60-4d40-b1c6-987654321c4d",
      "TypeName": "Custom:RackInfo",
      "DeletionStartTime": 1550254859.0,
      "LastStatus": "InProgress",
      "LastStatusMessage": "The Delete is in progress",
      "DeletionSummary": {
        "TotalCount": 1,
        "RemainingCount": 1,
        "SummaryItems": [
          {
            "Version": "1.0",
            "Count": 1,
            "RemainingCount": 1
          }
        ]
      },
      "LastStatusUpdateTime": 1550254859.0
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeInventoryDeletions](#) la Referencia de AWS CLI comandos.

## **describe-maintenance-window-execution-task-invocations**

En el siguiente ejemplo de código, se muestra cómo usar `describe-maintenance-window-execution-task-invocations`.

### AWS CLI

Obtención de las invocaciones de tareas específicas hechas para la ejecución de una tarea en un periodo de mantenimiento

En el siguiente ejemplo de `describe-maintenance-window-execution-task-invocations` se muestran las invocaciones de la tarea especificada que se ejecutó como parte de la ejecución del periodo de mantenimiento especificado.

```
aws ssm describe-maintenance-window-execution-task-invocations \
  --window-execution-id "518d5565-5969-4cca-8f0e-da3b2a638355" \
  --task-id "ac0c6ae1-daa3-4a89-832e-d384503b6586"
```

Salida:

```
{
  "WindowExecutionTaskInvocationIdentities": [
    {
      "Status": "SUCCESS",
      "Parameters": "{\"documentName\": \"AWS-RunShellScript\", \"instanceIds\": [\"i-0000293ffd8c57862\"], \"parameters\": {\"commands\": [\"df\"]}, \"maxConcurrency\": \"1\", \"maxErrors\": \"1\"}",
      "InvocationId": "e274b6e1-fe56-4e32-bd2a-8073c6381d8b",
      "StartTime": 1487692834.723,
      "EndTime": 1487692834.871,
      "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2a638355",
      "TaskExecutionId": "ac0c6ae1-daa3-4a89-832e-d384503b6586"
    }
  ]
}
```

Para obtener más información, consulte [Ver información sobre tareas y ejecuciones de tareas \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeMaintenanceWindowExecutionTaskInvocations](#) la Referencia de AWS CLI comandos.

## `describe-maintenance-window-execution-tasks`

En el siguiente ejemplo de código, se muestra cómo usar `describe-maintenance-window-execution-tasks`.

AWS CLI

Enumeración de todas las tareas asociadas a la ejecución de un periodo de mantenimiento

En el siguiente ejemplo de `ssm describe-maintenance-window-execution-tasks` se enumeran todas las tareas asociadas a la ejecución del periodo de mantenimiento especificado.

```
aws ssm describe-maintenance-window-execution-tasks \
  --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE"
```

Salida:

```
{
  "WindowExecutionTaskIdentities": [
    {
      "Status": "SUCCESS",
      "TaskArn": "AWS-RunShellScript",
      "StartTime": 1487692834.684,
      "TaskType": "RUN_COMMAND",
      "EndTime": 1487692835.005,
      "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",
      "TaskExecutionId": "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"
    }
  ]
}
```

Para obtener más información, consulte [Ver información sobre tareas y ejecuciones de tareas \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeMaintenanceWindowExecutionTasks](#) la Referencia de AWS CLI comandos.

## describe-maintenance-window-executions

En el siguiente ejemplo de código, se muestra cómo usar `describe-maintenance-window-executions`.

AWS CLI

Ejemplo 1: enumeración de todas las ejecuciones de un periodo de mantenimiento

En el siguiente ejemplo de `describe-maintenance-window-executions` se enumeran todas las ejecuciones del periodo de mantenimiento especificado.

```
aws ssm describe-maintenance-window-executions \
```

```
--window-id "mw-ab12cd34eEXAMPLE"
```

Salida:

```
{
  "WindowExecutions": [
    {
      "WindowId": "mw-ab12cd34eEXAMPLE",
      "WindowExecutionId": "6027b513-64fe-4cf0-be7d-1191aEXAMPLE",
      "Status": "IN_PROGRESS",
      "StartTime": "2021-08-04T11:00:00.000000-07:00"
    },
    {
      "WindowId": "mw-ab12cd34eEXAMPLE",
      "WindowExecutionId": "ff75b750-4834-4377-8f61-b3cadEXAMPLE",
      "Status": "SUCCESS",
      "StartTime": "2021-08-03T11:00:00.000000-07:00",
      "EndTime": "2021-08-03T11:37:21.450000-07:00"
    },
    {
      "WindowId": "mw-ab12cd34eEXAMPLE",
      "WindowExecutionId": "9fac7dd9-ff21-42a5-96ad-bbc4bEXAMPLE",
      "Status": "FAILED",
      "StatusDetails": "One or more tasks in the orchestration failed.",
      "StartTime": "2021-08-02T11:00:00.000000-07:00",
      "EndTime": "2021-08-02T11:22:36.190000-07:00"
    }
  ]
}
```

Ejemplo 2: enumeración de todas las ejecuciones de un periodo de mantenimiento antes de una fecha especificada

En el siguiente ejemplo de `describe-maintenance-window-executions` se enumeran todas las ejecuciones del periodo de mantenimiento especificado antes de la fecha especificada.

```
aws ssm describe-maintenance-window-executions \
  --window-id "mw-ab12cd34eEXAMPLE" \
  --filters "Key=ExecutedBefore,Values=2021-08-03T00:00:00Z"
```

Salida:

```
{
  "WindowExecutions": [
    {
      "WindowId": "mw-ab12cd34eEXAMPLE",
      "WindowExecutionId": "9fac7dd9-ff21-42a5-96ad-bbc4bEXAMPLE",
      "Status": "FAILED",
      "StatusDetails": "One or more tasks in the orchestration failed.",
      "StartTime": "2021-08-02T11:00:00.000000-07:00",
      "EndTime": "2021-08-02T11:22:36.190000-07:00"
    }
  ]
}
```

Ejemplo 3: enumeración de todas las ejecuciones de un periodo de mantenimiento después de una fecha especificada

En el siguiente ejemplo de `describe-maintenance-window-executions` se enumeran todas las ejecuciones del periodo de mantenimiento especificado después de la fecha especificada.

```
aws ssm describe-maintenance-window-executions \
  --window-id "mw-ab12cd34eEXAMPLE" \
  --filters "Key=ExecutedAfter,Values=2021-08-04T00:00:00Z"
```

Salida:

```
{
  "WindowExecutions": [
    {
      "WindowId": "mw-ab12cd34eEXAMPLE",
      "WindowExecutionId": "6027b513-64fe-4cf0-be7d-1191aEXAMPLE",
      "Status": "IN_PROGRESS",
      "StartTime": "2021-08-04T11:00:00.000000-07:00"
    }
  ]
}
```

Para obtener más información, consulte [Ver información sobre tareas y ejecuciones de tareas \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeMaintenanceWindowExecutions](#) la Referencia de AWS CLI comandos.

## describe-maintenance-window-schedule

En el siguiente ejemplo de código, se muestra cómo usar `describe-maintenance-window-schedule`.

### AWS CLI

Ejemplo 1: Para enumerar las próximas ejecuciones de un período de mantenimiento

En el siguiente `describe-maintenance-window-schedule` ejemplo, se enumeran todas las próximas ejecuciones del período de mantenimiento especificado.

```
aws ssm describe-maintenance-window-schedule \  
  --window-id mw-ab12cd34eEXAMPLE
```

Salida:

```
{  
  "ScheduledWindowExecutions": [  
    {  
      "WindowId": "mw-ab12cd34eEXAMPLE",  
      "Name": "My-First-Maintenance-Window",  
      "ExecutionTime": "2020-02-19T16:00Z"  
    },  
    {  
      "WindowId": "mw-ab12cd34eEXAMPLE",  
      "Name": "My-First-Maintenance-Window",  
      "ExecutionTime": "2020-02-26T16:00Z"  
    },  
    ...  
  ]  
}
```

Ejemplo 2: Para enumerar todas las próximas ejecuciones de un período de mantenimiento antes de una fecha específica

En el siguiente `describe-maintenance-window-schedule` ejemplo, se enumeran todas las próximas ejecuciones del período de mantenimiento especificado que se produzcan antes de la fecha especificada.

```
aws ssm describe-maintenance-window-schedule \  
  --window-id mw-0ecb1226dd7b2e9a6 \  
  --start-time
```



```
--filters "Key=ScheduledBefore,Values=2020-02-15T06:00:00Z"
```

Ejemplo 3: Para enumerar todas las próximas ejecuciones de un período de mantenimiento después de una fecha específica

En el siguiente `describe-maintenance-window-schedule` ejemplo, se enumeran todas las próximas ejecuciones del período de mantenimiento especificado que se produzcan después de la fecha especificada.

```
aws ssm describe-maintenance-window-schedule \  
  --window-id mw-0ecb1226dd7b2e9a6 \  
  --filters "Key=ScheduledAfter,Values=2020-02-15T06:00:00Z"
```

Para obtener más información, consulte [Ver información sobre el mantenimiento de Windows \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeMaintenanceWindowSchedule](#) la Referencia de AWS CLI comandos.

## `describe-maintenance-window-targets`

En el siguiente ejemplo de código, se muestra cómo usar `describe-maintenance-window-targets`.

### AWS CLI

Ejemplo 1: enumeración de todos los destinos de un periodo de mantenimiento

En el siguiente ejemplo de `describe-maintenance-window-targets` se enumeran todos los destinos de un periodo de mantenimiento.

```
aws ssm describe-maintenance-window-targets \  
  --window-id "mw-06cf17cbefEXAMPLE"
```

Salida:

```
{  
  "Targets": [  
    {  
      "ResourceType": "INSTANCE",  
      "OwnerInformation": "Single instance",
```

```

    "WindowId": "mw-06cf17cbefEXAMPLE",
    "Targets": [
      {
        "Values": [
          "i-0000293ffdEXAMPLE"
        ],
        "Key": "InstanceIds"
      }
    ],
    "WindowTargetId": "350d44e6-28cc-44e2-951f-4b2c9EXAMPLE"
  },
  {
    "ResourceType": "INSTANCE",
    "OwnerInformation": "Two instances in a list",
    "WindowId": "mw-06cf17cbefEXAMPLE",
    "Targets": [
      {
        "Values": [
          "i-0000293ffdEXAMPLE",
          "i-0cb2b964d3EXAMPLE"
        ],
        "Key": "InstanceIds"
      }
    ],
    "WindowTargetId": "e078a987-2866-47be-bedd-d9cf4EXAMPLE"
  }
]
}

```

Ejemplo 2: enumeración de todos los destinos de un periodo de mantenimiento que coincida con un valor específico de información del propietario

En este ejemplo de `describe-maintenance-window-targets` se enumeran todos los destinos de un periodo de mantenimiento con un valor específico.

```

aws ssm describe-maintenance-window-targets \
  --window-id "mw-0ecb1226ddEXAMPLE" \
  --filters "Key=OwnerInformation,Values=CostCenter1"

```

Salida:

```

{
  "Targets": [

```

```

    {
      "WindowId": "mw-0ecb1226ddEXAMPLE",
      "WindowTargetId": "da89dcc3-7f9c-481d-ba2b-edcb7d0057f9",
      "ResourceType": "INSTANCE",
      "Targets": [
        {
          "Key": "tag:Environment",
          "Values": [
            "Prod"
          ]
        }
      ],
      "OwnerInformation": "CostCenter1",
      "Name": "ProdTarget1"
    }
  ]
}

```

Para obtener más información, consulte [Ver información sobre el mantenimiento de Windows \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeMaintenanceWindowTargets](#) la Referencia de AWS CLI comandos.

## describe-maintenance-window-tasks

En el siguiente ejemplo de código, se muestra cómo usar `describe-maintenance-window-tasks`.

### AWS CLI

Ejemplo 1: enumeración de todas las tareas de un periodo de mantenimiento

En el siguiente ejemplo de `describe-maintenance-window-tasks` se enumeran todas las tareas del periodo de mantenimiento especificado.

```
aws ssm describe-maintenance-window-tasks \
  --window-id "mw-06cf17cbefEXAMPLE"
```

Salida:

```
{
```

```
"Tasks": [
  {
    "WindowId": "mw-06cf17cbefEXAMPLE",
    "WindowTaskId": "018b31c3-2d77-4b9e-bd48-c91edEXAMPLE",
    "TaskArn": "AWS-RestartEC2Instance",
    "TaskParameters": {},
    "Type": "AUTOMATION",
    "Description": "Restarting EC2 Instance for maintenance",
    "MaxConcurrency": "1",
    "MaxErrors": "1",
    "Name": "My-Automation-Example-Task",
    "Priority": 0,
    "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "Targets": [
      {
        "Key": "WindowTargetIds",
        "Values": [
          "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
        ]
      }
    ]
  },
  {
    "WindowId": "mw-06cf17cbefEXAMPLE",
    "WindowTaskId": "1943dee0-0a17-4978-9bf4-3cc2fEXAMPLE",
    "TaskArn": "AWS-DisableS3BucketPublicReadWrite",
    "TaskParameters": {},
    "Type": "AUTOMATION",
    "Description": "Automation task to disable read/write access on public
S3 buckets",
    "MaxConcurrency": "10",
    "MaxErrors": "5",
    "Name": "My-Disable-S3-Public-Read-Write-Access-Automation-Task",
    "Priority": 0,
    "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "Targets": [
      {
        "Key": "WindowTargetIds",
        "Values": [
          "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
        ]
      }
    ]
  }
]
```

```

    ]
  }
]
}

```

Ejemplo 2: Para enumerar todas las tareas de una ventana de mantenimiento que invoca el documento de RunPowerShellScript comando AWS-

En el siguiente ejemplo de `describe-maintenance-window-tasks` se enumeran todas las tareas del periodo de mantenimiento especificado que invoca el documento de comandos AWS-RunPowerShellScript.

```

aws ssm describe-maintenance-window-tasks \
  --window-id "mw-ab12cd34eEXAMPLE" \
  --filters "Key=TaskArn,Values=AWS-RunPowerShellScript"

```

Salida:

```

{
  "Tasks": [
    {
      "WindowId": "mw-ab12cd34eEXAMPLE",
      "WindowTaskId": "0d36e6b4-3a4f-411e-adcb-3558eEXAMPLE",
      "TaskArn": "AWS-RunPowerShellScript",
      "Type": "RUN_COMMAND",
      "Targets": [
        {
          "Key": "WindowTargetIds",
          "Values": [
            "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
          ]
        }
      ],
      "TaskParameters": {},
      "Priority": 1,
      "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
      "MaxConcurrency": "1",
      "MaxErrors": "1",
      "Name": "MyTask"
    }
  ]
}

```

```
}
```

Ejemplo 3: enumeración de todas las tareas de un periodo de mantenimiento que tienen una prioridad de 3

En el siguiente ejemplo de `describe-maintenance-window-tasks` se enumeran todas las tareas del periodo de mantenimiento especificado que tienen una `Priority` de 3.

```
aws ssm describe-maintenance-window-tasks \  
  --window-id "mw-ab12cd34eEXAMPLE" \  
  --filters "Key=Priority,Values=3"
```

Salida:

```
{  
  "Tasks": [  
    {  
      "WindowId": "mw-ab12cd34eEXAMPLE",  
      "WindowTaskId": "0d36e6b4-3a4f-411e-adcb-3558eEXAMPLE",  
      "TaskArn": "AWS-RunPowerShellScript",  
      "Type": "RUN_COMMAND",  
      "Targets": [  
        {  
          "Key": "WindowTargetIds",  
          "Values": [  
            "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"  
          ]  
        }  
      ],  
      "TaskParameters": {},  
      "Priority": 3,  
      "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/  
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",  
      "MaxConcurrency": "1",  
      "MaxErrors": "1",  
      "Name": "MyRunCommandTask"  
    },  
    {  
      "WindowId": "mw-ab12cd34eEXAMPLE",  
      "WindowTaskId": "ee45feff-ad65-4a6c-b478-5cab8EXAMPLE",  
      "TaskArn": "AWS-RestartEC2Instance",  
      "Type": "AUTOMATION",
```

```

    "Targets": [
      {
        "Key": "WindowTargetIds",
        "Values": [
          "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
        ]
      }
    ],
    "TaskParameters": {},
    "Priority": 3,
    "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "MaxConcurrency": "10",
    "MaxErrors": "5",
    "Name": "My-Automation-Task",
    "Description": "A description for my Automation task"
  }
]
}

```

Ejemplo 4: enumeración de todas las tareas de un periodo de mantenimiento que tienen una prioridad de 1 y uso del comando de ejecución

En este ejemplo de `describe-maintenance-window-tasks` se enumeran todas las tareas del periodo de mantenimiento especificado que tienen una `Priority` de 1 y el uso del `Run Command`.

```

aws ssm describe-maintenance-window-tasks \
  --window-id "mw-ab12cd34eEXAMPLE" \
  --filters "Key=Priority,Values=1" "Key=TaskType,Values=RUN_COMMAND"

```

Salida:

```

{
  "Tasks": [
    {
      "WindowId": "mw-ab12cd34eEXAMPLE",
      "WindowTaskId": "0d36e6b4-3a4f-411e-adcb-3558eEXAMPLE",
      "TaskArn": "AWS-RunPowerShellScript",
      "Type": "RUN_COMMAND",
      "Targets": [
        {

```

```

        "Key": "WindowTargetIds",
        "Values": [
            "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
        ]
    },
    "TaskParameters": {},
    "Priority": 1,
    "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
    ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "MaxConcurrency": "1",
    "MaxErrors": "1",
    "Name": "MyRunCommandTask"
}
]
}

```

Para obtener más información, consulte [Ver información sobre las ventanas de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeMaintenanceWindowTasks](#) la Referencia de AWS CLI comandos.

## describe-maintenance-windows-for-target

En el siguiente ejemplo de código, se muestra cómo usar `describe-maintenance-windows-for-target`.

### AWS CLI

Para enumerar todas las ventanas de mantenimiento asociadas a una instancia específica

En el siguiente `describe-maintenance-windows-for-target` ejemplo, se enumeran las ventanas de mantenimiento que tienen objetivos o tareas asociados a la instancia especificada.

```

aws ssm describe-maintenance-windows-for-target \
  --targets Key=InstanceIds,Values=i-1234567890EXAMPLE \
  --resource-type INSTANCE

```

Salida:

```
{
```



```
"WindowIdentities": [  
  {  
    "WindowId": "mw-0c5ed765acEXAMPLE",  
    "Name": "My-First-Maintenance-Window"  
  }  
]
```

Para obtener más información, consulte [Ver información sobre el mantenimiento de Windows \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeMaintenanceWindowsForTarget](#) la Referencia de AWS CLI comandos.

## describe-maintenance-windows

En el siguiente ejemplo de código, se muestra cómo usar `describe-maintenance-windows`.

### AWS CLI

Ejemplo 1: enumeración de todos los periodos de mantenimiento

En el siguiente `describe-maintenance-windows` ejemplo, se enumeran todos los períodos de mantenimiento de su AWS cuenta en la región actual.

```
aws ssm describe-maintenance-windows
```

Salida:

```
{  
  "WindowIdentities": [  
    {  
      "WindowId": "mw-0ecb1226ddEXAMPLE",  
      "Name": "MyMaintenanceWindow-1",  
      "Enabled": true,  
      "Duration": 2,  
      "Cutoff": 1,  
      "Schedule": "rate(180 minutes)",  
      "NextExecutionTime": "2020-02-12T23:19:20.596Z"  
    },  
    {  
      "WindowId": "mw-03eb9db428EXAMPLE",
```

```
        "Name": "MyMaintenanceWindow-2",
        "Enabled": true,
        "Duration": 3,
        "Cutoff": 1,
        "Schedule": "rate(7 days)",
        "NextExecutionTime": "2020-02-17T23:22:00.956Z"
    },
]
}
```

Ejemplo 2: enumeración de todos los periodos de mantenimiento habilitados

En el siguiente ejemplo de `describe-maintenance-windows` se enumeran todos los periodos de mantenimiento habilitados.

```
aws ssm describe-maintenance-windows \
  --filters "Key=Enabled,Values=true"
```

Ejemplo 3: enumeración de los periodos de mantenimiento que coincidan con un nombre específico

En este ejemplo de `describe-maintenance-windows` se enumeran todos los periodos de mantenimiento con el nombre especificado.

```
aws ssm describe-maintenance-windows \
  --filters "Key=Name,Values=MyMaintenanceWindow"
```

Para obtener más información, consulte [Ver información sobre el mantenimiento de Windows \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeMaintenanceWindows](#) la Referencia de AWS CLI comandos.

## **describe-ops-items**

En el siguiente ejemplo de código, se muestra cómo usar `describe-ops-items`.

### AWS CLI

Para enumerar un conjunto de OpsItems

En el siguiente describe-ops-items ejemplo, se muestra una lista de todas OpsItems las AWS cuentas abiertas.

```
aws ssm describe-ops-items \
  --ops-item-filters "Key=Status,Values=Open,Operator=Equal"
```

Salida:

```
{
  "OpsItemSummaries": [
    {
      "CreatedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-Role/
fbf77cbe264a33509569f23e4EXAMPLE",
      "CreatedTime": "2020-03-14T17:02:46.375000-07:00",
      "LastModifiedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-
Role/fbf77cbe264a33509569f23e4EXAMPLE",
      "LastModifiedTime": "2020-03-14T17:02:46.375000-07:00",
      "Source": "SSM",
      "Status": "Open",
      "OpsItemId": "oi-7cfc5EXAMPLE",
      "Title": "SSM Maintenance Window execution failed",
      "OperationalData": {
        "/aws/dedup": {
          "Value": "{\\"dedupString\\":\\"SSMOpsItems-SSM-maintenance-window-
execution-failed\\"}",
          "Type": "SearchableString"
        },
        "/aws/resources": {
          "Value": "[{\\"arn\\":\\"arn:aws:ssm:us-
east-2:111222333444:maintenancewindow/mw-034093d322EXAMPLE\\"}]",
          "Type": "SearchableString"
        }
      },
      "Category": "Availability",
      "Severity": "3"
    },
    {
      "CreatedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-Role/
fbf77cbe264a33509569f23e4EXAMPLE",
      "CreatedTime": "2020-02-26T11:43:15.426000-08:00",
      "LastModifiedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-
Role/fbf77cbe264a33509569f23e4EXAMPLE",
      "LastModifiedTime": "2020-02-26T11:43:15.426000-08:00",
```

```

    "Source": "EC2",
    "Status": "Open",
    "OpsItemId": "oi-6f966EXAMPLE",
    "Title": "EC2 instance stopped",
    "OperationalData": {
      "/aws/automations": {
        "Value": "[ { \"automationType\": \"AWS:SSM:Automation\",
\"automationId\": \"AWS-RestartEC2Instance\" } ]",
        "Type": "SearchableString"
      },
      "/aws/dedup": {
        "Value": "{ \"dedupString\": \"SSMOpsItems-EC2-instance-stopped
\" }",
        "Type": "SearchableString"
      },
      "/aws/resources": {
        "Value": "[ { \"arn\": \"arn:aws:ec2:us-
east-2:111222333444:instance/i-0beccfb02EXAMPLE\" } ]",
        "Type": "SearchableString"
      }
    },
    "Category": "Availability",
    "Severity": "3"
  }
]
}

```

Para obtener más información, consulte [Trabajar con OpsItems](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeOpsItems](#) la Referencia de AWS CLI comandos.

## describe-parameters

En el siguiente ejemplo de código, se muestra cómo usar `describe-parameters`.

### AWS CLI

Ejemplo 1: Creación de una lista de todos los parámetros

En el siguiente `describe-parameters` ejemplo, se enumeran todos los parámetros de la AWS cuenta corriente y la región.

**aws ssm describe-parameters**

Salida:

```

{
  "Parameters": [
    {
      "Name": "MySecureStringParameter",
      "Type": "SecureString",
      "KeyId": "alias/aws/ssm",
      "LastModifiedDate": 1582155479.205,
      "LastModifiedUser": "arn:aws:sts::111222333444:assumed-role/Admin/Richard-Roe-Managed",
      "Description": "This is a SecureString parameter",
      "Version": 2,
      "Tier": "Advanced",
      "Policies": [
        {
          "PolicyText": "{\"Type\":\"Expiration\",\"Version\":\"1.0\",
\\Attributes\":{\"Timestamp\":\"2020-07-07T22:30:00Z\"}}",
          "PolicyType": "Expiration",
          "PolicyStatus": "Pending"
        },
        {
          "PolicyText": "{\"Type\":\"ExpirationNotification\",\"Version\":
\\1.0\",\\Attributes\":{\"Before\":\"12\",\"Unit\":\"Hours\"}}",
          "PolicyType": "ExpirationNotification",
          "PolicyStatus": "Pending"
        }
      ]
    },
    {
      "Name": "MyStringListParameter",
      "Type": "StringList",
      "LastModifiedDate": 1582154764.222,
      "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
      "Description": "This is a StringList parameter",
      "Version": 1,
      "Tier": "Standard",
      "Policies": []
    },
    {
      "Name": "MyStringParameter",

```

```

        "Type": "String",
        "LastModifiedDate": 1582154711.976,
        "LastModifiedUser": "arn:aws:iam::111222333444:user/Alejandro-Rosalez",
        "Description": "This is a String parameter",
        "Version": 1,
        "Tier": "Standard",
        "Policies": []
    },
    {
        "Name": "latestAmi",
        "Type": "String",
        "LastModifiedDate": 1580862415.521,
        "LastModifiedUser": "arn:aws:sts::111222333444:assumed-role/lambda-ssm-
role/Automation-UpdateSSM-Param",
        "Version": 3,
        "Tier": "Standard",
        "Policies": []
    }
]
}

```

Ejemplo 2: Creación de una lista de todos los parámetros que coinciden con metadatos específicos

En este ejemplo de `describe-parameters` se enumeran todos los parámetros que coinciden con un filtro.

```
aws ssm describe-parameters --filters «Clave=Type, Values=» StringList
```

Salida:

```

{
  "Parameters": [
    {
      "Name": "MyStringListParameter",
      "Type": "StringList",
      "LastModifiedDate": 1582154764.222,
      "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
      "Description": "This is a StringList parameter",
      "Version": 1,
      "Tier": "Standard",
      "Policies": []
    }
  ]
}

```

```
]
}
```

Para obtener más información, consulte [Búsqueda de parámetros de Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

- API Para obtener más información, [DescribeParameters](#) consulte AWS CLI la Referencia de comandos.

## describe-patch-baselines

En el siguiente ejemplo de código, se muestra cómo usar `describe-patch-baselines`.

AWS CLI

Ejemplo 1: enumeración de todas las líneas de base de revisiones

En el siguiente ejemplo de `describe-patch-baselines` se recuperan los detalles de todas las líneas de base de revisiones de su cuenta en la región actual.

```
aws ssm describe-patch-baselines
```

Salida:

```
{
  "BaselineIdentities": [
    {
      "BaselineName": "AWS-SuseDefaultPatchBaseline",
      "DefaultBaseline": true,
      "BaselineDescription": "Default Patch Baseline for Suse Provided by
AWS.",
      "BaselineId": "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-0123fdb36e334a3b2",
      "OperatingSystem": "SUSE"
    },
    {
      "BaselineName": "AWS-DefaultPatchBaseline",
      "DefaultBaseline": false,
      "BaselineDescription": "Default Patch Baseline Provided by AWS.",
      "BaselineId": "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-020d361a05defe4ed",
      "OperatingSystem": "WINDOWS"
    }
  ]
}
```

```

    },
    ...
    {
        "BaselineName": "MyWindowsPatchBaseline",
        "DefaultBaseline": true,
        "BaselineDescription": "My patch baseline for EC2 instances for Windows
Server",
        "BaselineId": "pb-0ad00e0dd7EXAMPLE",
        "OperatingSystem": "WINDOWS"
    }
]
}

```

Ejemplo 2: Para enumerar todas las líneas base de los parches proporcionadas por AWS

En el siguiente `describe-patch-baselines` ejemplo se enumeran todas las líneas base de los parches proporcionadas por. AWS

```

aws ssm describe-patch-baselines \
  --filters "Key=OWNER,Values=[AWS]"

```

Ejemplo 3: enumeración de todas las líneas de base de revisiones de su propiedad

En el siguiente ejemplo de `describe-patch-baselines` se enumeran todas las líneas de base de revisiones personalizadas creadas en su cuenta en la región actual.

```

aws ssm describe-patch-baselines \
  --filters "Key=OWNER,Values=[Self]"

```

Para obtener más información, consulte [Acerca de las líneas de base de revisiones personalizadas y predefinidas](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte la Referencia [DescribePatchBaselines](#) de AWS CLI comandos.

## **describe-patch-group-state**

En el siguiente ejemplo de código, se muestra cómo usar `describe-patch-group-state`.

### AWS CLI

Obtención del estado de un grupo de revisiones



En el siguiente ejemplo de `describe-patch-group-state` se recupera el resumen general de conformidad de las revisiones de un grupo de revisiones.

```
aws ssm describe-patch-group-state \
  --patch-group "Production"
```

Salida:

```
{
  "Instances": 21,
  "InstancesWithCriticalNonCompliantPatches": 1,
  "InstancesWithFailedPatches": 2,
  "InstancesWithInstalledOtherPatches": 3,
  "InstancesWithInstalledPatches": 21,
  "InstancesWithInstalledPendingRebootPatches": 2,
  "InstancesWithInstalledRejectedPatches": 1,
  "InstancesWithMissingPatches": 3,
  "InstancesWithNotApplicablePatches": 4,
  "InstancesWithOtherNonCompliantPatches": 1,
  "InstancesWithSecurityNonCompliantPatches": 1,
  "InstancesWithUnreportedNotApplicablePatches": 2
}
```

Para obtener más información, consulte Acerca de los grupos de parches < <https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-patchgroups.html> > y [Descripción de los valores del estado de conformidad de los parches en](#) la Guía del usuario de Systems Manager.AWS

- Para obtener más información, consulte la Referencia de comandos. API [DescribePatchGroupState](#) AWS CLI

## **describe-patch-groups**

En el siguiente ejemplo de código, se muestra cómo usar `describe-patch-groups`.

AWS CLI

Visualización de los registros de grupos de revisiones

En el siguiente ejemplo de `describe-patch-groups` se enumeran los registros de los grupos de revisiones.

**aws ssm describe-patch-groups**

Salida:

```
{
  "Mappings": [
    {
      "PatchGroup": "Production",
      "BaselineIdentity": {
        "BaselineId": "pb-0123456789abcdef0",
        "BaselineName": "ProdPatching",
        "OperatingSystem": "WINDOWS",
        "BaselineDescription": "Patches for Production",
        "DefaultBaseline": false
      }
    },
    {
      "PatchGroup": "Development",
      "BaselineIdentity": {
        "BaselineId": "pb-0713accee01234567",
        "BaselineName": "DevPatching",
        "OperatingSystem": "WINDOWS",
        "BaselineDescription": "Patches for Development",
        "DefaultBaseline": true
      }
    },
    ...
  ]
}
```

Para obtener más información, consulte [Crear un grupo de parches < https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html>](https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html) y [Agregar un grupo de parches a una línea base de parches en la Guía del usuario de Systems Manager.AWS](#)

- Para obtener más información, consulte la Referencia de comandos. API [DescribePatchGroups](#) AWS CLI

**describe-patch-properties**

En el siguiente ejemplo de código, se muestra cómo usar `describe-patch-properties`.

## AWS CLI

Para enumerar la disponibilidad de los parches de Amazon Linux

En el siguiente `describe-patch-properties` ejemplo, se muestra una lista de los productos de Amazon Linux para los que hay parches disponibles en su AWS cuenta.

```
aws ssm describe-patch-properties \
  --operating-system AMAZON_LINUX \
  --property PRODUCT
```

Salida:

```
{
  "Properties": [
    {
      "Name": "AmazonLinux2012.03"
    },
    {
      "Name": "AmazonLinux2012.09"
    },
    {
      "Name": "AmazonLinux2013.03"
    },
    {
      "Name": "AmazonLinux2013.09"
    },
    {
      "Name": "AmazonLinux2014.03"
    },
    {
      "Name": "AmazonLinux2014.09"
    },
    {
      "Name": "AmazonLinux2015.03"
    },
    {
      "Name": "AmazonLinux2015.09"
    },
    {
      "Name": "AmazonLinux2016.03"
    },
    {
```

```
    "Name": "AmazonLinux2016.09"
  },
  {
    "Name": "AmazonLinux2017.03"
  },
  {
    "Name": "AmazonLinux2017.09"
  },
  {
    "Name": "AmazonLinux2018.03"
  }
]
}
```

Para obtener más información, consulte [Acerca de las líneas de base de revisiones](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribePatchProperties](#) la Referencia de AWS CLI comandos.

## describe-sessions

En el siguiente ejemplo de código, se muestra cómo usar `describe-sessions`.

### AWS CLI

Ejemplo 1: Para enumerar todas las sesiones activas del Administrador de sesiones

En este `describe-sessions` ejemplo, se recupera una lista de las sesiones activas que se crearon más recientemente (tanto las sesiones conectadas como las desconectadas) en los últimos 30 días y que fueron iniciadas por el usuario especificado. Este comando devuelve solo los resultados de las conexiones a destinos iniciadas mediante el Administrador de sesiones. No muestra las conexiones realizadas a través de otros medios, como las conexiones a escritorio remoto o SSH.

```
aws ssm describe-sessions \
  --state "Active" \
  --filters "key=owner,value=arn:aws:sts::123456789012:assumed-role/Administrator/Shirley-Rodriguez"
```

Salida:

```
{
  "Sessions": [
    {
      "SessionId": "John-07a16060613c408b5",
      "Target": "i-1234567890abcdef0",
      "Status": "Connected",
      "StartDate": 1550676938.352,
      "Owner": "arn:aws:sts::123456789012:assumed-role/Administrator/Shirley-
Rodriguez",
      "OutputUrl": {}
    },
    {
      "SessionId": "John-01edf534b8b56e8eb",
      "Target": "i-9876543210abcdef0",
      "Status": "Connected",
      "StartDate": 1550676842.194,
      "Owner": "arn:aws:sts::123456789012:assumed-role/Administrator/Shirley-
Rodriguez",
      "OutputUrl": {}
    }
  ]
}
```

Ejemplo 2: Para enumerar todas las sesiones del Administrador de sesiones terminadas

En este describe-sessions ejemplo, se recupera una lista de las sesiones finalizadas más recientemente en los últimos 30 días para todos los usuarios.

```
aws ssm describe-sessions \
  --state "History"
```

Salida:

```
{
  "Sessions": [
    {
      "SessionId": "Mary-Major-0022b1eb2b0d9e3bd",
      "Target": "i-1234567890abcdef0",
      "Status": "Terminated",
      "StartDate": 1550520701.256,
      "EndDate": 1550521931.563,
```

```

    "Owner": "arn:aws:sts::123456789012:assumed-role/Administrator/Mary-
Major"
  },
  {
    "SessionId": "Jane-Roe-0db53f487931ed9d4",
    "Target": "i-9876543210abcdef0",
    "Status": "Terminated",
    "StartDate": 1550161369.149,
    "EndDate": 1550162580.329,
    "Owner": "arn:aws:sts::123456789012:assumed-role/Administrator/Jane-Roe"
  },
  ...
],
"NextToken": "--token string truncated--"
}

```

Para obtener más información, consulte [Ver el historial de sesiones](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DescribeSessions](#) la Referencia de AWS CLI comandos.

## disassociate-ops-item-related-item

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-ops-item-related-item`.

### AWS CLI

Para eliminar una asociación de artículos relacionados

En el siguiente `disassociate-ops-item-related-item` ejemplo, se elimina la asociación entre el elemento OpsItem y un elemento relacionado.

```

aws ssm disassociate-ops-item-related-item \
  --ops-item-id "oi-f99f2EXAMPLE" \
  --association-id "e2036148-cccb-490e-ac2a-390e5EXAMPLE"

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Trabajar con incidentes de Incident Manager OpsCenter en](#) la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [DisassociateOpsItemRelatedItem](#) la Referencia de AWS CLI comandos.

## get-automation-execution

En el siguiente ejemplo de código, se muestra cómo usar `get-automation-execution`.

### AWS CLI

Visualización de los detalles sobre una ejecución de automatización

En el siguiente ejemplo de `get-automation-execution` se muestra información detallada sobre una ejecución de Automatización.

```
aws ssm get-automation-execution \
  --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Salida:

```
{
  "AutomationExecution": {
    "AutomationExecutionId": "73c8eef8-f4ee-4a05-820c-e354fEXAMPLE",
    "DocumentName": "AWS-StartEC2Instance",
    "DocumentVersion": "1",
    "ExecutionStartTime": 1583737233.748,
    "ExecutionEndTime": 1583737234.719,
    "AutomationExecutionStatus": "Success",
    "StepExecutions": [
      {
        "StepName": "startInstances",
        "Action": "aws:changeInstanceState",
        "ExecutionStartTime": 1583737234.134,
        "ExecutionEndTime": 1583737234.672,
        "StepStatus": "Success",
        "Inputs": {
          "DesiredState": "\"running\"",
          "InstanceIds": "[\"i-0cb99161f6EXAMPLE\"]"
        },
        "Outputs": {
          "InstanceStates": [
            "running"
          ]
        }
      }
    ]
  }
}
```

```

        },
        "StepExecutionId": "95e70479-cf20-4d80-8018-7e4e2EXAMPLE",
        "OverriddenParameters": {}
    }
],
"StepExecutionsTruncated": false,
"Parameters": {
    "AutomationAssumeRole": [
        ""
    ],
    "InstanceId": [
        "i-0cb99161f6EXAMPLE"
    ]
},
"Outputs": {},
"Mode": "Auto",
"ExecutedBy": "arn:aws:sts::29884EXAMPLE:assumed-role/mw_service_role/
OrchestrationService",
"Targets": [],
"ResolvedTargets": {
    "ParameterValues": [],
    "Truncated": false
}
}
}

```

Para obtener más información, consulte [Tutorial: Parchear un Linux AMI \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetAutomationExecution](#) la Referencia de AWS CLI comandos.

## get-calendar-state

En el siguiente ejemplo de código, se muestra cómo usar `get-calendar-state`.

### AWS CLI

Ejemplo 1: Para obtener el estado actual de un calendario de cambios

En este `get-calendar-state` ejemplo se devuelve el estado de un calendario en el momento actual. Como en el ejemplo no se especifica una hora, se informa del estado actual del calendario.



```
aws ssm get-calendar-state \  
  --calendar-names "MyCalendar"
```

Salida:

```
{  
  "State": "OPEN",  
  "AtTime": "2020-02-19T22:28:51Z",  
  "NextTransitionTime": "2020-02-24T21:15:19Z"  
}
```

Ejemplo 2: Para obtener el estado de un calendario de cambios a una hora específica

Este `get-calendar-state` ejemplo devuelve el estado de un calendario a la hora especificada.

```
aws ssm get-calendar-state \  
  --calendar-names "MyCalendar" \  
  --at-time "2020-07-19T21:15:19Z"
```

Salida:

```
{  
  "State": "CLOSED",  
  "AtTime": "2020-07-19T21:15:19Z"  
}
```

Para obtener más información, consulte [Obtener el estado del calendario de cambios](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetCalendarState](#) la Referencia de AWS CLI comandos.

## get-command-invocation

En el siguiente ejemplo de código, se muestra cómo usar `get-command-invocation`.

### AWS CLI

Visualización de los detalles de la invocación de un comando

En el siguiente ejemplo de `get-command-invocation` se muestran todas las invocaciones del comando especificado en la instancia especificada.

```
aws ssm get-command-invocation \  
  --command-id "ef7fd8-9b57-4151-a15c-db9a12345678" \  
  --instance-id "i-1234567890abcdef0"
```

Salida:

```
{  
  "CommandId": "ef7fd8-9b57-4151-a15c-db9a12345678",  
  "InstanceId": "i-1234567890abcdef0",  
  "Comment": "b48291dd-ba76-43e0-b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-  
d6ce8EXAMPLE",  
  "DocumentName": "AWS-UpdateSSMAgent",  
  "DocumentVersion": "",  
  "PluginName": "aws:updateSsmAgent",  
  "ResponseCode": 0,  
  "ExecutionStartDateTime": "2020-02-19T18:18:03.419Z",  
  "ExecutionElapsedTime": "PT0.091S",  
  "ExecutionEndDateTime": "2020-02-19T18:18:03.419Z",  
  "Status": "Success",  
  "StatusDetails": "Success",  
  "StandardOutputContent": "Updating amazon-ssm-agent from 2.3.842.0 to latest  
\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/  
ssm-agent-manifest.json\namazon-ssm-agent 2.3.842.0 has already been installed,  
update skipped\n",  
  "StandardOutputUrl": "",  
  "StandardErrorContent": "",  
  "StandardErrorUrl": "",  
  "CloudWatchOutputConfig": {  
    "CloudWatchLogGroupName": "",  
    "CloudWatchOutputEnabled": false  
  }  
}
```

Para obtener más información, consulte [Descripción de los estados del comando](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetCommandInvocation](#) la Referencia de AWS CLI comandos.

## get-connection-status

En el siguiente ejemplo de código, se muestra cómo usar `get-connection-status`.

### AWS CLI

Visualización del estado de conexión de una instancia administrada

En este ejemplo de `get-connection-status` se devuelve el estado de conexión de la instancia administrada especificada.

```
aws ssm get-connection-status \  
  --target i-1234567890abcdef0
```

Salida:

```
{  
  "Target": "i-1234567890abcdef0",  
  "Status": "connected"  
}
```

- Para API obtener más información, consulte [GetConnectionStatus](#) la Referencia de AWS CLI comandos.

## get-default-patch-baseline

En el siguiente ejemplo de código, se muestra cómo usar `get-default-patch-baseline`.

### AWS CLI

Ejemplo 1: visualización de la línea de base de revisiones de Windows predeterminada

En el siguiente ejemplo de `get-default-patch-baseline` se recuperan los detalles de la línea de base de revisiones predeterminada para Windows Server.

```
aws ssm get-default-patch-baseline
```

Salida:

```
{  
  "BaselineId": "pb-0713accee01612345",  
  "OperatingSystem": "WINDOWS"
```

```
}
```

Ejemplo 2: visualización de la línea de base de revisiones predeterminada para Amazon Linux

En el siguiente ejemplo de `get-default-patch-baseline` se recuperan los detalles de la línea de base de revisiones predeterminada para Amazon Linux.

```
aws ssm get-default-patch-baseline \  
  --operating-system AMAZON_LINUX
```

Salida:

```
{  
  "BaselineId": "pb-047c6eb9c8fc12345",  
  "OperatingSystem": "AMAZON_LINUX"  
}
```

Para obtener más información, consulte [Acerca de las líneas base de parches predefinidas y personalizadas < https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-baselines.html>](https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-baselines.html) y [Establecer una línea base de parches existente como predeterminada en la Guía del](#) usuario de Systems Manager.AWS

- Para obtener API más información, [GetDefaultPatchBaseline](#) consulte AWS CLI la Referencia de comandos.

## get-deployable-patch-snapshot-for-instance

En el siguiente ejemplo de código, se muestra cómo usar `get-deployable-patch-snapshot-for-instance`.

AWS CLI

Recuperación de la instantánea actual de la línea de base de revisiones que usa una instancia

En el siguiente ejemplo de `get-deployable-patch-snapshot-for-instance` se recuperan los detalles de la instantánea actual correspondiente a la línea de base de revisiones especificada que usa una instancia. Este comando debe ejecutarse desde la instancia con las credenciales de la instancia. Para asegurarse de que usa las credenciales de la instancia, ejecute `aws configure` y especifique solo la región de la instancia. Deje vacíos los campos `Access Key` y `Secret Key`.

Consejo: Use `uuidgen` para generar un `snapshot-id`.

```
aws ssm get-deployable-patch-snapshot-for-instance \  
  --instance-id "i-1234567890abcdef0" \  
  --snapshot-id "521c3536-930c-4aa9-950e-01234567abcd"
```

Salida:

```
{  
  "InstanceId": "i-1234567890abcdef0",  
  "SnapshotId": "521c3536-930c-4aa9-950e-01234567abcd",  
  "Product": "AmazonLinux2018.03",  
  "SnapshotDownloadUrl": "https://patch-baseline-snapshot-us-  
east-1.s3.amazonaws.com/  
ed85194ef27214f5984f28b4d664d14f7313568fea7d4b6ac6c10ad1f729d7e7-773304212436/  
AMAZON_LINUX-521c3536-930c-4aa9-950e-01234567abcd?X-Amz-Algorithm=AWS4-HMAC-  
SHA256&X-Amz-Date=20190215T164031Z&X-Amz-SignedHeaders=host&X-Amz-Expires=86400&X-  
Amz-Credential=AKIAJ5C56P35AEBRX2QQ%2F20190215%2Fus-east-1%2Fs3%2Faws4_request&X-  
Amz-Signature=efaaaf6e3878e77f48a6697e015efdbda9c426b09c5822055075c062f6ad2149"  
}
```

Para obtener más información, consulte [Nombre del parámetro: Snapshot ID](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetDeployablePatchSnapshotForInstance](#) la Referencia de AWS CLI comandos.

## get-document

En el siguiente ejemplo de código, se muestra cómo usar `get-document`.

### AWS CLI

Obtención del contenido de un documento

En el siguiente ejemplo de `get-document` se muestra el contenido de un documento de Systems Manager.

```
aws ssm get-document \  
  --name "AWS-RunShellScript"
```

Salida:

```
{
  "Name": "AWS-RunShellScript",
  "DocumentVersion": "1",
  "Status": "Active",
  "Content": "{\n  \"schemaVersion\": \"1.2\",\n  \"description\": \"Run a shell script or specify the commands to run.\",\n  \"parameters\": {\n    \"commands\": {\n      \"type\": \"StringList\",\n      \"description\": \"(Required) Specify a shell script or a command to run.\",\n      \"minItems\": 1,\n      \"displayType\": \"textarea\",\n      \"workingDirectory\": {\n        \"type\": \"String\",\n        \"default\": \"\",\n        \"description\": \"(Optional) The path to the working directory on your instance.\",\n        \"maxChars\": 4096,\n        \"executionTimeout\": {\n          \"type\": \"String\",\n          \"default\": \"3600\",\n          \"description\": \"(Optional) The time in seconds for a command to complete before it is considered to have failed. Default is 3600 (1 hour). Maximum is 172800 (48 hours).\",\n          \"allowedPattern\": \"([1-9][0-9]{0,4})|(1[0-6][0-9]{4})|(17[0-1][0-9]{3})|(172[0-7][0-9]{2})|(172800)\"\n        }\n      },\n      \"runtimeConfig\": {\n        \"aws:runShellScript\": {\n          \"properties\": [\n            {\n              \"id\": \"0.aws:runShellScript\",\n              \"runCommand\": \"{{ commands }}\",\n              \"workingDirectory\": \"{{ workingDirectory }}\",\n              \"timeoutSeconds\": \"{{ executionTimeout }}\"\n            }\n          ]\n        }\n      }\n    }\n  },\n  \"DocumentType\": \"Command\",\n  \"DocumentFormat\": \"JSON\"\n}"
```

Para obtener más información, consulte [Documentos de AWS Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetDocument](#) la Referencia de AWS CLI comandos.

## get-inventory-schema

En el siguiente ejemplo de código, se muestra cómo usar `get-inventory-schema`.

### AWS CLI

Visualización del esquema del inventario

En este ejemplo se obtiene una lista con los nombres de los tipos de inventario de la cuenta.

## Comando:

```
aws ssm get-inventory-schema
```

## Salida:

```
{
  "Schemas": [
    {
      "TypeName": "AWS:AWSComponent",
      "Version": "1.0",
      "Attributes": [
        {
          "Name": "Name",
          "DataType": "STRING"
        },
        {
          "Name": "ApplicationType",
          "DataType": "STRING"
        },
        {
          "Name": "Publisher",
          "DataType": "STRING"
        },
        {
          "Name": "Version",
          "DataType": "STRING"
        },
        {
          "Name": "InstalledTime",
          "DataType": "STRING"
        },
        {
          "Name": "Architecture",
          "DataType": "STRING"
        },
        {
          "Name": "URL",
          "DataType": "STRING"
        }
      ]
    },
    ...
  ]
}
```

```
  ],
  "NextToken": "--token string truncated--"
}
```

Visualización del esquema de inventario de un tipo de inventario específico

En este ejemplo se devuelve el esquema de inventario para un tipo de inventario de AWS componentes. AWS

Comando:

```
aws ssm get-inventory-schema --type-name "AWS:AWSComponent"
```

- Para API obtener más información, consulte [GetInventorySchema](#) la Referencia de AWS CLI comandos.

## get-inventory

En el siguiente ejemplo de código, se muestra cómo usar `get-inventory`.

AWS CLI

Visualización del inventario

En este ejemplo se obtienen los metadatos personalizados del inventario.

Comando:

```
aws ssm get-inventory
```

Salida:

```
{
  "Entities": [
    {
      "Data": {
        "AWS:InstanceInformation": {
          "Content": [
            {
              "ComputerName": "ip-172-31-44-222.us-
west-2.compute.internal",
              "InstanceId": "i-0cb2b964d3e14fd9f",
```



```

        "IpAddress": "172.31.44.222",
        "AgentType": "amazon-ssm-agent",
        "ResourceType": "EC2Instance",
        "AgentVersion": "2.0.672.0",
        "PlatformVersion": "2016.09",
        "PlatformName": "Amazon Linux AMI",
        "PlatformType": "Linux"
    }
  ],
  "TypeName": "AWS:InstanceInformation",
  "SchemaVersion": "1.0",
  "CaptureTime": "2017-02-20T18:03:58Z"
}
},
  "Id": "i-0cb2b964d3e14fd9f"
}
]
}

```

- Para API obtener más información, consulte [GetInventory](#) la Referencia de AWS CLI comandos.

## get-maintenance-window-execution-task-invocation

En el siguiente ejemplo de código, se muestra cómo usar `get-maintenance-window-execution-task-invocation`.

### AWS CLI

Para obtener información sobre la invocación de una tarea de una ventana de mantenimiento

En el siguiente `get-maintenance-window-execution-task-invocation` ejemplo, se muestra información sobre la invocación de la tarea especificada que forma parte de la ejecución del período de mantenimiento especificado.

```

aws ssm get-maintenance-window-execution-task-invocation \
  --window-execution-id "bc494bfa-e63b-49f6-8ad1-aa9f2EXAMPLE" \
  --task-id "96f2ad59-97e3-461d-a63d-40c8aEXAMPLE" \
  --invocation-id "a5273e2c-d2c6-4880-b3e1-5e550EXAMPLE"

```

Salida:

```
{
```

```

    "Status": "SUCCESS",
    "Parameters": "{\"comment\":\"\\\", \"documentName\":\"AWS-RunPowerShellScript\",
    \"instanceIds\":[\"i-1234567890EXAMPLE\"], \"maxConcurrency\":\"1\", \"maxErrors\":
    \"1\", \"parameters\":{\"executionTimeout\":[\"3600\"], \"workingDirectory\":[\"\"],
    \"commands\":[\"echo Hello\"]}, \"timeoutSeconds\":600}\",
    "ExecutionId": "03b6baa0-5460-4e15-83f2-ea685EXAMPLE",
    "InvocationId": "a5273e2c-d2c6-4880-b3e1-5e550EXAMPLE",
    "StartTime": 1549998326.421,
    "TaskType": "RUN_COMMAND",
    "EndTime": 1550001931.784,
    "WindowExecutionId": "bc494bfa-e63b-49f6-8ad1-aa9f2EXAMPLE",
    "StatusDetails": "Failed",
    "TaskExecutionId": "96f2ad59-97e3-461d-a63d-40c8aEXAMPLE"
  }

```

Para obtener más información, consulte [Ver información sobre tareas y ejecuciones de tareas \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetMaintenanceWindowExecutionTaskInvocation](#) la Referencia de AWS CLI comandos.

## get-maintenance-window-execution-task

En el siguiente ejemplo de código, se muestra cómo usar `get-maintenance-window-execution-task`.

### AWS CLI

Obtención de información sobre la ejecución de una tarea del periodo de mantenimiento

En el siguiente ejemplo de `get-maintenance-window-execution-task` se muestra información sobre una tarea que forma parte de la ejecución del periodo de mantenimiento especificado.

```

aws ssm get-maintenance-window-execution-task \
  --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE" \
  --task-id "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"

```

Salida:

```

{
  "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",

```

```
"TaskExecutionId": "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE",
"TaskArn": "AWS-RunPatchBaseline",
"ServiceRole": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
"Type": "RUN_COMMAND",
"TaskParameters": [
  {
    "BaselineOverride": {
      "Values": [
        ""
      ]
    },
    "InstallOverrideList": {
      "Values": [
        ""
      ]
    },
    "Operation": {
      "Values": [
        "Scan"
      ]
    },
    "RebootOption": {
      "Values": [
        "RebootIfNeeded"
      ]
    },
    "SnapshotId": {
      "Values": [
        "{{ aws:ORCHESTRATION_ID }}"
      ]
    },
    "aws:InstanceId": {
      "Values": [
        "i-02573cafcfEXAMPLE",
        "i-0471e04240EXAMPLE",
        "i-07782c72faEXAMPLE"
      ]
    }
  }
],
"Priority": 1,
"MaxConcurrency": "1",
"MaxErrors": "3",
```

```
"Status": "SUCCESS",
"StartTime": "2021-08-04T11:45:35.088000-07:00",
"EndTime": "2021-08-04T11:53:09.079000-07:00"
}
```

Para obtener más información, consulte [Ver información sobre tareas y ejecuciones de tareas \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetMaintenanceWindowExecutionTask](#) Referencia de AWS CLI comandos.

## get-maintenance-window-execution

En el siguiente ejemplo de código, se muestra cómo usar `get-maintenance-window-execution`.

### AWS CLI

Obtención de información sobre la ejecución de una tarea del periodo de mantenimiento

En el siguiente ejemplo de `get-maintenance-window-execution` se muestra información sobre una tarea que se ejecutó como parte de la ejecución del periodo de mantenimiento especificado.

```
aws ssm get-maintenance-window-execution \
  --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE"
```

Salida:

```
{
  "Status": "SUCCESS",
  "TaskIds": [
    "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"
  ],
  "StartTime": 1487692834.595,
  "EndTime": 1487692835.051,
  "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",
}
```

Para obtener más información, consulte [Ver información sobre tareas y ejecuciones de tareas \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetMaintenanceWindowExecution](#) la Referencia de AWS CLI comandos.

## get-maintenance-window-task

En el siguiente ejemplo de código, se muestra cómo usar `get-maintenance-window-task`.

### AWS CLI

Para obtener información sobre una tarea de la ventana de mantenimiento

En el siguiente `get-maintenance-window-task` ejemplo, se recuperan los detalles sobre la tarea de la ventana de mantenimiento especificada.

```
aws ssm get-maintenance-window-task \  
  --window-id mw-0c5ed765acEXAMPLE \  
  --window-task-id 0e842a8d-2d44-4886-bb62-af8dcEXAMPLE
```

Salida:

```
{  
  "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/  
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",  
  "MaxErrors": "1",  
  "TaskArn": "AWS-RunPowerShellScript",  
  "MaxConcurrency": "1",  
  "WindowTaskId": "0e842a8d-2d44-4886-bb62-af8dcEXAMPLE",  
  "TaskParameters": {},  
  "Priority": 1,  
  "TaskInvocationParameters": {  
    "RunCommand": {  
      "Comment": "",  
      "TimeoutSeconds": 600,  
      "Parameters": {  
        "commands": [  
          "echo Hello"  
        ],  
        "executionTimeout": [  
          "3600"  
        ],  
        "workingDirectory": [  
          ""  
        ]  
      }  
    }  
  }  
}
```

```

    ]
  }
}
},
"WindowId": "mw-0c5ed765acEXAMPLE",
"TaskType": "RUN_COMMAND",
"Targets": [
  {
    "Values": [
      "84c818da-b619-4d3d-9651-946f3EXAMPLE"
    ],
    "Key": "WindowTargetIds"
  }
],
"Name": "ExampleTask"
}

```

Para obtener más información, consulte [Ver información sobre el mantenimiento de Windows \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetMaintenanceWindowTask](#) la Referencia de AWS CLI comandos.

## get-maintenance-window

En el siguiente ejemplo de código, se muestra cómo usar `get-maintenance-window`.

### AWS CLI

Obtención de información sobre un periodo de mantenimiento

En el siguiente ejemplo de `get-maintenance-window` se recuperan los detalles sobre el periodo de mantenimiento especificado.

```
aws ssm get-maintenance-window \
  --window-id "mw-03eb9db428EXAMPLE"
```

Salida:

```
{
  "AllowUnassociatedTargets": true,
  "CreatedDate": 1515006912.957,
```

```

    "Cutoff": 1,
    "Duration": 6,
    "Enabled": true,
    "ModifiedDate": 2020-01-01T10:04:04.099Z,
    "Name": "My-Maintenance-Window",
    "Schedule": "rate(3 days)",
    "WindowId": "mw-03eb9db428EXAMPLE",
    "NextExecutionTime": "2020-02-25T00:08:15.099Z"
  }

```

Para obtener más información, consulte [Ver información sobre las ventanas de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetMaintenanceWindow](#) la Referencia de AWS CLI comandos.

## get-ops-item

En el siguiente ejemplo de código, se muestra cómo usar `get-ops-item`.

### AWS CLI

Para ver información sobre un OpsItem

En el siguiente `get-ops-item` ejemplo, se muestran detalles sobre lo especificado OpsItem.

```

aws ssm get-ops-item \
  --ops-item-id oi-0b725EXAMPLE

```

Salida:

```

{
  "OpsItem": {
    "CreatedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
    "CreatedTime": "2019-12-04T15:52:16.793000-08:00",
    "Description": "CloudWatch Event Rule SSMOpsItems-EC2-instance-terminated was triggered. Your EC2 instance has terminated. See below for more details.",
    "LastModifiedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
    "LastModifiedTime": "2019-12-04T15:52:16.793000-08:00",
    "Notifications": [],
    "RelatedOpsItems": [],
  }
}

```

```

    "Status": "Open",
    "OpsItemId": "oi-0b725EXAMPLE",
    "Title": "EC2 instance terminated",
    "Source": "EC2",
    "OperationalData": {
      "/aws/automations": {
        "Value": "[ { \"automationType\": \"AWS:SSM:Automation\",
\"automationId\": \"AWS-CreateManagedWindowsInstance\" }, { \"automationType\":
\"AWS:SSM:Automation\", \"automationId\": \"AWS-CreateManagedLinuxInstance\" } ]",
        "Type": "SearchableString"
      },
      "/aws/dedup": {
        "Value": "{\"dedupString\": \"SSMOpsItems-EC2-instance-terminated
\"}",
        "Type": "SearchableString"
      },
      "/aws/resources": {
        "Value": "[{\"arn\": \"arn:aws:ec2:us-east-2:111222333444:instance/
i-05adec7e97EXAMPLE\"}]",
        "Type": "SearchableString"
      },
      "event-time": {
        "Value": "2019-12-04T23:52:16Z",
        "Type": "String"
      },
      "instance-state": {
        "Value": "terminated",
        "Type": "String"
      }
    },
    "Category": "Availability",
    "Severity": "4"
  }
}

```

Para obtener más información, consulte [Trabajar con OpsItems](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetOpsItem](#) la Referencia de AWS CLI comandos.

## get-ops-summary

En el siguiente ejemplo de código, se muestra cómo usar `get-ops-summary`.



## AWS CLI

Para ver un resumen de todos OpsItems

En el siguiente `get-ops-summary` ejemplo, se muestra un resumen de todo lo que OpsItems hay en tu AWS cuenta.

```
aws ssm get-ops-summary
```

Salida:

```
{
  "Entities": [
    {
      "Id": "oi-4309fEXAMPLE",
      "Data": {
        "AWS:OpsItem": {
          "CaptureTime": "2020-02-26T18:58:32.918Z",
          "Content": [
            {
              "AccountId": "111222333444",
              "Category": "Availability",
              "CreatedBy": "arn:aws:sts::111222333444:assumed-role/
OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
              "CreatedTime": "2020-02-26T19:10:44.149Z",
              "Description": "CloudWatch Event Rule SSMOpsItems-EC2-
instance-terminated was triggered. Your EC2 instance has terminated. See below for
more details.",
              "LastModifiedBy": "arn:aws:sts::111222333444:assumed-
role/OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
              "LastModifiedTime": "2020-02-26T19:10:44.149Z",
              "Notifications": "",
              "OperationalData": "{\"/aws/automations\":
{\"type\": \"SearchableString\", \"value\": \"[ { \\\"automationType\\\": \\
\"AWS:SSM:Automation\\\", \\\"automationId\\\": \\\"AWS-CreateManagedWindowsInstance
\\\" }, { \\\"automationType\\\": \\\"AWS:SSM:Automation\\\", \\\"automationId
\\\": \\\"AWS-CreateManagedLinuxInstance\\\" } ]\"}, \"/aws/resources\":
{\"type\": \"SearchableString\", \"value\": \"[{\\\"arn\\\": \\\"arn:aws:ec2:us-
east-2:111222333444:instance/i-0acbd0800fEXAMPLE\\\"}]\"}, \"/aws/dedup\": {\"type\":
\"SearchableString\", \"value\": \"{\\\"dedupString\\\": \\\"SSMOpsItems-EC2-instance-
terminated\\\"}\"}}",
              "OpsItemId": "oi-4309fEXAMPLE",
              "RelatedItems": "",
            }
          ]
        }
      }
    }
  ]
}
```

```

        "Severity": "3",
        "Source": "EC2",
        "Status": "Open",
        "Title": "EC2 instance terminated"
    }
  ]
}
},
{
  "Id": "oi-bb2a0e6a4541",
  "Data": {
    "AWS:OpsItem": {
      "CaptureTime": "2019-11-26T19:20:06.161Z",
      "Content": [
        {
          "AccountId": "111222333444",
          "Category": "Availability",
          "CreatedBy": "arn:aws:sts::111222333444:assumed-role/
OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
          "CreatedTime": "2019-11-26T20:00:07.237Z",
          "Description": "CloudWatch Event Rule SSMOpsItems-SSM-
maintenance-window-execution-failed was triggered. Your SSM Maintenance Window
execution has failed. See below for more details.",
          "LastModifiedBy": "arn:aws:sts::111222333444:assumed-
role/OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
          "LastModifiedTime": "2019-11-26T20:00:07.237Z",
          "Notifications": "",
          "OperationalData": "{\"/aws/resources\":{\"type
\": \"SearchableString\", \"value\": \"[{\\\"arn\\\": \\\"arn:aws:ssm:us-
east-2:111222333444:maintenancewindow/mw-0e83ba440dEXAMPLE\\\"]}\"}, \"/aws/dedup\":
{\"type\": \"SearchableString\", \"value\": \"[{\\\"dedupString\\\": \\\"SSMOpsItems-SSM-
maintenance-window-execution-failed\\\"]}\"}",
          "OpsItemId": "oi-bb2a0EXAMPLE",
          "RelatedItems": "",
          "Severity": "3",
          "Source": "SSM",
          "Status": "Open",
          "Title": "SSM Maintenance Window execution failed"
        }
      ]
    }
  }
}
}
}

```

```
]
}
```

Para obtener más información, consulte [Trabajar con OpsItems](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetOpsSummary](#) la Referencia de AWS CLI comandos.

## get-parameter-history

En el siguiente ejemplo de código, se muestra cómo usar `get-parameter-history`.

### AWS CLI

Obtención del historial de valores de un parámetro

En el siguiente ejemplo de `get-parameter-history` se enumera el historial de cambios del parámetro especificado, incluido su valor.

```
aws ssm get-parameter-history \
  --name "MyStringParameter"
```

Salida:

```
{
  "Parameters": [
    {
      "Name": "MyStringParameter",
      "Type": "String",
      "LastModifiedDate": 1582154711.976,
      "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
      "Description": "This is the first version of my String parameter",
      "Value": "Veni",
      "Version": 1,
      "Labels": [],
      "Tier": "Standard",
      "Policies": []
    },
    {
      "Name": "MyStringParameter",
      "Type": "String",
```

```
    "LastModifiedDate": 1582156093.471,  
    "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",  
    "Description": "This is the second version of my String parameter",  
    "Value": "Vidi",  
    "Version": 2,  
    "Labels": [],  
    "Tier": "Standard",  
    "Policies": []  
  },  
  {  
    "Name": "MyStringParameter",  
    "Type": "String",  
    "LastModifiedDate": 1582156117.545,  
    "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",  
    "Description": "This is the third version of my String parameter",  
    "Value": "Vici",  
    "Version": 3,  
    "Labels": [],  
    "Tier": "Standard",  
    "Policies": []  
  }  
]  
}
```

Para obtener más información, consulte [Trabajo con versiones de parámetros](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetParameterHistory](#) la Referencia de AWS CLI comandos.

## get-parameter

En el siguiente ejemplo de código, se muestra cómo usar `get-parameter`.

### AWS CLI

Ejemplo 1: visualización del valor de un parámetro

En el siguiente ejemplo de `get-parameter` se enumera el valor del parámetro único especificado.

```
aws ssm get-parameter \
```

```
--name "MyStringParameter"
```

Salida:

```
{
  "Parameter": {
    "Name": "MyStringParameter",
    "Type": "String",
    "Value": "Veni",
    "Version": 1,
    "LastModifiedDate": 1530018761.888,
    "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/MyStringParameter"
    "DataType": "text"
  }
}
```

Para obtener más información, consulte [Uso de Parameter Store](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: Para descifrar el valor de un parámetro SecureString

En el siguiente ejemplo de `get-parameter` se cambia el valor del parámetro SecureString especificado.

```
aws ssm get-parameter \  
  --name "MySecureStringParameter" \  
  --with-decryption
```

Salida:

```
{
  "Parameter": {
    "Name": "MySecureStringParameter",
    "Type": "SecureString",
    "Value": "16679b88-310b-4895-a943-e0764EXAMPLE",
    "Version": 2,
    "LastModifiedDate": 1582155479.205,
    "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/
MySecureStringParameter"
    "DataType": "text"
  }
}
```

Para obtener más información, consulte [Uso de Parameter Store](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 3: visualización del valor de un parámetro mediante etiquetas

En el siguiente ejemplo de `get-parameter` se enumera el valor del parámetro único especificado con una etiqueta especificada.

```
aws ssm get-parameter \  
  --name "MyParameter:label"
```

Salida:

```
{  
  "Parameter": {  
    "Name": "MyParameter",  
    "Type": "String",  
    "Value": "parameter version 2",  
    "Version": 2,  
    "Selector": ":label",  
    "LastModifiedDate": "2021-07-12T09:49:15.865000-07:00",  
    "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/MyParameter",  
    "DataType": "text"  
  }  
}
```

Para obtener más información, consulte [Trabajo con etiquetas de parámetros](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 4: visualización del valor de un parámetro mediante etiquetas

En el siguiente ejemplo de `get-parameter` se enumera el valor de la versión de parámetro único especificado.

```
aws ssm get-parameter \  
  --name "MyParameter:2"
```

Salida:

```
{  
  "Parameter": {  
    "Name": "MyParameter",
```

```

    "Type": "String",
    "Value": "parameter version 2",
    "Version": 2,
    "Selector": ":2",
    "LastModifiedDate": "2021-07-12T09:49:15.865000-07:00",
    "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/MyParameter",
    "DataType": "text"
  }
}

```

Para obtener más información, consulte [Trabajo con etiquetas de parámetros](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetParameter](#) la Referencia de AWS CLI comandos.

## get-parameters-by-path

En el siguiente ejemplo de código, se muestra cómo usar `get-parameters-by-path`.

### AWS CLI

Para enumerar los parámetros de una ruta específica

En el siguiente `get-parameters-by-path` ejemplo, se enumeran los parámetros de la jerarquía especificada.

```

aws ssm get-parameters-by-path \
  --path "/site/newyork/department/"

```

Salida:

```

{
  "Parameters": [
    {
      "Name": "/site/newyork/department/marketing",
      "Type": "String",
      "Value": "Floor 2",
      "Version": 1,
      "LastModifiedDate": 1530018761.888,
      "ARN": "arn:aws:ssm:us-east-1:111222333444:parameter/site/newyork/department/marketing"
    }
  ]
}

```

```

    },
    {
      "Name": "/site/newyork/department/infotech",
      "Type": "String",
      "Value": "Floor 3",
      "Version": 1,
      "LastModifiedDate": 1530018823.429,
      "ARN": "arn:aws:ssm:us-east-1:111222333444:parameter/site/newyork/
department/infotech"
    },
    ...
  ]
}

```

Para obtener más información, consulte [Trabajar con jerarquías de parámetros](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetParametersByPath](#) la Referencia de AWS CLI comandos.

## get-parameters

En el siguiente ejemplo de código, se muestra cómo usar `get-parameters`.

### AWS CLI

Ejemplo 1: enumeración de los valores de un parámetro

En el siguiente ejemplo de `get-parameters` se enumeran los valores de los tres parámetros especificados.

```

aws ssm get-parameters \
  --names "MyStringParameter" "MyStringListParameter" "MyInvalidParameterName"

```

Salida:

```

{
  "Parameters": [
    {
      "Name": "MyStringListParameter",
      "Type": "StringList",
      "Value": "alpha,beta,gamma",

```



```

        "Version": 1,
        "LastModifiedDate": 1582154764.222,
        "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/
MyStringListParameter"
        "DataType": "text"
    },
    {
        "Name": "MyStringParameter",
        "Type": "String",
        "Value": "Vici",
        "Version": 3,
        "LastModifiedDate": 1582156117.545,
        "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/MyStringParameter"
        "DataType": "text"
    }
],
"InvalidParameters": [
    "MyInvalidParameterName"
]
}

```

Para obtener más información, consulte [Uso de Parameter Store](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: enumeración de los nombres y valores de varios parámetros mediante la opción “--query”

En el siguiente ejemplo de `get-parameters` se enumeran los nombres y valores de los parámetros especificados.

```

aws ssm get-parameters \
  --names MyStringParameter MyStringListParameter \
  --query "Parameters[*].{Name:Name,Value:Value}"

```

Salida:

```

[
  {
    "Name": "MyStringListParameter",
    "Value": "alpha,beta,gamma"
  },
  {
    "Name": "MyStringParameter",

```

```
    "Value": "Vidi"  
  }  
]
```

Para obtener más información, consulte [Uso de Parameter Store](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 3: visualización del valor de un parámetro mediante etiquetas

En el siguiente ejemplo de `get-parameter` se enumera el valor del parámetro único especificado con una etiqueta especificada.

```
aws ssm get-parameter \  
  --name "MyParameter:Label"
```

Salida:

```
{  
  "Parameters": [  
    {  
      "Name": "MyLabelParameter",  
      "Type": "String",  
      "Value": "parameter by label",  
      "Version": 1,  
      "Selector": ":label",  
      "LastModifiedDate": "2021-07-12T09:49:15.865000-07:00",  
      "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/MyParameter",  
      "DataType": "text"  
    },  
    {  
      "Name": "MyVersionParameter",  
      "Type": "String",  
      "Value": "parameter by version",  
      "Version": 2,  
      "Selector": ":2",  
      "LastModifiedDate": "2021-03-24T16:20:28.236000-07:00",  
      "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/unlabel-param",  
      "DataType": "text"  
    }  
  ],  
  "InvalidParameters": []  
}
```

Para obtener más información, consulte [Trabajo con etiquetas de parámetros](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetParameters](#) la Referencia de AWS CLI comandos.

## get-patch-baseline-for-patch-group

En el siguiente ejemplo de código, se muestra cómo usar `get-patch-baseline-for-patch-group`.

### AWS CLI

Visualización de la línea de base de revisiones de un grupo de revisiones

En el siguiente ejemplo de `get-patch-baseline-for-patch-group` se recuperan los detalles sobre la línea de base de revisiones del grupo de revisiones especificado.

```
aws ssm get-patch-baseline-for-patch-group \
  --patch-group "DEV"
```

Salida:

```
{
  "PatchGroup": "DEV",
  "BaselineId": "pb-0123456789abcdef0",
  "OperatingSystem": "WINDOWS"
}
```

Para obtener más información, consulte [Crear un grupo de parches < https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html>](https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html) y [Agregar un grupo de parches a una línea base de parches en la Guía del](#) usuario de Systems Manager.AWS

- Para obtener más información, consulte la Referencia de comandos. API [GetPatchBaselineForPatchGroup](#) AWS CLI

## get-patch-baseline

En el siguiente ejemplo de código, se muestra cómo usar `get-patch-baseline`.

## AWS CLI

### Visualización de una línea de base de revisiones

En el siguiente ejemplo de `get-patch-baseline` se recuperan los detalles de la línea de base de revisiones especificada.

```
aws ssm get-patch-baseline \  
--baseline-id "pb-0123456789abcdef0"
```

Salida:

```
{  
  "BaselineId": "pb-0123456789abcdef0",  
  "Name": "WindowsPatching",  
  "OperatingSystem": "WINDOWS",  
  "GlobalFilters": {  
    "PatchFilters": []  
  },  
  "ApprovalRules": {  
    "PatchRules": [  
      {  
        "PatchFilterGroup": {  
          "PatchFilters": [  
            {  
              "Key": "PRODUCT",  
              "Values": [  
                "WindowsServer2016"  
              ]  
            }  
          ]  
        },  
        "ComplianceLevel": "CRITICAL",  
        "ApproveAfterDays": 0,  
        "EnableNonSecurity": false  
      }  
    ]  
  },  
  "ApprovedPatches": [],  
  "ApprovedPatchesComplianceLevel": "UNSPECIFIED",  
  "ApprovedPatchesEnableNonSecurity": false,  
  "RejectedPatches": [],  
  "RejectedPatchesAction": "ALLOW_AS_DEPENDENCY",
```

```

    "PatchGroups": [
      "QA",
      "DEV"
    ],
    "CreateDate": 1550244180.465,
    "ModifiedDate": 1550244180.465,
    "Description": "Patches for Windows Servers",
    "Sources": []
  }

```

Para obtener más información, consulte [Acerca de las líneas de base de revisiones](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetPatchBaseline](#) la Referencia de AWS CLI comandos.

## get-service-setting

En el siguiente ejemplo de código, se muestra cómo usar `get-service-setting`.

### AWS CLI

Para recuperar la configuración del servicio para el rendimiento del almacén de parámetros

El siguiente `get-service-setting` ejemplo recupera la configuración de servicio actual del rendimiento del almacén de parámetros en la región especificada.

```

aws ssm get-service-setting \
  --setting-id arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-
store/high-throughput-enabled

```

Salida:

```

{
  "ServiceSetting": {
    "SettingId": "/ssm/parameter-store/high-throughput-enabled",
    "SettingValue": "false",
    "LastModifiedDate": 1555532818.578,
    "LastModifiedUser": "System",
    "ARN": "arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-
store/high-throughput-enabled",
    "Status": "Default"
  }
}

```

```
}  
}
```

Para obtener más información, consulte [Aumentar el rendimiento del almacén de parámetros](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [GetServiceSetting](#) la Referencia de AWS CLI comandos.

## label-parameter-version

En el siguiente ejemplo de código, se muestra cómo usar `label-parameter-version`.

### AWS CLI

Ejemplo 1: Para añadir una etiqueta a la última versión de un parámetro

En el siguiente `label-parameter-version` ejemplo, se agrega una etiqueta a la última versión del parámetro especificado.

```
aws ssm label-parameter-version \  
  --name "MyStringParameter" \  
  --labels "ProductionReady"
```

Salida:

```
{  
  "InvalidLabels": [],  
  "ParameterVersion": 3  
}
```

Para obtener más información, consulte [Trabajo con etiquetas de parámetros](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: Para añadir una etiqueta a una versión específica de un parámetro

El siguiente `label-parameter-version` ejemplo agrega una etiqueta a la versión especificada de un parámetro.

```
aws ssm label-parameter-version \  
  --name "MyStringParameter" \  
  --labels "ProductionReady" \  
  --parameter-version 1
```

```
--parameter-version "2" --labels "DeveLopmentReady"
```

Para obtener más información, consulte [Trabajo con etiquetas de parámetros](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [LabelParameterVersion](#) la Referencia de AWS CLI comandos.

## list-association-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-association-versions`.

### AWS CLI

Enumeración de todas las versiones de una asociación para un ID de asociación específico

En el siguiente ejemplo de `list-association-versions` se enumeran todas las versiones de las asociaciones especificadas.

```
aws ssm list-association-versions \  
  --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Salida:

```
{  
  "AssociationVersions": [  
    {  
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",  
      "AssociationVersion": "1",  
      "CreateDate": 1550505536.726,  
      "Name": "AWS-UpdateSSMAgent",  
      "Parameters": {  
        "allowDowngrade": [  
          "false"  
        ],  
        "version": [  
          ""  
        ]  
      },  
      "Targets": [  
        {  
          "Key": "InstanceIds",
```

```

        "Values": [
            "i-1234567890abcdef0"
        ]
    },
    ],
    "ScheduleExpression": "cron(0 00 12 ? * SUN *)",
    "AssociationName": "UpdateSSMAgent"
}
]
}

```

Para obtener más información, consulte [Trabajo con asociaciones en Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [ListAssociationVersions](#) la Referencia de AWS CLI comandos.

## list-associations

En el siguiente ejemplo de código, se muestra cómo usar `list-associations`.

### AWS CLI

Ejemplo 1: enumeración de las asociaciones de una instancia específica

En el siguiente ejemplo de lista de asociaciones se enumeran todas las asociaciones con la letra AssociationName U. pdateSSMAgent

```

aws ssm list-associations /
  --association-filter-list "key=AssociationName,value=UpdateSSMAgent"

```

Salida:

```

{
  "Associations": [
    {
      "Name": "AWS-UpdateSSMAgent",
      "InstanceId": "i-1234567890abcdef0",
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "AssociationVersion": "1",
      "Targets": [
        {

```



```

        "Key": "InstanceIds",
        "Values": [
            "i-016648b75dd622dab"
        ]
    },
],
"Overview": {
    "Status": "Pending",
    "DetailedStatus": "Associated",
    "AssociationStatusAggregatedCount": {
        "Pending": 1
    }
},
"ScheduleExpression": "cron(0 00 12 ? * SUN *)",
"AssociationName": "UpdateSSMAgent"
}
]
}

```

Para obtener más información, consulte [Trabajo con asociaciones en Systems Manager](#) en la Guía del usuario de Systems Manager.

Ejemplo 2: enumeración de las asociaciones de un documento específico

En el siguiente ejemplo de `list-associations` se enumeran todas las asociaciones del documento especificado.

```

aws ssm list-associations /
  --association-filter-list "key=Name,value=AWS-UpdateSSMAgent"

```

Salida:

```

{
  "Associations": [
    {
      "Name": "AWS-UpdateSSMAgent",
      "InstanceId": "i-1234567890abcdef0",
      "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
      "AssociationVersion": "1",
      "Targets": [
        {
          "Key": "InstanceIds",
          "Values": [

```

```

        "i-1234567890abcdef0"
      ]
    }
  ],
  "LastExecutionDate": 1550505828.548,
  "Overview": {
    "Status": "Success",
    "DetailedStatus": "Success",
    "AssociationStatusAggregatedCount": {
      "Success": 1
    }
  },
  "ScheduleExpression": "cron(0 00 12 ? * SUN *)",
  "AssociationName": "UpdateSSMAgent"
},
{
  "Name": "AWS-UpdateSSMAgent",
  "InstanceId": "i-9876543210abcdef0",
  "AssociationId": "fbc07ef7-b985-4684-b82b-0123456789ab",
  "AssociationVersion": "1",
  "Targets": [
    {
      "Key": "InstanceIds",
      "Values": [
        "i-9876543210abcdef0"
      ]
    }
  ],
  "LastExecutionDate": 1550507531.0,
  "Overview": {
    "Status": "Success",
    "AssociationStatusAggregatedCount": {
      "Success": 1
    }
  }
}
]
}

```

Para obtener más información, consulte [Trabajo con asociaciones en Systems Manager](#) en la Guía del usuario de Systems Manager.

- Para API obtener más información, consulte la Referencia [ListAssociations](#) de AWS CLI comandos.

## list-command-invocations

En el siguiente ejemplo de código, se muestra cómo usar `list-command-invocations`.

### AWS CLI

Enumeración de las invocaciones de un comando específico

En el siguiente ejemplo de `list-command-invocations` se enumeran todas las invocaciones de un comando.

```
aws ssm list-command-invocations \  
  --command-id "ef7fd8-9b57-4151-a15c-db9a12345678" \  
  --details
```

Salida:

```
{  
  "CommandInvocations": [  
    {  
      "CommandId": "ef7fd8-9b57-4151-a15c-db9a12345678",  
      "InstanceId": "i-02573cafcfEXAMPLE",  
      "InstanceName": "",  
      "Comment": "b48291dd-ba76-43e0-b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",  
      "DocumentName": "AWS-UpdateSSMAgent",  
      "DocumentVersion": "",  
      "RequestedDateTime": 1582136283.089,  
      "Status": "Success",  
      "StatusDetails": "Success",  
      "StandardOutputUrl": "",  
      "StandardErrorUrl": "",  
      "CommandPlugins": [  
        {  
          "Name": "aws:updateSsmAgent",  
          "Status": "Success",  
          "StatusDetails": "Success",  
          "ResponseCode": 0,  
          "ResponseStartDateTime": 1582136283.419,  
          "ResponseFinishDateTime": 1582136283.51,  
          "Output": "Updating amazon-ssm-agent from 2.3.842.0 to latest  
\\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/
```

```

ssm-agent-manifest.json\namazon-ssm-agent 2.3.842.0 has already been installed,
update skipped\n",
    "StandardOutputUrl": "",
    "StandardErrorUrl": "",
    "OutputS3Region": "us-east-2",
    "OutputS3BucketName": "",
    "OutputS3KeyPrefix": ""
  }
],
"ServiceRole": "",
"NotificationConfig": {
  "NotificationArn": "",
  "NotificationEvents": [],
  "NotificationType": ""
},
"CloudWatchOutputConfig": {
  "CloudWatchLogGroupName": "",
  "CloudWatchOutputEnabled": false
}
},
{
  "CommandId": "ef7fdfd8-9b57-4151-a15c-db9a12345678",
  "InstanceId": "i-0471e04240EXAMPLE",
  "InstanceName": "",
  "Comment": "b48291dd-ba76-43e0-
b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",
  "DocumentName": "AWS-UpdateSSMAgent",
  "DocumentVersion": "",
  "RequestedDateTime": 1582136283.02,
  "Status": "Success",
  "StatusDetails": "Success",
  "StandardOutputUrl": "",
  "StandardErrorUrl": "",
  "CommandPlugins": [
    {
      "Name": "aws:updateSsmAgent",
      "Status": "Success",
      "StatusDetails": "Success",
      "ResponseCode": 0,
      "ResponseStartDateTime": 1582136283.812,
      "ResponseFinishDateTime": 1582136295.031,
      "Output": "Updating amazon-ssm-agent from 2.3.672.0 to latest
\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/
ssm-agent-manifest.json\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/

```

```

amazon-ssm-us-east-2/amazon-ssm-agent-updater/2.3.842.0/amazon-ssm-agent-updater-
snap-amd64.tar.gz\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/
amazon-ssm-us-east-2/amazon-ssm-agent/2.3.672.0/amazon-ssm-agent-snap-amd64.tar.gz
\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/
amazon-ssm-agent/2.3.842.0/amazon-ssm-agent-snap-amd64.tar.gz\nInitiating amazon-
ssm-agent update to 2.3.842.0\namazon-ssm-agent updated successfully to 2.3.842.0",
    "StandardOutputUrl": "",
    "StandardErrorUrl": "",
    "OutputS3Region": "us-east-2",
    "OutputS3BucketName": "",
    "OutputS3KeyPrefix": "8bee3135-398c-4d31-99b6-e42d2EXAMPLE/i-0471e04240EXAMPLE/awsupdateSsmAgent"
  }
],
"ServiceRole": "",
"NotificationConfig": {
  "NotificationArn": "",
  "NotificationEvents": [],
  "NotificationType": ""
},
"CloudWatchOutputConfig": {
  "CloudWatchLogGroupName": "",
  "CloudWatchOutputEnabled": false
}
}
]
}

```

Para obtener más información, consulte [Descripción de los estados del comando](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [ListCommandInvocations](#) la Referencia de AWS CLI comandos.

## list-commands

En el siguiente ejemplo de código, se muestra cómo usar `list-commands`.

### AWS CLI

Ejemplo 1: obtención del estado de un comando específico

En el siguiente ejemplo de `list-commands` se recupera y muestra el estado del comando especificado.

```
aws ssm list-commands \  
  --command-id "0831e1a8-a1ac-4257-a1fd-c831bEXAMPLE"
```

Ejemplo 2: obtención del estado de los comandos solicitados después de una fecha específica

En el siguiente ejemplo de `list-commands` se recuperan los detalles de los comandos solicitados después de la fecha especificada.

```
aws ssm list-commands \  
  --filter "key=InvokedAfter,value=2020-02-01T00:00:00Z"
```

Ejemplo 3: Para enumerar todos los comandos solicitados en una AWS cuenta

En el siguiente `list-commands` ejemplo, se enumeran todos los comandos solicitados por los usuarios de la AWS cuenta y la región actuales.

```
aws ssm list-commands
```

Salida:

```
{  
  "Commands": [  
    {  
      "CommandId": "8bee3135-398c-4d31-99b6-e42d2EXAMPLE",  
      "DocumentName": "AWS-UpdateSSMAgent",  
      "DocumentVersion": "",  
      "Comment": "b48291dd-ba76-43e0-  
b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",  
      "ExpiresAfter": "2020-02-19T11:28:02.500000-08:00",  
      "Parameters": {},  
      "InstanceIds": [  
        "i-028ea792daEXAMPLE",  
        "i-02feef8c46EXAMPLE",  
        "i-038613f3f0EXAMPLE",  
        "i-03a530a2d4EXAMPLE",  
        "i-083b678d37EXAMPLE",  
        "i-0dee81debaEXAMPLE"  
      ],  
      "Targets": [],  
    }  
  ]  
}
```

```

    "RequestedDateTime": "2020-02-19T10:18:02.500000-08:00",
    "Status": "Success",
    "StatusDetails": "Success",
    "OutputS3BucketName": "",
    "OutputS3KeyPrefix": "",
    "MaxConcurrency": "50",
    "MaxErrors": "100%",
    "TargetCount": 6,
    "CompletedCount": 6,
    "ErrorCount": 0,
    "DeliveryTimedOutCount": 0,
    "ServiceRole": "",
    "NotificationConfig": {
      "NotificationArn": "",
      "NotificationEvents": [],
      "NotificationType": ""
    },
    "CloudWatchOutputConfig": {
      "CloudWatchLogGroupName": "",
      "CloudWatchOutputEnabled": false
    }
  }
}
{
  "CommandId": "e9ade581-c03d-476b-9b07-26667EXAMPLE",
  "DocumentName": "AWS-FindWindowsUpdates",
  "DocumentVersion": "1",
  "Comment": "",
  "ExpiresAfter": "2020-01-24T12:37:31.874000-08:00",
  "Parameters": {
    "KbArticleIds": [
      ""
    ],
    "UpdateLevel": [
      "All"
    ]
  },
  "InstanceIds": [],
  "Targets": [
    {
      "Key": "InstanceIds",
      "Values": [
        "i-00ec29b21eEXAMPLE",
        "i-09911ddd90EXAMPLE"
      ]
    }
  ]
}

```

```

    }
  ],
  "RequestedDateTime": "2020-01-24T11:27:31.874000-08:00",
  "Status": "Success",
  "StatusDetails": "Success",
  "OutputS3BucketName": "my-us-east-2-bucket",
  "OutputS3KeyPrefix": "my-rc-output",
  "MaxConcurrency": "50",
  "MaxErrors": "0",
  "TargetCount": 2,
  "CompletedCount": 2,
  "ErrorCount": 0,
  "DeliveryTimedOutCount": 0,
  "ServiceRole": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
  "NotificationConfig": {
    "NotificationArn": "arn:aws:sns:us-east-2:111222333444:my-us-east-2-
notification-arn",
    "NotificationEvents": [
      "All"
    ],
    "NotificationType": "Invocation"
  },
  "CloudWatchOutputConfig": {
    "CloudWatchLogGroupName": "",
    "CloudWatchOutputEnabled": false
  }
}
{
  "CommandId": "d539b6c3-70e8-4853-80e5-0ce4fEXAMPLE",
  "DocumentName": "AWS-RunPatchBaseline",
  "DocumentVersion": "1",
  "Comment": "",
  "ExpiresAfter": "2020-01-24T12:21:04.350000-08:00",
  "Parameters": {
    "InstallOverrideList": [
      ""
    ],
    "Operation": [
      "Install"
    ],
    "RebootOption": [
      "RebootIfNeeded"
    ]
  },

```



```
        "SnapshotId": [
            ""
        ]
    },
    "InstanceIds": [],
    "Targets": [
        {
            "Key": "InstanceIds",
            "Values": [
                "i-00ec29b21eEXAMPLE",
                "i-09911ddd90EXAMPLE"
            ]
        }
    ],
    "RequestedDateTime": "2020-01-24T11:11:04.350000-08:00",
    "Status": "Success",
    "StatusDetails": "Success",
    "OutputS3BucketName": "my-us-east-2-bucket",
    "OutputS3KeyPrefix": "my-rc-output",
    "MaxConcurrency": "50",
    "MaxErrors": "0",
    "TargetCount": 2,
    "CompletedCount": 2,
    "ErrorCount": 0,
    "DeliveryTimedOutCount": 0,
    "ServiceRole": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "NotificationConfig": {
        "NotificationArn": "arn:aws:sns:us-east-2:111222333444:my-us-east-2-
notification-arn",
        "NotificationEvents": [
            "All"
        ],
        "NotificationType": "Invocation"
    },
    "CloudWatchOutputConfig": {
        "CloudWatchLogGroupName": "",
        "CloudWatchOutputEnabled": false
    }
}
]
```

Para obtener más información, consulte [Running Commands Using Systems Manager Run Command](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [ListCommands](#) la Referencia de AWS CLI comandos.

## list-compliance-items

En el siguiente ejemplo de código, se muestra cómo usar `list-compliance-items`.

### AWS CLI

Enumeración de los elementos de conformidad de una instancia específica

En este ejemplo se enumeran todos los elementos de conformidad de la instancia especificada.

Comando:

```
aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-types "ManagedInstance"
```

Salida:

```
{
  "ComplianceItems": [
    {
      "ComplianceType": "Association",
      "ResourceType": "ManagedInstance",
      "ResourceId": "i-1234567890abcdef0",
      "Id": "8dfe3659-4309-493a-8755-0123456789ab",
      "Title": "",
      "Status": "COMPLIANT",
      "Severity": "UNSPECIFIED",
      "ExecutionSummary": {
        "ExecutionTime": 1550408470.0
      },
      "Details": {
        "DocumentName": "AWS-GatherSoftwareInventory",
        "DocumentVersion": "1"
      }
    },
    {
```

```

    "ComplianceType": "Association",
    "ResourceType": "ManagedInstance",
    "ResourceId": "i-1234567890abcdef0",
    "Id": "e4c2ed6d-516f-41aa-aa2a-0123456789ab",
    "Title": "",
    "Status": "COMPLIANT",
    "Severity": "UNSPECIFIED",
    "ExecutionSummary": {
      "ExecutionTime": 1550508475.0
    },
    "Details": {
      "DocumentName": "AWS-UpdateSSMAgent",
      "DocumentVersion": "1"
    }
  },
  ...
],
"NextToken": "--token string truncated--"
}

```

Enumeración de los elementos de conformidad de una instancia y un ID de asociación específicos

En este ejemplo se enumeran todos los elementos de conformidad de la instancia y el ID de asociación especificados.

Comando:

```

aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-
types "ManagedInstance" --
filters "Key=ComplianceType,Values=Association,Type=EQUAL" "Key=Id,Values=e4c2ed6d-516f-41aa-aa2a-0123456789ab,Type=EQUAL"

```

Enumeración de los elementos de conformidad de una instancia después de una fecha y hora específicas

En este ejemplo se enumeran todos los elementos de conformidad de una instancia después de la fecha y hora especificada.

Comando:

```
aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-  
types "ManagedInstance" --  
filters "Key=ExecutionTime,Values=2019-02-18T16:00:00Z,Type=GREATER_THAN"
```

- Para API obtener más información, consulte [ListComplianceItems](#) la Referencia de AWS CLI comandos.

## list-compliance-summaries

En el siguiente ejemplo de código, se muestra cómo usar `list-compliance-summaries`.

### AWS CLI

Enumeración de los resúmenes de conformidad de todos los tipos de conformidad

En este ejemplo se enumeran los resúmenes de conformidad de todos los tipos de conformidad de su cuenta.

Comando:

```
aws ssm list-compliance-summaries
```

Salida:

```
{  
  "ComplianceSummaryItems": [  
    {  
      "ComplianceType": "Association",  
      "CompliantSummary": {  
        "CompliantCount": 2,  
        "SeveritySummary": {  
          "CriticalCount": 0,  
          "HighCount": 0,  
          "MediumCount": 0,  
          "LowCount": 0,  
          "InformationalCount": 0,  
          "UnspecifiedCount": 2  
        }  
      },  
      "NonCompliantSummary": {  
        "NonCompliantCount": 0,  
        "SeveritySummary": {
```

```

        "CriticalCount": 0,
        "HighCount": 0,
        "MediumCount": 0,
        "LowCount": 0,
        "InformationalCount": 0,
        "UnspecifiedCount": 0
    }
},
{
    "ComplianceType": "Patch",
    "CompliantSummary": {
        "CompliantCount": 1,
        "SeveritySummary": {
            "CriticalCount": 0,
            "HighCount": 0,
            "MediumCount": 0,
            "LowCount": 0,
            "InformationalCount": 0,
            "UnspecifiedCount": 1
        }
    },
    "NonCompliantSummary": {
        "NonCompliantCount": 1,
        "SeveritySummary": {
            "CriticalCount": 1,
            "HighCount": 0,
            "MediumCount": 0,
            "LowCount": 0,
            "InformationalCount": 0,
            "UnspecifiedCount": 0
        }
    }
},
    ...
],
"NextToken": "eyJ0ZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}

```

Enumeración de los resúmenes de conformidad de un tipo de conformidad específico

En este ejemplo se muestra el resumen de conformidad del tipo de conformidad de la revisión.

Comando:

```
aws ssm list-compliance-summaries --  
filters "Key=ComplianceType,Values=Patch,Type=EQUAL"
```

- Para API obtener más información, consulte [ListComplianceSummaries](#) la Referencia de AWS CLI comandos.

## list-document-metadata-history

En el siguiente ejemplo de código, se muestra cómo usar `list-document-metadata-history`.

### AWS CLI

Ejemplo: para ver el historial y el estado de aprobación de una plantilla de cambios

En el siguiente `list-document-metadata-history` ejemplo, se devuelve el historial de aprobaciones de la plantilla de cambios de Change Manager especificada.

```
aws ssm list-document-metadata-history \  
--name MyChangeManageTemplate \  
--metadata DocumentReviews
```

Salida:

```
{  
  "Name": "MyChangeManagerTemplate",  
  "DocumentVersion": "1",  
  "Author": "arn:aws:iam::111222333444:user/JohnDoe",  
  "Metadata": {  
    "ReviewerResponse": [  
      {  
        "CreateTime": "2021-07-30T11:58:28.025000-07:00",  
        "UpdateTime": "2021-07-30T12:01:19.274000-07:00",  
        "ReviewStatus": "APPROVED",  
        "Comment": [  
          {  
            "Type": "COMMENT",  
            "Content": "I approve this template version"  
          }  
        ],  
        "Reviewer": "arn:aws:iam::111222333444:user/ShirleyRodriguez"  
      }  
    ],  
  }  
}
```

```

        "CreateTime": "2021-07-30T11:58:28.025000-07:00",
        "UpdateTime": "2021-07-30T11:58:28.025000-07:00",
        "ReviewStatus": "PENDING"
    }
]
}
}

```

Para obtener más información, consulte [Revisión y aprobación o rechazo de plantillas de cambios](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte la Referencia [ListDocumentMetadataHistory](#) de AWS CLI comandos.

## list-document-versions

En el siguiente ejemplo de código, se muestra cómo usar `list-document-versions`.

### AWS CLI

Enumeración de las versiones de los documentos

En el siguiente ejemplo de `list-document-versions` se enumeran todas las versiones de un documento de Systems Manager.

```
aws ssm list-document-versions \
  --name "Example"
```

Salida:

```

{
  "DocumentVersions": [
    {
      "Name": "Example",
      "DocumentVersion": "1",
      "CreateDate": 1583257938.266,
      "IsDefaultVersion": true,
      "DocumentFormat": "YAML",
      "Status": "Active"
    }
  ]
}

```

Para obtener más información, consulte [Ejecución de comandos mediante una versión de documento específica](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [ListDocumentVersions](#) la Referencia de AWS CLI comandos.

## list-documents

En el siguiente ejemplo de código, se muestra cómo usar `list-documents`.

### AWS CLI

#### Ejemplo 1: enumeración de documentos

En el siguiente ejemplo de `list-documents` se enumeran los documentos propiedad de la cuenta solicitante etiquetados con la etiqueta personalizada.

```
aws ssm list-documents \  
  --filters Key=Owner,Values=Self Key=tag:DocUse,Values=Testing
```

Salida:

```
{  
  "DocumentIdentifiers": [  
    {  
      "Name": "Example",  
      "Owner": "29884EXAMPLE",  
      "PlatformTypes": [  
        "Windows",  
        "Linux"  
      ],  
      "DocumentVersion": "1",  
      "DocumentType": "Automation",  
      "SchemaVersion": "0.3",  
      "DocumentFormat": "YAML",  
      "Tags": [  
        {  
          "Key": "DocUse",  
          "Value": "Testing"  
        }  
      ]  
    }  
  ]  
}
```



```
}
```

Para obtener más información, consulte [Documentos de AWS Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: enumeración de los documentos compartidos

En el siguiente `list-documents` ejemplo se enumeran los documentos compartidos, incluidos los documentos compartidos privados que no son propiedad de AWS.

```
aws ssm list-documents \
  --filters Key=Name,Values=sharedDocNamePrefix Key=Owner,Values=Private
```

Salida:

```
{
  "DocumentIdentifiers": [
    {
      "Name": "Example",
      "Owner": "12345EXAMPLE",
      "PlatformTypes": [
        "Windows",
        "Linux"
      ],
      "DocumentVersion": "1",
      "DocumentType": "Command",
      "SchemaVersion": "0.3",
      "DocumentFormat": "YAML",
      "Tags": []
    }
  ]
}
```

Para obtener más información, consulte [Documentos de AWS Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [ListDocuments](#) la Referencia de AWS CLI comandos.

## **list-inventory-entries**

En el siguiente ejemplo de código, se muestra cómo usar `list-inventory-entries`.

## AWS CLI

Ejemplo 1: visualización de las entradas de tipos de inventario específicos de una instancia

En el siguiente `list-inventory-entries` ejemplo, se enumeran las entradas de inventario del tipo de AWS inventario: Application en una instancia específica.

```
aws ssm list-inventory-entries \  
  --instance-id "i-1234567890abcdef0" \  
  --type-name "AWS:Application"
```

Salida:

```
{  
  "TypeName": "AWS:Application",  
  "InstanceId": "i-1234567890abcdef0",  
  "SchemaVersion": "1.1",  
  "CaptureTime": "2019-02-15T12:17:55Z",  
  "Entries": [  
    {  
      "Architecture": "i386",  
      "Name": "Amazon SSM Agent",  
      "PackageId": "{88a60be2-89a1-4df8-812a-80863c2a2b68}",  
      "Publisher": "Amazon Web Services",  
      "Version": "2.3.274.0"  
    },  
    {  
      "Architecture": "x86_64",  
      "InstalledTime": "2018-05-03T13:42:34Z",  
      "Name": "AmazonCloudWatchAgent",  
      "Publisher": "",  
      "Version": "1.200442.0"  
    }  
  ]  
}
```

Ejemplo 2: visualización de las entradas de inventario personalizadas asignadas a una instancia

En el siguiente ejemplo de `list-inventory-entries` se muestra una entrada de inventario personalizada asignada a una instancia.

```
aws ssm list-inventory-entries \  
  --instance-id "i-1234567890abcdef0" \  
  --type-name "AWS:Application"
```

```
--type-name "Custom:RackInfo"
```

Salida:

```
{
  "TypeName": "Custom:RackInfo",
  "InstanceId": "i-1234567890abcdef0",
  "SchemaVersion": "1.0",
  "CaptureTime": "2021-05-22T10:01:01Z",
  "Entries": [
    {
      "RackLocation": "Bay B/Row C/Rack D/Shelf E"
    }
  ]
}
```

- Para API obtener más información, consulte la Referencia [ListInventoryEntries](#) de AWS CLI comandos.

## list-ops-item-related-items

En el siguiente ejemplo de código, se muestra cómo usar `list-ops-item-related-items`.

AWS CLI

Para enumerar los recursos relacionados de un OpsItem

El siguiente `list-ops-item-related-items` ejemplo muestra los recursos de elementos relacionados de un. OpsItem

```
aws ssm list-ops-item-related-items \
  --ops-item-id "oi-f99f2EXAMPLE"
```

Salida:

```
{
  "Summaries": [
    {
      "OpsItemId": "oi-f99f2EXAMPLE",
      "AssociationId": "e2036148-cccb-490e-ac2a-390e5EXAMPLE",
      "ResourceType": "AWS::SSMIncidents::IncidentRecord",
      "AssociationType": "IsParentOf",
    }
  ]
}
```

```

    "ResourceUri": "arn:aws:ssm-incidents::111122223333:incident-record/
example-response/64bd9b45-1d0e-2622-840d-03a87a1451fa",
    "CreatedBy": {
        "Arn": "arn:aws:sts::111122223333:assumed-role/
AWSServiceRoleForIncidentManager/IncidentResponse"
    },
    "CreatedTime": "2021-08-11T18:47:14.994000+00:00",
    "LastModifiedBy": {
        "Arn": "arn:aws:sts::111122223333:assumed-role/
AWSServiceRoleForIncidentManager/IncidentResponse"
    },
    "LastModifiedTime": "2021-08-11T18:47:14.994000+00:00"
}
]
}

```

Para obtener más información, consulte [Trabajar con incidentes de Incident Manager OpsCenter en](#) la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [ListOpsItemRelatedItems](#) la Referencia de AWS CLI comandos.

## list-resource-compliance-summaries

En el siguiente ejemplo de código, se muestra cómo usar `list-resource-compliance-summaries`.

### AWS CLI

Enumeración de los recuentos resumidos de conformidad para cada recurso

En este ejemplo se enumeran los recuentos resumidos de conformidad para cada recurso.

Comando:

```
aws ssm list-resource-compliance-summaries
```

Salida:

```

{
  "ResourceComplianceSummaryItems": [
    {
      "ComplianceType": "Association",

```

```
"ResourceType": "ManagedInstance",
"ResourceId": "i-1234567890abcdef0",
>Status": "COMPLIANT",
"OverallSeverity": "UNSPECIFIED",
"ExecutionSummary": {
  "ExecutionTime": 1550509273.0
},
"CompliantSummary": {
  "CompliantCount": 2,
  "SeveritySummary": {
    "CriticalCount": 0,
    "HighCount": 0,
    "MediumCount": 0,
    "LowCount": 0,
    "InformationalCount": 0,
    "UnspecifiedCount": 2
  }
},
"NonCompliantSummary": {
  "NonCompliantCount": 0,
  "SeveritySummary": {
    "CriticalCount": 0,
    "HighCount": 0,
    "MediumCount": 0,
    "LowCount": 0,
    "InformationalCount": 0,
    "UnspecifiedCount": 0
  }
},
{
  "ComplianceType": "Patch",
  "ResourceType": "ManagedInstance",
  "ResourceId": "i-9876543210abcdef0",
  "Status": "COMPLIANT",
  "OverallSeverity": "UNSPECIFIED",
  "ExecutionSummary": {
    "ExecutionTime": 1550248550.0,
    "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
    "ExecutionType": "Command"
  },
  "CompliantSummary": {
    "CompliantCount": 397,
    "SeveritySummary": {
```

```

        "CriticalCount": 0,
        "HighCount": 0,
        "MediumCount": 0,
        "LowCount": 0,
        "InformationalCount": 0,
        "UnspecifiedCount": 397
    }
},
"NonCompliantSummary": {
    "NonCompliantCount": 0,
    "SeveritySummary": {
        "CriticalCount": 0,
        "HighCount": 0,
        "MediumCount": 0,
        "LowCount": 0,
        "InformationalCount": 0,
        "UnspecifiedCount": 0
    }
}
}
],
"NextToken": "--token string truncated--"
}

```

Enumeración de los resúmenes de conformidad de un tipo de conformidad específico para cada recurso

En este ejemplo se enumeran los resúmenes de conformidad de cada recurso del tipo de conformidad de la revisión.

Comando:

```
aws ssm list-resource-compliance-summaries --
filters "Key=ComplianceType,Values=Patch,Type=EQUAL"
```

- Para API obtener más información, consulte [ListResourceComplianceSummaries](#) la Referencia de AWS CLI comandos.

## list-resource-data-sync

En el siguiente ejemplo de código, se muestra cómo usar `list-resource-data-sync`.

## AWS CLI

Para enumerar las configuraciones de sincronización de datos de sus recursos

En este ejemplo, se recupera información sobre las configuraciones de sincronización de datos de los recursos.

```
aws ssm list-resource-data-sync
```

Salida:

```
{
  "ResourceDataSyncItems": [
    {
      "SyncName": "MyResourceDataSync",
      "S3Destination": {
        "BucketName": "ssm-resource-data-sync",
        "SyncFormat": "JsonSerDe",
        "Region": "us-east-1"
      },
      "LastSyncTime": 1550261472.003,
      "LastSuccessfulSyncTime": 1550261472.003,
      "LastStatus": "Successful",
      "SyncCreatedTime": 1543235736.72,
      "LastSyncStatusMessage": "The sync was successfully completed"
    }
  ]
}
```

- Para API obtener más información, consulte [ListResourceDataSync](#) la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

## AWS CLI

Enumeración de las etiquetas aplicadas a una línea de base de revisiones

En el siguiente ejemplo de `list-tags-for-resource` se muestran las etiquetas de una línea de base de revisiones.

```
aws ssm list-tags-for-resource \  
  --resource-type "PatchBaseline" \  
  --resource-id "pb-0123456789abcdef0"
```

Salida:

```
{  
  "TagList": [  
    {  
      "Key": "Environment",  
      "Value": "Production"  
    },  
    {  
      "Key": "Region",  
      "Value": "EMEA"  
    }  
  ]  
}
```

Para obtener más información, consulte [Etiquetado de AWS recursos](#) en la Referencia AWS general.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## modify-document-permission

En el siguiente ejemplo de código, se muestra cómo usar `modify-document-permission`.

AWS CLI

Modificación de los permisos de los documentos

En el siguiente ejemplo de `modify-document-permission` se comparte un documento de Systems Manager públicamente.

```
aws ssm modify-document-permission \  
  --name "Example" \  
  --permission-type "Share" \  
  --account-ids-to-add "ALL"
```

Este comando no genera ninguna salida.



Para obtener más información, consulte [Compartir un documento de Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [ModifyDocumentPermission](#) la Referencia de AWS CLI comandos.

## put-compliance-items

En el siguiente ejemplo de código, se muestra cómo usar `put-compliance-items`.

### AWS CLI

Registro de un tipo de conformidad y de los detalles de conformidad en una instancia designada

En este ejemplo se registra el tipo de conformidad `Custom:AVCheck` en la instancia administrada especificada. No se obtienen resultados si el comando se ejecuta correctamente.

Comando:

```
aws ssm put-compliance-items --resource-id "i-1234567890abcdef0" --  
resource-type "ManagedInstance" --compliance-type "Custom:AVCheck"  
--execution-summary "ExecutionTime=2019-02-18T16:00:00Z" --  
items "Id=Version2.0,Title=ScanHost,Severity=CRITICAL,Status=COMPLIANT"
```

- Para API obtener más información, consulte [PutComplianceItems](#) la Referencia de AWS CLI comandos.

## put-inventory

En el siguiente ejemplo de código, se muestra cómo usar `put-inventory`.

### AWS CLI

Asignación de metadatos de cliente a una instancia

En este ejemplo se asigna información de ubicación de bastidores a una instancia. No se obtienen resultados si el comando se ejecuta correctamente.

Comando (Linux):

```
aws ssm put-inventory --instance-id "i-016648b75dd622dab" --items  
'[{"TypeName": "Custom:RackInfo", "SchemaVersion": "1.0", "CaptureTime":
```

```
"2019-01-22T10:01:01Z", "Content": [{"RackLocation": "Bay B/Row C/Rack D/Shelf E"}]"]'
```

Comando (Windows):

```
aws ssm put-inventory --instance-id "i-016648b75dd622dab" --
items "TypeName=Custom:RackInfo,SchemaVersion=1.0,CaptureTime=2019-01-22T10:01:01Z,Content=[
B/Row C/Rack D/Shelf F']"]"
```

- Para API obtener más información, consulte [PutInventory](#) la Referencia de AWS CLI comandos.

## put-parameter

En el siguiente ejemplo de código, se muestra cómo usar `put-parameter`.

### AWS CLI

Ejemplo 1: Cambio del valor de un parámetro

En el siguiente ejemplo de `put-parameter` se cambia el valor del parámetro especificado.

```
aws ssm put-parameter \
  --name "MyStringParameter" \
  --type "String" \
  --value "Vici" \
  --overwrite
```

Salida:

```
{
  "Version": 2,
  "Tier": "Standard"
}
```

Para obtener más información, consulte [Crear un parámetro de Systems Manager \(AWS CLI\)](#), «Administrar niveles de parámetros < <https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html> >» y «Trabajar con políticas de [parámetros](#)» en la Guía del usuario de Systems Manager.AWS

Ejemplo 2: Creación de un parámetro avanzado

En el siguiente ejemplo de `put-parameter` se crea un parámetro avanzado.

```
aws ssm put-parameter \  
  --name "MyAdvancedParameter" \  
  --description "This is an advanced parameter" \  
  --value "Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod  
tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,  
quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat  
[truncated]" \  
  --type "String" \  
  --tier Advanced
```

Salida:

```
{  
  "Version": 1,  
  "Tier": "Advanced"  
}
```

Para obtener más información, consulte [Crear un parámetro de Systems Manager \(AWS CLI\)](#), «Administrar niveles de parámetros < <https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>>`\_`» y «Trabajar con políticas de [parámetros](#)» en la Guía del usuario de Systems Manager.AWS

Ejemplo 3: Cambio de un parámetro estándar a un parámetro avanzado

En el siguiente ejemplo de `put-parameter` se convierte un parámetro estándar existente en un parámetro avanzado.

```
aws ssm put-parameter \  
  --name "MyConvertedParameter" \  
  --value "abc123" \  
  --type "String" \  
  --tier Advanced \  
  --overwrite
```

Salida:

```
{  
  "Version": 2,  
  "Tier": "Advanced"  
}
```

Para obtener más información, consulte [Crear un parámetro de Systems Manager \(AWS CLI\)](#), «Administrar niveles de parámetros < <https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>>`\_\_» y «Trabajar con políticas de [parámetros](#)» en la Guía del usuario de Systems Manager.AWS

Ejemplo 4: Creación de un parámetro con una política adjunta

En el siguiente ejemplo de `put-parameter` se crea un parámetro avanzado con una política de parámetros adjunta.

```
aws ssm put-parameter \
  --name "/Finance/Payroll/q2accesskey" \
  --value "P@sSw)rd" \
  --type "SecureString" \
  --tier Advanced \
  --policies "[{"Type": "Expiration", "Version": "1.0", "Attributes": {"Timestamp": "2020-06-30T00:00:00.000Z"}}, {"Type": "ExpirationNotification", "Version": "1.0", "Attributes": {"Before": "5", "Unit": "Days"}}, {"Type": "NoChangeNotification", "Version": "1.0", "Attributes": {"After": "60", "Unit": "Days"}}]"
```

Salida:

```
{
  "Version": 1,
  "Tier": "Advanced"
}
```

Para obtener más información, consulte [Crear un parámetro de Systems Manager \(AWS CLI\)](#), «Administrar niveles de parámetros < <https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>>`\_\_» y «Trabajar con políticas de [parámetros](#)» en la Guía del usuario de Systems Manager.AWS

Ejemplo 5: Adición de una política a un parámetro existente

En el siguiente ejemplo de `put-parameter` se adjunta una política a un parámetro avanzado existente.

```
aws ssm put-parameter \
  --name "/Finance/Payroll/q2accesskey" \
  --value "N3wP@sSw)rd" \
  --type "SecureString" \
```

```
--tier Advanced \
--policies "[{"Type":"Expiration","Version":"1.0","Attributes":
{"Timestamp":"2020-06-30T00:00:00.000Z"}}, {"Type":"ExpirationNotification",
"Version":"1.0","Attributes":{"Before":"5","Unit":"Days"}}, {"Type":"
NoChangeNotification","Version":"1.0","Attributes":{"After":"60","Unit
":"Days"}}]"
--overwrite
```

Salida:

```
{
  "Version": 2,
  "Tier": "Advanced"
}
```

Para obtener más información, consulte [Crear un parámetro de Systems Manager \(AWS CLI\)](#), «Administrar niveles de parámetros < <https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>» y «Trabajar con políticas de [parámetros](#)» en la Guía del usuario de Systems Manager.AWS

- Para obtener API más información, [PutParameter](#) consulte AWS CLI la Referencia de comandos.

## register-default-patch-baseline

En el siguiente ejemplo de código, se muestra cómo usar `register-default-patch-baseline`.

AWS CLI

Configuración de la línea de base de revisiones predeterminada

En el siguiente ejemplo de `register-default-patch-baseline` se registra la línea de base de revisiones personalizada especificada como la línea de base de revisiones predeterminada para el tipo de sistema operativo que admite.

```
aws ssm register-default-patch-baseline \
  --baseline-id "pb-abc123cf9bEXAMPLE"
```

Salida:

```
{
```

```
"BaselineId": "pb-abc123cf9bEXAMPLE"
}
```

El siguiente `register-default-patch-baseline` ejemplo registra la línea base del parche predeterminada proporcionada AWS por CentOS como la línea base del parche predeterminada.

```
aws ssm register-default-patch-baseline \
  --baseline-id "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
  pb-0574b43a65ea646ed"
```

Salida:

```
{
  "BaselineId": "pb-abc123cf9bEXAMPLE"
}
```

Para obtener más información, consulte [Acerca de las líneas de base de revisiones personalizadas y predefinidas](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [RegisterDefaultPatchBaseline](#) la Referencia de AWS CLI comandos.

## register-patch-baseline-for-patch-group

En el siguiente ejemplo de código, se muestra cómo usar `register-patch-baseline-for-patch-group`.

AWS CLI

Registro de una línea de base de revisiones de un grupo de revisiones

En el siguiente ejemplo de `register-patch-baseline-for-patch-group` se registra una línea de base de revisiones para un grupo de revisiones.

```
aws ssm register-patch-baseline-for-patch-group \
  --baseline-id "pb-045f10b4f382baeda" \
  --patch-group "Production"
```

Salida:

```
{
```

```
"BaselineId": "pb-045f10b4f382baeda",  
"PatchGroup": "Production"  
}
```

Para obtener más información, consulte [Crear un grupo de parches < https:// docs.aws.amazon.com/systems-manager/ latest/userguide/ sysman-patch-group-tagging .html>\\_\\_](https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html) y [Agregar un grupo de parches a una línea base de parches en la Guía del](#) usuario de Systems Manager.AWS

- Para obtener más información, consulte la Referencia de comandos. API [RegisterPatchBaselineForPatchGroup](#) AWS CLI

## register-target-with-maintenance-window

En el siguiente ejemplo de código, se muestra cómo usar `register-target-with-maintenance-window`.

### AWS CLI

Ejemplo 1: registro de un único destino con un periodo de mantenimiento

En el siguiente ejemplo de `register-target-with-maintenance-window` se registra una instancia con un periodo de mantenimiento.

```
aws ssm register-target-with-maintenance-window \  
  --window-id "mw-ab12cd34ef56gh78" \  
  --target "Key=InstanceIds,Values=i-0000293ffd8c57862" \  
  --owner-information "Single instance" \  
  --resource-type "INSTANCE"
```

Salida:

```
{  
  "WindowTargetId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"  
}
```

Ejemplo 2: Para registrar varios objetivos con una ventana de mantenimiento mediante una instancia IDs

En el siguiente `register-target-with-maintenance-window` ejemplo, se registran dos instancias con una ventana de mantenimiento especificando su instanciaIDs.

```
aws ssm register-target-with-maintenance-window \  
  --window-id "mw-ab12cd34ef56gh78" \  
  --target "Key=InstanceIds,Values=i-0000293ffd8c57862,i-0cb2b964d3e14fd9f" \  
  --owner-information "Two instances in a list" \  
  --resource-type "INSTANCE"
```

Salida:

```
{  
  "WindowTargetId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"  
}
```

Ejemplo 3: registro de destinos con un periodo de mantenimiento mediante etiquetas de recursos

En el siguiente ejemplo de `register-target-with-maintenance-window` se registran las instancias con un periodo de mantenimiento especificando las etiquetas de recursos que se han aplicado a las instancias.

```
aws ssm register-target-with-maintenance-window \  
  --window-id "mw-06cf17cbefcb4bf4f" \  
  --targets "Key=tag:Environment,Values=Prod" "Key=Role,Values=Web" \  
  --owner-information "Production Web Servers" \  
  --resource-type "INSTANCE"
```

Salida:

```
{  
  "WindowTargetId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"  
}
```

Ejemplo 4: registro de destinos mediante un grupo de claves de etiquetas

En el siguiente ejemplo de `register-target-with-maintenance-window` se registran instancias que tienen una o más claves asignadas, independientemente de los valores de la clave.

```
aws ssm register-target-with-maintenance-window \  
  --window-id "mw-0c50858d01EXAMPLE" \  
  --resource-type "INSTANCE" \  
  --target "Key=tag:Environment,Values=Prod" "Key=Role,Values=Web" \  
  --owner-information "Production Web Servers" \  
  --resource-type "INSTANCE"
```



```
--target "Key=tag-key, Values=Name, Instance-Type, CostCenter"
```

Salida:

```
{
  "WindowTargetId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Ejemplo 5: registro de destinos con un nombre de grupo de recursos

En el siguiente ejemplo de `register-target-with-maintenance-window` se registra un grupo de recursos especificado, independientemente del tipo de recursos que contiene.

```
aws ssm register-target-with-maintenance-window \
  --window-id "mw-0c50858d01EXAMPLE" \
  --resource-type "RESOURCE_GROUP" \
  --target "Key=resource-groups:Name, Values=MyResourceGroup"
```

Salida:

```
{
  "WindowTargetId": "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Para obtener más información, consulte [Registrar una instancia de destino con la ventana de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [RegisterTargetWithMaintenanceWindow](#) la Referencia de AWS CLI comandos.

## **register-task-with-maintenance-window**

En el siguiente ejemplo de código, se muestra cómo usar `register-task-with-maintenance-window`.

AWS CLI

Ejemplo 1: registro de una tarea de Automatización con un periodo de mantenimiento

En el siguiente ejemplo de `register-task-with-maintenance-window` se registra una tarea de Automatización con un periodo de mantenimiento destinado a una instancia.

```
aws ssm register-task-with-maintenance-window \
  --window-id "mw-082dcd7649EXAMPLE" \
  --targets Key=InstanceIds,Values=i-1234520122EXAMPLE \
  --task-arn AWS-RestartEC2Instance \
  --service-role-arn arn:aws:iam::111222333444:role/SSM --task-type AUTOMATION \
  --task-invocation-parameters "{\"Automation\":{\"DocumentVersion\":\"\$LATEST\"},\"Parameters\":{\"InstanceId\":[\"{{RESOURCE_ID}}\"]}}\" \
  --priority 0 \
  --max-concurrency 1 \
  --max-errors 1 \
  --name "AutomationExample" \
  --description "Restarting EC2 Instance for maintenance"
```

Salida:

```
{
  "WindowTaskId": "11144444-5555-6666-7777-88888888"
}
```

Para obtener más información, consulte [Registrar una tarea con la ventana de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: registro de una tarea de Lambda con un periodo de mantenimiento

En el siguiente ejemplo de `register-task-with-maintenance-window` se registra una tarea de Lambda con un periodo de mantenimiento destinado a una instancia.

```
aws ssm register-task-with-maintenance-window \
  --window-id "mw-082dcd7649dee04e4" \
  --targets Key=InstanceIds,Values=i-12344d305eEXAMPLE \
  --task-arn arn:aws:lambda:us-east-1:111222333444:function:SSMTestLAMBDA \
  --service-role-arn arn:aws:iam::111222333444:role/SSM \
  --task-type LAMBDA \
  --task-invocation-parameters '{"Lambda":{"Payload":{"InstanceId":\"{{RESOURCE_ID}}\", \"targetType\": \"{{TARGET_TYPE}}\"}, \"Qualifier\": \"$LATEST\"}}' \
  --priority 0 \
  --max-concurrency 10 \
  --max-errors 5 \
  --name "Lambda_Example" \
  --description "My Lambda Example"
```

Salida:

```
{
  "WindowTaskId": "22244444-5555-6666-7777-88888888"
}
```

Para obtener más información, consulte [Registrar una tarea con la ventana de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 3: registro de una tarea de ejecución de un comando con un periodo de mantenimiento

En el siguiente ejemplo de `register-task-with-maintenance-window` se registra una tarea de ejecución de un comando con un periodo de mantenimiento destinado a una instancia.

```
aws ssm register-task-with-maintenance-window \
  --window-id "mw-082dcd7649dee04e4" \
  --targets "Key=InstanceIds,Values=i-12344d305eEXAMPLE" \
  --service-role-arn "arn:aws:iam::111222333444:role/SSM" \
  --task-type "RUN_COMMAND" \
  --name "SSMInstallPowerShellModule" \
  --task-arn "AWS-InstallPowerShellModule" \
  --task-invocation-parameters "{\"RunCommand\":{\"Comment\":\"\",
  \"OutputS3BucketName\": \"runcommandlogs\", \"Parameters\": {\"commands\": [\"Get-
  Module -ListAvailable\"], \"executionTimeout\": [\"3600\"], \"source\": [\"https://
  /gallery.technet.microsoft.com/EZ0out-33ae0fb7/file/110351/1/EZ0out.zip\"],
  \"workingDirectory\": [\"\\\\\\\\\\\\\\\\\"]}, \"TimeoutSeconds\": 600}}" \
  --max-concurrency 1 \
  --max-errors 1 \
  --priority 10
```

Salida:

```
{
  "WindowTaskId": "33344444-5555-6666-7777-88888888"
}
```

Para obtener más información, consulte [Registrar una tarea con la ventana de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 4: registro de una tarea de Step Functions con un periodo de mantenimiento

En el siguiente ejemplo de `register-task-with-maintenance-window` se registra una tarea de Step Functions con un periodo de mantenimiento destinado a una instancia.

```
aws ssm register-task-with-maintenance-window \
  --window-id "mw-1234d787d6EXAMPLE" \
  --targets Key=WindowTargetIds,Values=12347414-69c3-49f8-95b8-ed2dcEXAMPLE \
  --task-arn arn:aws:states:us-
east-1:111222333444:stateMachine:SSMTestStateMachine \
  --service-role-arn arn:aws:iam::111222333444:role/MaintenanceWindows \
  --task-type STEP_FUNCTIONS \
  --task-invocation-parameters '{"StepFunctions":{"Input":{"InstanceId\":"
\ "{{RESOURCE_ID}} \"}"}}' \
  --priority 0 \
  --max-concurrency 10 \
  --max-errors 5 \
  --name "Step_Functions_Example" \
  --description "My Step Functions Example"
```

Salida:

```
{
  "WindowTaskId": "444444444-5555-6666-7777-88888888"
}
```

Para obtener más información, consulte [Registrar una tarea con la ventana de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 5: registro de una tarea mediante un ID de destino de un periodo de mantenimiento

En el siguiente ejemplo de `register-task-with-maintenance-window` se registra una tarea mediante un ID de destino de un periodo de mantenimiento. El ID de destino del periodo de mantenimiento estaba en el resultado del comando `aws ssm register-target-with-maintenance-window`. También puede recuperarlo del resultado del comando `aws ssm describe-maintenance-window-targets`.

```
aws ssm register-task-with-maintenance-window \
  --targets "Key=WindowTargetIds,Values=350d44e6-28cc-44e2-951f-4b2c9EXAMPLE" \
  --task-arn "AWS-RunShellScript" \
  --service-role-arn "arn:aws:iam::111222333444:role/MaintenanceWindowsRole" \
  --window-id "mw-ab12cd34eEXAMPLE" \
  --task-type "RUN_COMMAND" \
  --task-parameters '{"commands\":"Values\":[\df\"]}' \
  --max-concurrency 1 \
  --max-errors 1 \
```

```
--priority 10
```

Salida:

```
{
  "WindowTaskId": "33344444-5555-6666-7777-88888888"
}
```

Para obtener más información, consulte [Registrar una tarea con la ventana de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [RegisterTaskWithMaintenanceWindow](#) la Referencia de AWS CLI comandos.

## **remove-tags-from-resource**

En el siguiente ejemplo de código, se muestra cómo usar `remove-tags-from-resource`.

AWS CLI

Eliminación de una etiqueta de una línea de base de revisiones

En el siguiente ejemplo de `remove-tags-from-resource` se eliminan las etiquetas de una línea de base de revisiones.

```
aws ssm remove-tags-from-resource \
  --resource-type "PatchBaseline" \
  --resource-id "pb-0123456789abcdef0" \
  --tag-keys "Region"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Etiquetado de AWS recursos](#) en la Referencia AWS general.

- Para API obtener más información, consulte [RemoveTagsFromResource](#) la Referencia de AWS CLI comandos.

## **reset-service-setting**

En el siguiente ejemplo de código, se muestra cómo usar `reset-service-setting`.

## AWS CLI

Para restablecer la configuración del servicio para el rendimiento del almacén de parámetros

En el siguiente `reset-service-setting` ejemplo, se restablece la configuración del servicio para el rendimiento del almacén de parámetros en la región especificada para que deje de utilizar el rendimiento incrementado.

```
aws ssm reset-service-setting \
  --setting-id arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-
  store/high-throughput-enabled
```

Salida:

```
{
  "ServiceSetting": {
    "SettingId": "/ssm/parameter-store/high-throughput-enabled",
    "SettingValue": "false",
    "LastModifiedDate": 1555532818.578,
    "LastModifiedUser": "System",
    "ARN": "arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-
    store/high-throughput-enabled",
    "Status": "Default"
  }
}
```

Para obtener más información, consulte [Aumentar el rendimiento del almacén de parámetros](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [ResetServiceSetting](#) la Referencia de AWS CLI comandos.

## **resume-session**

En el siguiente ejemplo de código, se muestra cómo usar `resume-session`.

## AWS CLI

Para reanudar una sesión del administrador de sesiones

En este `resume-session` ejemplo, se reanuda una sesión del administrador de sesiones con una instancia después de haberla desconectado. Tenga en cuenta que este comando interactivo

requiere que el complemento Session Manager esté instalado en la máquina cliente que realiza la llamada.

```
aws ssm resume-session \  
  --session-id Mary-Major-07a16060613c408b5
```

Salida:

```
{  
  "SessionId": "Mary-Major-07a16060613c408b5",  
  "TokenValue":  
    "AAEAAVbTGsa0nyvcUoNGqifbv5r/8lgxuQljCuY8qVcv0noBAAAAAFxtd3jIXAFUUXGTJ7zF/  
    AWJJPwDvi0lF5p3d1AgrqVIV06IEXhkHLz0/1gXKRKEME71E6TL0p1LDJAMZ  
    +kREejkZu4c5AxMkrQjMF+gtHP1bYJKTwtHQd1wjulPLex08SH17g5R/  
    wekrj6WsDUpnEegFBfGftpAIz2GXQVfTJXKfkc5qepQ11C11D0IT2doz0qXgHwfQHfAKLErM5dWDZqKwyT1Z3iw7unQd  
    +ihfGa6MEJJ97Jmat/a2TspEn0jNn9Mvu5iwXIW2yCvWZrGUj+/  
    QI5Xr7s1XJBEEnSKR54o4fN0GV9RWl0RZsZm1mki0JJtiwwgZ",  
  "StreamUrl": "wss://ssmmessages.us-east-2.amazonaws.com/v1/data-channel/Mary-  
    Major-07a16060613c408b5?role=publish_subscribe"  
}
```

Para obtener más información, consulte [Instalación del complemento Session Manager AWS CLI en la Guía del usuario de AWS Systems Manager](#).

- Para API obtener más información, consulte [ResumeSession](#) la Referencia de AWS CLI comandos.

## send-automation-signal

En el siguiente ejemplo de código, se muestra cómo usar `send-automation-signal`.

AWS CLI

Para enviar una señal a una ejecución de automatización

El siguiente `send-automation-signal` ejemplo envía una señal de aprobación a una ejecución de automatización.

```
aws ssm send-automation-signal \  
  --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE \  
  --signal-type "Approve"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Ejecución de un flujo de trabajo de automatización con aprobadores](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [SendAutomationSignal](#) la Referencia de AWS CLI comandos.

## send-command

En el siguiente ejemplo de código, se muestra cómo usar send-command.

### AWS CLI

Ejemplo 1: ejecución de un comando en una o más instancias remotas

En el siguiente ejemplo de send-command se ejecuta un comando echo en una instancia de destino.

```
aws ssm send-command \  
  --document-name "AWS-RunShellScript" \  
  --parameters 'commands=["echo HelloWorld"]' \  
  --targets "Key=instanceids,Values=i-1234567890abcdef0" \  
  --comment "echo HelloWorld"
```

Salida:

```
{  
  "Command": {  
    "CommandId": "92853adf-ba41-4cd6-9a88-142d1EXAMPLE",  
    "DocumentName": "AWS-RunShellScript",  
    "DocumentVersion": "",  
    "Comment": "echo HelloWorld",  
    "ExpiresAfter": 1550181014.717,  
    "Parameters": {  
      "commands": [  
        "echo HelloWorld"  
      ]  
    },  
    "InstanceIds": [  
      "i-0f00f008a2dcbef2"  
    ],  
    "Targets": [],
```



```

    "RequestedDateTime": 1550173814.717,
    "Status": "Pending",
    "StatusDetails": "Pending",
    "OutputS3BucketName": "",
    "OutputS3KeyPrefix": "",
    "MaxConcurrency": "50",
    "MaxErrors": "0",
    "TargetCount": 1,
    "CompletedCount": 0,
    "ErrorCount": 0,
    "DeliveryTimedOutCount": 0,
    "ServiceRole": "",
    "NotificationConfig": {
      "NotificationArn": "",
      "NotificationEvents": [],
      "NotificationType": ""
    },
    "CloudWatchOutputConfig": {
      "CloudWatchLogGroupName": "",
      "CloudWatchOutputEnabled": false
    }
  }
}

```

Para obtener más información, consulte [Running Commands Using Systems Manager Run Command](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: obtención de información sobre la IP de una instancia

En el siguiente ejemplo de `send-command` se obtiene información sobre la IP de una instancia.

```

aws ssm send-command \
  --instance-ids "i-1234567890abcdef0" \
  --document-name "AWS-RunShellScript" \
  --comment "IP config" \
  --parameters "commands=ifconfig"

```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte [Running Commands Using Systems Manager Run Command](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 3: ejecución de un comando en instancias con etiquetas específicas

En el siguiente send-command ejemplo, se ejecuta un comando en instancias que tienen la clave de etiqueta ENV "» y el valor «Dev».

```
aws ssm send-command \  
  --targets "Key=tag:ENV,Values=Dev" \  
  --document-name "AWS-RunShellScript" \  
  --parameters "commands=ifconfig"
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte [Running Commands Using Systems Manager Run Command](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 4: ejecutar un comando que envíe SNS notificaciones

En el siguiente send-command ejemplo, se ejecuta un comando que envía SNS notificaciones para todos los eventos de notificación y el tipo de Command notificación.

```
aws ssm send-command \  
  --instance-ids "i-1234567890abcdef0" \  
  --document-name "AWS-RunShellScript" \  
  --comment "IP config" \  
  --parameters "commands=ifconfig" \  
  --service-role-arn "arn:aws:iam::123456789012:role/SNS_Role" \  
  --notification-config "NotificationArn=arn:aws:sns:us-  
east-1:123456789012:SNSTopicName,NotificationEvents=All,NotificationType=Command"
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte [Running Commands Using Systems Manager Run Command](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 5: Para ejecutar un comando que se envíe a S3 y CloudWatch

El siguiente send-command ejemplo ejecuta un comando que envía los detalles del comando a un bucket de S3 y a un grupo de CloudWatch registros.

```
aws ssm send-command \  
  --instance-ids "i-1234567890abcdef0" \  
  --document-name "AWS-RunShellScript" \  
  --comment "IP config" \  
  --parameters "commands=ifconfig" \  
  --output-s3-bucket-name "s3-bucket-name" \  
  --output-cloudwatch-log-group-name "cloudwatch-log-group-name"
```

```
--output-s3-key-prefix "runcommand" \  
--cloud-watch-output-  
config "CloudWatchOutputEnabled=true,CloudWatchLogGroupName=CWLGroupName"
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte [Running Commands Using Systems Manager Run Command](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 6: ejecución de comandos en varias instancias con etiquetas diferentes

En el siguiente ejemplo de `send-command` se ejecuta un comando en instancias con dos claves y valores de etiqueta diferentes.

```
aws ssm send-command \  
--document-name "AWS-RunPowerShellScript" \  
--parameters commands=["echo helloWorld"] \  
--targets Key=tag:Env,Values=Dev Key=tag:Role,Values=WebServers
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte [Running Commands Using Systems Manager Run Command](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 7: definición de varias instancias como destino con la misma clave de etiqueta

En el siguiente ejemplo de `send-command` se ejecuta un comando en instancias que tienen la misma clave de etiqueta, pero con valores diferentes.

```
aws ssm send-command \  
--document-name "AWS-RunPowerShellScript" \  
--parameters commands=["echo helloWorld"] \  
--targets Key=tag:Env,Values=Dev,Test
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte [Running Commands Using Systems Manager Run Command](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 8: ejecución de un comando que usa un documento compartido

En el siguiente ejemplo de `send-command` se ejecuta un documento compartido en una instancia de destino.

```
aws ssm send-command \  
  --document-name "arn:aws:ssm:us-east-1:123456789012:document/ExampleDocument" \  
  --targets "Key=instanceids,Values=i-1234567890abcdef0"
```

Consulte el ejemplo 1 para ver una salida de muestra.

Para obtener más información, consulte [Uso de SSM documentos compartidos](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [SendCommand](#) la Referencia de AWS CLI comandos.

## start-associations-once

En el siguiente ejemplo de código, se muestra cómo usar `start-associations-once`.

### AWS CLI

Para ejecutar una asociación de forma inmediata y solo una vez

En el siguiente `start-associations-once` ejemplo, se ejecuta la asociación especificada inmediatamente y solo una vez. No se obtienen resultados si el comando se ejecuta correctamente.

```
aws ssm start-associations-once \  
  --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Visualización de los historiales de asociación](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [StartAssociationsOnce](#) la Referencia de AWS CLI comandos.

## start-automation-execution

En el siguiente ejemplo de código, se muestra cómo usar `start-automation-execution`.

### AWS CLI

Ejemplo 1: ejecución de un documento de automatización

En el siguiente ejemplo de `start-automation-execution` se ejecuta un documento de Automatización.

```
aws ssm start-automation-execution \  
  --document-name "AWS-UpdateLinuxAmi" \  
  --parameters "AutomationAssumeRole=arn:aws:iam::123456789012:role/  
SSMAutomationRole,SourceAmiId=ami-EXAMPLE,IamInstanceProfileName=EC2InstanceRole"
```

Salida:

```
{  
  "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"  
}
```

Para obtener más información, consulte [Ejecución manual de un flujo de trabajo de Automatización](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: ejecución de un documento de automatización compartido

En el siguiente ejemplo de `start-automation-execution` se ejecuta un documento de Automatización compartido.

```
aws ssm start-automation-execution \  
  --document-name "arn:aws:ssm:us-east-1:123456789012:document/ExampleDocument"
```

Salida:

```
{  
  "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"  
}
```

Para obtener más información, consulte [Uso de SSM documentos compartidos](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [StartAutomationExecution](#) la Referencia de AWS CLI comandos.

## **start-change-request-execution**

En el siguiente ejemplo de código, se muestra cómo usar `start-change-request-execution`.

## AWS CLI

Ejemplo 1: Para iniciar una solicitud de cambio

En el siguiente `start-change-request-execution` ejemplo, se inicia una solicitud de cambio con un mínimo de opciones especificadas.

```
aws ssm start-change-request-execution \  
  --change-request-name MyChangeRequest \  
  --document-name AWS-HelloWorldChangeTemplate \  
  --runbooks '[{"DocumentName": "AWS-HelloWorld", "Parameters":  
  {"AutomationAssumeRole": ["arn:aws:iam:us-east-2:1112223233444:role/  
  MyChangeManagerAssumeRole"]}'] \  
  --parameters  
  Approver="JohnDoe", ApproverType="IamUser", ApproverSnsTopicArn="arn:aws:sns:us-  
  east-2:1112223233444:MyNotificationTopic"
```

Salida:

```
{  
  "AutomationExecutionId": "9d32a4fc-f944-11e6-4105-0a1b2EXAMPLE"  
}
```

Ejemplo 2: Para iniciar una solicitud de cambio mediante un JSON archivo externo

En el siguiente `start-automation-execution` ejemplo, se inicia una solicitud de cambio con varias opciones especificadas en un JSON archivo.

```
aws ssm start-change-request-execution \  
  --cli-input-json file://MyChangeRequest.json
```

Contenidos de `MyChangeRequest.json`:

```
{  
  "ChangeRequestName": "MyChangeRequest",  
  "DocumentName": "AWS-HelloWorldChangeTemplate",  
  "DocumentVersion": "$DEFAULT",  
  "ScheduledTime": "2021-12-30T03:00:00",  
  "ScheduledEndTime": "2021-12-30T03:05:00",  
  "Tags": [  
    {  
      "Key": "Purpose",
```

```

        "Value": "Testing"
    }
],
"Parameters": {
    "Approver": [
        "JohnDoe"
    ],
    "ApproverType": [
        "IamUser"
    ],
    "ApproverSnsTopicArn": [
        "arn:aws:sns:us-east-2:111222333444;:MyNotificationTopic"
    ]
},
"Runbooks": [
    {
        "DocumentName": "AWS-HelloWorld",
        "DocumentVersion": "1",
        "MaxConcurrency": "1",
        "MaxErrors": "1",
        "Parameters": {
            "AutomationAssumeRole": [
                "arn:aws:iam::111222333444:role/MyChangeManagerAssumeRole"
            ]
        }
    }
],
"ChangeDetails": "### Document Name: HelloWorldChangeTemplate\n\n## What does this document do?\n\nThis change template demonstrates the feature set available for creating change templates for Change Manager. This template starts a Runbook workflow for the Automation document called AWS-HelloWorld.\n\n## Input Parameters\n\n* ApproverSnsTopicArn: (Required) Amazon Simple Notification Service ARN for approvers.\n* Approver: (Required) The name of the approver to send this request to.\n* ApproverType: (Required) The type of reviewer.\n  * Allowed Values: IamUser, IamGroup, IamRole, SSOGroup, SSOUser\n\n## Output Parameters\n\nThis document has no outputs \n"
}

```

**Salida:**

```

{
  "AutomationExecutionId": "9d32a4fc-f944-11e6-4105-0a1b2EXAMPLE"
}

```

Para obtener más información, consulte [Creación de solicitudes de cambio](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [StartChangeRequestExecution](#) la Referencia de AWS CLI comandos.

## start-session

En el siguiente ejemplo de código, se muestra cómo usar `start-session`.

### AWS CLI

Ejemplo 1: para iniciar una sesión de Session Manager

Este ejemplo de `start-session` establece una conexión a una instancia para una sesión de Session Manager. Tenga en cuenta que este comando interactivo requiere que el complemento Session Manager esté instalado en la máquina cliente que realiza la llamada.

```
aws ssm start-session \  
  --target "i-1234567890abcdef0"
```

Salida:

```
Starting session with SessionId: Jane-Roe-07a16060613c408b5
```

Ejemplo 2: Para iniciar una sesión del administrador de sesiones mediante SSH

Este `start-session` ejemplo establece una conexión con una instancia para una sesión del administrador de sesiones mediante SSH. Tenga en cuenta que este comando interactivo requiere que el complemento Session Manager esté instalado en la máquina cliente que realiza la llamada y que el comando usa el usuario predeterminado de la instancia, como en el caso `ec2-user` de EC2 las instancias de Linux.

```
ssh -i /path/my-key-pair.pem ec2-user@i-02573cafcfEXAMPLE
```

Salida:

```
Starting session with SessionId: ec2-user-07a16060613c408b5
```



Para obtener más información, consulte [Iniciar una sesión](#) e [instalar el complemento Session Manager AWS CLI en la](#) Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [StartSession](#) la Referencia de AWS CLI comandos.

## **stop-automation-execution**

En el siguiente ejemplo de código, se muestra cómo usar `stop-automation-execution`.

### AWS CLI

Detención de una ejecución de automatización

En el siguiente ejemplo de `stop-automation-execution` se detiene un documento de Automatización.

```
aws ssm stop-automation-execution
  --automation-execution-id "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Ejecución manual de un flujo de trabajo de Automatización](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [StopAutomationExecution](#) la Referencia de AWS CLI comandos.

## **terminate-session**

En el siguiente ejemplo de código, se muestra cómo usar `terminate-session`.

### AWS CLI

Para finalizar una sesión del administrador de sesiones

Este `terminate-session` ejemplo finaliza permanentemente una sesión creada por el usuario «Shirley-Rodriguez» y cierra la conexión de datos entre el cliente del administrador de sesiones y el SSM agente de la instancia.

```
aws ssm terminate-session \
  --session-id "Shirley-Rodriguez-07a16060613c408b5"
```

Salida:

```
{
  "SessionId": "Shirley-Rodriguez-07a16060613c408b5"
}
```

Para obtener más información, consulte [Finalizar una sesión](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [TerminateSession](#) la Referencia de AWS CLI comandos.

## **unlabel-parameter-version**

En el siguiente ejemplo de código, se muestra cómo usar `unlabel-parameter-version`.

AWS CLI

Para eliminar etiquetas de parámetros

En el siguiente `unlabel-parameter-version` ejemplo, se eliminan las etiquetas especificadas de la versión del parámetro en cuestión.

```
aws ssm unlabel-parameter-version \
  --name "parameterName" \
  --parameter-version "version" \
  --labels "label_1" "label_2" "label_3"
```

Salida:

```
{
  "RemovedLabels": [
    "label_1"
    "label_2"
    "label_3"
  ],
  "InvalidLabels": []
}
```

Para obtener más información, consulte [Eliminar etiquetas de parámetros \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [UnlabelParameterVersion](#) la Referencia de AWS CLI comandos.

## update-association-status

En el siguiente ejemplo de código, se muestra cómo usar `update-association-status`.

### AWS CLI

Actualización del estado de la asociación

En el siguiente ejemplo de `update-association-status` se actualiza el estado de la asociación entre una instancia y un documento.

```
aws ssm update-association-status \
  --name "AWS-UpdateSSMAgent" \
  --instance-id "i-1234567890abcdef0" \
  --association-
status "Date=1424421071.939,Name=Pending,Message=temp_status_change,AdditionalInfo=Additional
Config-Needed"
```

Salida:

```
{
  "AssociationDescription": {
    "Name": "AWS-UpdateSSMAgent",
    "InstanceId": "i-1234567890abcdef0",
    "AssociationVersion": "1",
    "Date": 1550507529.604,
    "LastUpdateAssociationDate": 1550507806.974,
    "Status": {
      "Date": 1424421071.0,
      "Name": "Pending",
      "Message": "temp_status_change",
      "AdditionalInfo": "Additional-Config-Needed"
    },
    "Overview": {
      "Status": "Success",
      "AssociationStatusAggregatedCount": {
        "Success": 1
      }
    }
  },
}
```

```

    "DocumentVersion": "$DEFAULT",
    "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
    "Targets": [
      {
        "Key": "InstanceIds",
        "Values": [
          "i-1234567890abcdef0"
        ]
      }
    ],
    "LastExecutionDate": 1550507808.0,
    "LastSuccessfulExecutionDate": 1550507808.0
  }
}

```

Para obtener más información, consulte [Trabajo con asociaciones en Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [UpdateAssociationStatus](#) la Referencia de AWS CLI comandos.

## update-association

En el siguiente ejemplo de código, se muestra cómo usar `update-association`.

### AWS CLI

Ejemplo 1: actualización de una asociación de documentos

En el siguiente ejemplo de `update-association` se actualiza una asociación con una nueva versión del documento.

```

aws ssm update-association \
  --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
  --document-version "\$LATEST"

```

Salida:

```

{
  "AssociationDescription": {
    "Name": "AWS-UpdateSSMAgent",
    "AssociationVersion": "2",

```

```

    "Date": 1550508093.293,
    "LastUpdateAssociationDate": 1550508106.596,
    "Overview": {
      "Status": "Pending",
      "DetailedStatus": "Creating"
    },
    "DocumentVersion": "$LATEST",
    "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
    "Targets": [
      {
        "Key": "tag:Name",
        "Values": [
          "Linux"
        ]
      }
    ],
    "LastExecutionDate": 1550508094.879,
    "LastSuccessfulExecutionDate": 1550508094.879
  }
}

```

Para obtener más información, consulte [Edición y creación de una nueva versión de una asociación](#) en la Guía del usuario de AWS Systems Manager.

Ejemplo 2: actualización de la expresión de programación de una asociación

En el siguiente ejemplo de `update-association` se actualiza la expresión de programación de la asociación especificada.

```

aws ssm update-association \
  --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
  --schedule-expression "cron(0 0 0/4 1/1 * ? *)"

```

Salida:

```

{
  "AssociationDescription": {
    "Name": "AWS-HelloWorld",
    "AssociationVersion": "2",
    "Date": "2021-02-08T13:54:19.203000-08:00",
    "LastUpdateAssociationDate": "2021-06-29T11:51:07.933000-07:00",
    "Overview": {
      "Status": "Pending",

```

```

        "DetailedStatus": "Creating"
    },
    "DocumentVersion": "$DEFAULT",
    "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
    "Targets": [
        {
            "Key": "aws:NoOpAutomationTag",
            "Values": [
                "AWS-NoOpAutomationTarget-Value"
            ]
        }
    ],
    "ScheduleExpression": "cron(0 0 0/4 1/1 * ? *)",
    "LastExecutionDate": "2021-06-26T19:00:48.110000-07:00",
    "ApplyOnlyAtCronInterval": false
}
}

```

Para obtener más información, consulte [Edición y creación de una nueva versión de una asociación](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [UpdateAssociation](#) la Referencia de AWS CLI comandos.

## update-document-default-version

En el siguiente ejemplo de código, se muestra cómo usar `update-document-default-version`.

AWS CLI

Actualización de la versión predeterminada de un documento

En el siguiente ejemplo de `update-document-default-version` se actualiza la versión predeterminada de un documento de Systems Manager.

```

aws ssm update-document-default-version \
  --name "Example" \
  --document-version "2"

```

Salida:

```
{
```

```
"Description": {
  "Name": "Example",
  "DefaultVersion": "2"
}
```

Para obtener más información, consulte [Redacción SSM del contenido de un documento](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [UpdateDocumentDefaultVersion](#) la Referencia de AWS CLI comandos.

## update-document-metadata

En el siguiente ejemplo de código, se muestra cómo usar `update-document-metadata`.

### AWS CLI

Ejemplo: para aprobar la última versión de una plantilla de cambios

A continuación, se `update-document-metadata` proporciona una aprobación para la última versión de una plantilla de cambios que se ha enviado para su revisión.

```
aws ssm update-document-metadata \
  --name MyChangeManagerTemplate \
  --document-reviews 'Action=Approve, Comment=[{Type=Comment, Content=Approved!}]'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Revisión y aprobación o rechazo de plantillas de cambios](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte la Referencia [UpdateDocumentMetadata](#) de AWS CLI comandos.

## update-document

En el siguiente ejemplo de código, se muestra cómo usar `update-document`.

### AWS CLI

Creación de una nueva versión de un documento

En el siguiente ejemplo de `update-document` se crea una nueva versión de un documento cuando se ejecuta en un computador con Windows. El documento especificado por `--document` debe tener el JSON formato indicado. Tenga en cuenta que se debe hacer referencia a `file://` seguido de la ruta del archivo de contenido. Debido a que `$` está al principio del parámetro `--document-version`, en Windows debe escribir el valor entre comillas dobles. En Linux, macOS o cuando se le PowerShell solicite, debe escribir el valor entre comillas simples.

Versión de Windows:

```
aws ssm update-document \  
  --name "RunShellScript" \  
  --content "file://RunShellScript.json" \  
  --document-version "$LATEST"
```

Versión de Linux o Mac:

```
aws ssm update-document \  
  --name "RunShellScript" \  
  --content "file://RunShellScript.json" \  
  --document-version '$LATEST'
```

Salida:

```
{  
  "DocumentDescription": {  
    "Status": "Updating",  
    "Hash": "f775e5df4904c6fa46686c4722fae9de1950dace25cd9608ff8d622046b68d9b",  
    "Name": "RunShellScript",  
    "Parameters": [  
      {  
        "Type": "StringList",  
        "Name": "commands",  
        "Description": "(Required) Specify a shell script or a command to  
run."  
      }  
    ],  
    "DocumentType": "Command",  
    "PlatformTypes": [  
      "Linux"  
    ],  
    "DocumentVersion": "2",
```



```

    "HashType": "Sha256",
    "CreateDate": 1487899655.152,
    "Owner": "809632081692",
    "SchemaVersion": "2.0",
    "DefaultVersion": "1",
    "LatestVersion": "2",
    "Description": "Run an updated script"
  }
}

```

- Para API obtener más información, consulte [UpdateDocument](#) la Referencia de AWS CLI comandos.

## update-maintenance-window-target

En el siguiente ejemplo de código, se muestra cómo usar `update-maintenance-window-target`.

### AWS CLI

Para actualizar un objetivo de ventana de mantenimiento

En el siguiente `update-maintenance-window-target` ejemplo, se actualiza solo el nombre de un objetivo de ventana de mantenimiento.

```

aws ssm update-maintenance-window-target \
  --window-id "mw-0c5ed765acEXAMPLE" \
  --window-target-id "57e8344e-fe64-4023-8191-6bf05EXAMPLE" \
  --name "NewName" \
  --no-replace

```

Salida:

```

{
  "Description": "",
  "OwnerInformation": "",
  "WindowTargetId": "57e8344e-fe64-4023-8191-6bf05EXAMPLE",
  "WindowId": "mw-0c5ed765acEXAMPLE",
  "Targets": [
    {
      "Values": [

```

```

        "i-1234567890EXAMPLE"
      ],
      "Key": "InstanceIds"
    }
  ],
  "Name": "NewName"
}

```

Para obtener más información, consulte [Actualizar una ventana de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [UpdateMaintenanceWindowTarget](#) la Referencia de AWS CLI comandos.

## update-maintenance-window-task

En el siguiente ejemplo de código, se muestra cómo usar `update-maintenance-window-task`.

### AWS CLI

Para actualizar una tarea de la ventana de mantenimiento

En el siguiente `update-maintenance-window-task` ejemplo, se actualiza la función de servicio de una tarea de la ventana de mantenimiento.

```

aws ssm update-maintenance-window-task \
  --window-id "mw-0c5ed765acEXAMPLE" \
  --window-task-id "23d3809e-9fbe-4ddf-b41a-b49d7EXAMPLE" \
  --service-role-arn "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM"

```

Salida:

```

{
  "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
  "MaxErrors": "1",
  "TaskArn": "AWS-UpdateEC2Config",
  "MaxConcurrency": "1",
  "WindowTaskId": "23d3809e-9fbe-4ddf-b41a-b49d7EXAMPLE",
  "TaskParameters": {},
  "Priority": 1,
}

```

```

    "TaskInvocationParameters": {
      "RunCommand": {
        "TimeoutSeconds": 600,
        "Parameters": {
          "allowDowngrade": [
            "false"
          ]
        }
      }
    },
    "WindowId": "mw-0c5ed765acEXAMPLE",
    "Description": "UpdateEC2Config",
    "Targets": [
      {
        "Values": [
          "57e8344e-fe64-4023-8191-6bf05EXAMPLE"
        ],
        "Key": "WindowTargetIds"
      }
    ],
    "Name": "UpdateEC2Config"
  }
}

```

Para obtener más información, consulte [Actualizar una ventana de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [UpdateMaintenanceWindowTask](#) la Referencia de AWS CLI comandos.

## update-maintenance-window

En el siguiente ejemplo de código, se muestra cómo usar `update-maintenance-window`.

### AWS CLI

Ejemplo 1: actualización de un periodo de mantenimiento

En el siguiente ejemplo de `update-maintenance-window` se actualiza el nombre de un periodo de mantenimiento.

```

aws ssm update-maintenance-window \
  --window-id "mw-1a2b3c4d5e6f7g8h9" \

```

```
--name "My-Renamed-MW"
```

Salida:

```
{
  "Cutoff": 1,
  "Name": "My-Renamed-MW",
  "Schedule": "cron(0 16 ? * TUE *)",
  "Enabled": true,
  "AllowUnassociatedTargets": true,
  "WindowId": "mw-1a2b3c4d5e6f7g8h9",
  "Duration": 4
}
```

Ejemplo 2: desactivación de un periodo de mantenimiento

En el siguiente ejemplo de `update-maintenance-window` se desactiva un periodo de mantenimiento.

```
aws ssm update-maintenance-window \
  --window-id "mw-1a2b3c4d5e6f7g8h9" \
  --no-enabled
```

Ejemplo 3: activación de un periodo de mantenimiento

En el siguiente ejemplo de `update-maintenance-window` se activa un periodo de mantenimiento.

```
aws ssm update-maintenance-window \
  --window-id "mw-1a2b3c4d5e6f7g8h9" \
  --enabled
```

Para obtener más información, consulte [Actualizar una ventana de mantenimiento \(AWS CLI\)](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [UpdateMaintenanceWindow](#) la Referencia de AWS CLI comandos.

## update-managed-instance-role

En el siguiente ejemplo de código, se muestra cómo usar `update-managed-instance-role`.

## AWS CLI

Para actualizar la IAM función de una instancia gestionada

En el siguiente `update-managed-instance-role` ejemplo, se actualiza el perfil de IAM instancia de una instancia gestionada.

```
aws ssm update-managed-instance-role \  
  --instance-id "mi-08ab247cdfEXAMPLE" \  
  --iam-role "ExampleRole"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte el [paso 4: Crear un perfil de IAM instancia para Systems Manager](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [UpdateManagedInstanceRole](#) la Referencia de AWS CLI comandos.

## update-ops-item

En el siguiente ejemplo de código, se muestra cómo usar `update-ops-item`.

### AWS CLI

Para actualizar un OpsItem

En el siguiente `update-ops-item` ejemplo se actualizan la descripción, la prioridad y la categoría de un OpsItem. Además, el comando especifica un SNS tema al que se envían las notificaciones cuando OpsItem se edita o cambia.

```
aws ssm update-ops-item \  
  --ops-item-id "oi-287b5EXAMPLE" \  
  --description "Primary OpsItem for failover event 2020-01-01-fh398yf" \  
  --priority 2 \  
  --category "Security" \  
  --notifications "Arn=arn:aws:sns:us-east-2:111222333444:my-us-east-2-topic"
```

Salida:

```
This command produces no output.
```

Para obtener más información, consulte [Trabajar con OpsItems](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [UpdateOpsItem](#) la Referencia de AWS CLI comandos.

## update-patch-baseline

En el siguiente ejemplo de código, se muestra cómo usar `update-patch-baseline`.

### AWS CLI

Ejemplo 1: actualización de una línea de base de revisiones

En el siguiente ejemplo de `update-patch-baseline` se agregan dos revisiones especificadas como rechazadas y una revisión como aprobada a la línea de base de revisiones especificada.

```
aws ssm update-patch-baseline \  
  --baseline-id "pb-0123456789abcdef0" \  
  --rejected-patches "KB2032276" "MS10-048" \  
  --approved-patches "KB2124261"
```

Salida:

```
{  
  "BaselineId": "pb-0123456789abcdef0",  
  "Name": "WindowsPatching",  
  "OperatingSystem": "WINDOWS",  
  "GlobalFilters": {  
    "PatchFilters": []  
  },  
  "ApprovalRules": {  
    "PatchRules": [  
      {  
        "PatchFilterGroup": {  
          "PatchFilters": [  
            {  
              "Key": "PRODUCT",  
              "Values": [  
                "WindowsServer2016"  
              ]  
            }  
          ]  
        }  
      }  
    ]  
  }  
}
```

```

        },
        "ComplianceLevel": "CRITICAL",
        "ApproveAfterDays": 0,
        "EnableNonSecurity": false
    }
]
},
"ApprovedPatches": [
    "KB2124261"
],
"ApprovedPatchesComplianceLevel": "UNSPECIFIED",
"ApprovedPatchesEnableNonSecurity": false,
"RejectedPatches": [
    "KB2032276",
    "MS10-048"
],
"RejectedPatchesAction": "ALLOW_AS_DEPENDENCY",
"CreateDate": 1550244180.465,
"ModifiedDate": 1550244180.465,
"Description": "Patches for Windows Servers",
"Sources": []
}

```

## Ejemplo 2: cambio del nombre de una línea de base de revisiones

En el siguiente ejemplo de `update-patch-baseline` se cambia el nombre de la línea de base de revisiones especificada.

```

aws ssm update-patch-baseline \
  --baseline-id "pb-0713accee01234567" \
  --name "Windows-Server-2012-R2-Important-and-Critical-Security-Updates"

```

Para obtener más información, consulte Actualizar o eliminar una línea base de parches <<https://docs.aws.amazon.com/systems-manager/latest/userguide/patch-baseline-update-or-delete.html>> en la Guía del usuario de Systems Manager.AWS

- Para obtener API más información, [UpdatePatchBaseline](#) consulte AWS CLI la Referencia de comandos.

## update-resource-data-sync

En el siguiente ejemplo de código, se muestra cómo usar `update-resource-data-sync`.

## AWS CLI

Para actualizar la sincronización de datos de un recurso

En el siguiente `update-resource-data-sync` ejemplo, se actualiza la sincronización `SyncFromSource` de datos de un recurso.

```
aws ssm update-resource-data-sync \  
  --sync-name exampleSync \  
  --sync-type SyncFromSource \  
  --sync-source '{"SourceType":"SingleAccountMultiRegions", "SourceRegions":["us-east-1", "us-west-2"]}'
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración del explorador de Systems Manager para mostrar datos de varias cuentas y regiones](#) en la Guía del usuario de AWS Systems Manager.

- Para API obtener más información, consulte [UpdateResourceDataSync](#) la Referencia de AWS CLI comandos.

## update-service-setting

En el siguiente ejemplo de código, se muestra cómo usar `update-service-setting`.

### AWS CLI

Para actualizar la configuración del servicio para el rendimiento del almacén de parámetros

En el siguiente `update-service-setting` ejemplo, se actualiza la configuración actual del servicio del rendimiento del almacén de parámetros en la región especificada para utilizar un rendimiento mayor.

```
aws ssm update-service-setting \  
  --setting-id arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-store/high-throughput-enabled \  
  --setting-value true
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Aumentar el rendimiento del almacén de parámetros](#) en la Guía del usuario de AWS Systems Manager.



- Para API obtener más información, consulte [UpdateServiceSetting](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon Textract usando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Amazon Textract.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **analyze-document**

En el siguiente ejemplo de código, se muestra cómo usar `analyze-document`.

AWS CLI

Análisis del texto de un documento

En el siguiente ejemplo de `analyze-document` se muestra cómo analizar el texto de un documento.

Linux/macOS:

```
aws textract analyze-document \  
  --document '{"S3Object":{"Bucket":"bucket","Name":"document"}}' \  
  --feature-types ['TABLES','FORMS']
```

Windows:

```
aws textract analyze-document \  
  --document "{\\"S3Object\\":{\\"Bucket\\":\\"bucket\\",\\"Name\\":\\"document\\"}}" \  
  --feature-types ["TABLES","FORMS"] \  
  --region region-name
```

Salida:

```
{  
  "Blocks": [  
    {  
      "Geometry": {  
        "BoundingBox": {  
          "Width": 1.0,  
          "Top": 0.0,  
          "Left": 0.0,  
          "Height": 1.0  
        },  
        "Polygon": [  
          {  
            "Y": 0.0,  
            "X": 0.0  
          },  
          {  
            "Y": 0.0,  
            "X": 1.0  
          },  
          {  
            "Y": 1.0,  
            "X": 1.0  
          },  
          {  
            "Y": 1.0,  
            "X": 0.0  
          }  
        ]  
      },  
      "Relationships": [  
        {  
          "Type": "CHILD",  
          "Ids": [  
            "87586964-d50d-43e2-ace5-8a890657b9a0",  
            "a1e72126-21d9-44f4-a8d6-5c385f9002ba",  
            "e889d012-8a6b-4d2e-b7cd-7a8b327d876a"  
          ]  
        }  
      ]  
    }  
  ]  
}
```

```

        ]
      }
    ],
    "BlockType": "PAGE",
    "Id": "c2227f12-b25d-4e1f-baea-1ee180d926b2"
  }
],
"DocumentMetadata": {
  "Pages": 1
}
}

```

Para obtener más información, consulte [Análisis del texto de un documento con Amazon Textract](#) en la Guía para desarrolladores de Amazon Textract

- Para API obtener más información, consulte [AnalyzeDocument](#) la Referencia de AWS CLI comandos.

## detect-document-text

En el siguiente ejemplo de código, se muestra cómo usar `detect-document-text`.

### AWS CLI

#### Detección de texto en un documento

En el siguiente ejemplo de `detect-document-text` se muestra cómo detectar texto en un documento.

Linux/macOS:

```
aws textract detect-document-text \
  --document '{"S3Object":{"Bucket":"bucket","Name":"document"}}'
```

Windows:

```
aws textract detect-document-text \
  --document '{"S3Object":{"Bucket":"bucket","Name":"document"}}' \
  --region region-name
```

Salida:

```
{
  "Blocks": [
    {
      "Geometry": {
        "BoundingBox": {
          "Width": 1.0,
          "Top": 0.0,
          "Left": 0.0,
          "Height": 1.0
        },
        "Polygon": [
          {
            "Y": 0.0,
            "X": 0.0
          },
          {
            "Y": 0.0,
            "X": 1.0
          },
          {
            "Y": 1.0,
            "X": 1.0
          },
          {
            "Y": 1.0,
            "X": 0.0
          }
        ]
      },
      "Relationships": [
        {
          "Type": "CHILD",
          "Ids": [
            "896a9f10-9e70-4412-81ce-49ead73ed881",
            "0da18623-dc4c-463d-a3d1-9ac050e9e720",
            "167338d7-d38c-4760-91f1-79a8ec457bb2"
          ]
        }
      ],
      "BlockType": "PAGE",
      "Id": "21f0535e-60d5-4bc7-adf2-c05dd851fa25"
    },
    {
```

```
    "Relationships": [
      {
        "Type": "CHILD",
        "Ids": [
          "62490c26-37ea-49fa-8034-7a9ff9369c9c",
          "1e4f3f21-05bd-4da9-ba10-15d01e66604c"
        ]
      }
    ],
    "Confidence": 89.11581420898438,
    "Geometry": {
      "BoundingBox": {
        "Width": 0.33642634749412537,
        "Top": 0.17169663310050964,
        "Left": 0.13885067403316498,
        "Height": 0.49159330129623413
      },
      "Polygon": [
        {
          "Y": 0.17169663310050964,
          "X": 0.13885067403316498
        },
        {
          "Y": 0.17169663310050964,
          "X": 0.47527703642845154
        },
        {
          "Y": 0.6632899641990662,
          "X": 0.47527703642845154
        },
        {
          "Y": 0.6632899641990662,
          "X": 0.13885067403316498
        }
      ]
    },
    "Text": "He llo,",
    "BlockType": "LINE",
    "Id": "896a9f10-9e70-4412-81ce-49ead73ed881"
  },
  {
    "Relationships": [
      {
        "Type": "CHILD",
```

```
        "Ids": [
            "19b28058-9516-4352-b929-64d7cef29daf"
        ]
    },
],
"Confidence": 85.5694351196289,
"Geometry": {
    "BoundingBox": {
        "Width": 0.33182239532470703,
        "Top": 0.23131252825260162,
        "Left": 0.5091826915740967,
        "Height": 0.3766750991344452
    },
    "Polygon": [
        {
            "Y": 0.23131252825260162,
            "X": 0.5091826915740967
        },
        {
            "Y": 0.23131252825260162,
            "X": 0.8410050868988037
        },
        {
            "Y": 0.607987642288208,
            "X": 0.8410050868988037
        },
        {
            "Y": 0.607987642288208,
            "X": 0.5091826915740967
        }
    ]
},
"Text": "worlc",
"BlockType": "LINE",
"Id": "0da18623-dc4c-463d-a3d1-9ac050e9e720"
}
],
"DocumentMetadata": {
    "Pages": 1
}
}
```

Para obtener más información, consulte [Detección del texto de un documento con Amazon Textract](#) en la Guía para desarrolladores de Amazon Textract

- Para API obtener más información, consulte [DetectDocumentText](#) la Referencia de AWS CLI comandos.

## get-document-analysis

En el siguiente ejemplo de código, se muestra cómo usar `get-document-analysis`.

### AWS CLI

Obtención de los resultados del análisis de texto asíncrono de un documento de varias páginas

En el siguiente ejemplo de `get-document-analysis` se muestra cómo obtener los resultados del análisis de texto asíncrono de un documento de varias páginas.

```
aws textract get-document-analysis \  
  --job-id df7cf32ebbd2a5de113535fcf4d921926a701b09b4e7d089f3aebadb41e0712b \  
  --max-results 1000
```

Salida:

```
{  
  "Blocks": [  
    {  
      "Geometry": {  
        "BoundingBox": {  
          "Width": 1.0,  
          "Top": 0.0,  
          "Left": 0.0,  
          "Height": 1.0  
        },  
        "Polygon": [  
          {  
            "Y": 0.0,  
            "X": 0.0  
          },  
          {  
            "Y": 0.0,  
            "X": 1.0  
          }  
        ]  
      }  
    ]  
  }
```

```

        {
            "Y": 1.0,
            "X": 1.0
        },
        {
            "Y": 1.0,
            "X": 0.0
        }
    ]
},
"Relationships": [
    {
        "Type": "CHILD",
        "Ids": [
            "75966e64-81c2-4540-9649-d66ec341cd8f",
            "bb099c24-8282-464c-a179-8a9fa0a057f0",
            "5ebf522d-f9e4-4dc7-bfae-a288dc094595"
        ]
    }
],
"BlockType": "PAGE",
"Id": "247c28ee-b63d-4aeb-9af0-5f7ea8ba109e",
"Page": 1
}
],
"NextToken": "cY1W3eTFvoB0cH7YrKVudI4Gb0H8J0xAYLo8xI/JunCIPWCthaKQ+07n/
ElyutsSy0+1VOImoTRmP1zw4P0RFtaeV9Bzhnfedpx1YqwB4xaGDA==",
"DocumentMetadata": {
    "Pages": 1
},
"JobStatus": "SUCCEEDED"
}

```

Para obtener más información, consulte [Detección y análisis de texto en documentos de varias páginas](#) en la Guía para desarrolladores de Amazon Textract

- Para API obtener más información, consulte [GetDocumentAnalysis](#) la Referencia de AWS CLI comandos.

## get-document-text-detection

En el siguiente ejemplo de código, se muestra cómo usar `get-document-text-detection`.



## AWS CLI

Para obtener los resultados de la detección asíncrona de texto en un documento de varias páginas

En el siguiente `get-document-text-detection` ejemplo, se muestra cómo obtener los resultados de la detección asíncrona de texto en un documento de varias páginas.

```
aws textract get-document-text-detection \  
  --job-id 57849a3dc627d4df74123dca269d69f7b89329c870c65bb16c9fd63409d200b9 \  
  --max-results 1000
```

### Salida

```
{  
  "Blocks": [  
    {  
      "Geometry": {  
        "BoundingBox": {  
          "Width": 1.0,  
          "Top": 0.0,  
          "Left": 0.0,  
          "Height": 1.0  
        },  
        "Polygon": [  
          {  
            "Y": 0.0,  
            "X": 0.0  
          },  
          {  
            "Y": 0.0,  
            "X": 1.0  
          },  
          {  
            "Y": 1.0,  
            "X": 1.0  
          },  
          {  
            "Y": 1.0,  
            "X": 0.0  
          }  
        ]  
      },  
    },  
  ],  
}
```

```

    "Relationships": [
      {
        "Type": "CHILD",
        "Ids": [
          "1b926a34-0357-407b-ac8f-ec473160c6a9",
          "0c35dc17-3605-4c9d-af1a-d9451059df51",
          "dea3db8a-52c2-41c0-b50c-81f66f4aa758"
        ]
      }
    ],
    "BlockType": "PAGE",
    "Id": "84671a5e-8c99-43be-a9d1-6838965da33e",
    "Page": 1
  }
],
"NextToken": "GcqyoAJuZwuj0T35EN4LCI3EUzMtiLq3nKyFFHvU5q1SaIdEBcSty+njNgoWwuMP/
muqc96S4o5NzDqehhXvhkodMyV050JGyms51srCxibWJw==",
"DocumentMetadata": {
  "Pages": 1
},
"JobStatus": "SUCCEEDED"
}

```

Para obtener más información, consulte [Detección y análisis de texto en documentos de varias páginas](#) en la [Guía para desarrolladores de Amazon Textract](#)

- Para obtener API más información, consulte la [Referencia de comandos `GetDocumentTextDetection`](#).AWS CLI

## start-document-analysis

En el siguiente ejemplo de código, se muestra cómo usar `start-document-analysis`.

### AWS CLI

Cómo empezar a analizar el texto de un documento de varias páginas

En el siguiente ejemplo de `start-document-analysis` se muestra cómo iniciar el análisis asíncrono de texto de un documento de varias páginas.

Linux/macOS:

```
aws textract start-document-analysis \
```

```
--document-location '{"S3Object":{"Bucket":"bucket","Name":"document"}}' \
--feature-types ["TABLES","FORMS"] \
--notification-channel "SNSTopicArn=arn:snsTopic,RoleArn=roleArn"
```

Windows:

```
aws textract start-document-analysis \
  --document-location "{\"S3Object\":{\"Bucket\":\"bucket\",\"Name\":\"document\
  \\\}}\" \
  --feature-types "[\"TABLES\", \"FORMS\"]\" \
  --region region-name \
  --notification-channel "SNSTopicArn=arn:snsTopic,RoleArn=roleArn"
```

Salida:

```
{
  "JobId": "df7cf32ebbd2a5de113535fcf4d921926a701b09b4e7d089f3aebadb41e0712b"
}
```

Para obtener más información, consulte Detección y análisis de texto en documentos de varias páginas en la Guía para desarrolladores de Amazon Textract

- Para API obtener más información, consulte [StartDocumentAnalysis](#) la Referencia de AWS CLI comandos.

## start-document-text-detection

En el siguiente ejemplo de código, se muestra cómo usar start-document-text-detection.

AWS CLI

Cómo empezar a detectar texto en un documento de varias páginas

En el siguiente ejemplo de start-document-text-detection se muestra cómo iniciar la detección asíncrona de texto de un documento de varias páginas.

Linux/macOS:

```
aws textract start-document-text-detection \
  --document-location '{"S3Object":{"Bucket":"bucket","Name":"document"}}' \
  --notification-channel "SNSTopicArn=arn:snsTopic,RoleArn=roleARN"
```

Windows:

```
aws textract start-document-text-detection \
  --document-location "{\"S3Object\":{\"Bucket\":\"bucket\"},\"Name\":\"document\"}\" \
  --region region-name \
  --notification-channel "SNSTopicArn=arn:snsTopic,RoleArn=roleArn"
```

Salida:

```
{
  "JobId": "57849a3dc627d4df74123dca269d69f7b89329c870c65bb16c9fd63409d200b9"
}
```

Para obtener más información, consulte [Detección y análisis de texto en documentos de varias páginas en la Guía para desarrolladores de Amazon Textract](#)

- Para API obtener más información, consulte [StartDocumentTextDetection](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon Transcribe utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface con Amazon Transcribe.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **create-language-model**

En el siguiente ejemplo de código, se muestra cómo usar `create-language-model`.

## AWS CLI

Ejemplo 1: Crear un modelo de lenguaje personalizado utilizando datos de entrenamiento y ajuste.

En el siguiente `create-language-model` ejemplo, se crea un modelo de idioma personalizado. Puede utilizar un modelo de lenguaje personalizado para mejorar el rendimiento de la transcripción en ámbitos como el jurídico, la hostelería, las finanzas y los seguros. Para el código de idioma, introduce un código de idioma válido. Para `base-model-name` ello, especifique el modelo base que mejor se adapte a la frecuencia de muestreo del audio que desea transcribir con su modelo de idioma personalizado. En `model-name`, especifique el nombre con el que quiere llamar al modelo de lenguaje personalizado.

```
aws transcribe create-language-model \  
  --language-code language-code \  
  --base-model-name base-model-name \  
  --model-name cli-clm-example \  
  --input-data-config S3Uri="s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix-for-  
training-data",TuningDataS3Uri="s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix-for-  
tuning-data",DataAccessRoleArn="arn:aws:iam::AWS-account-number:role/IAM-role-with-  
permissions-to-create-a-custom-language-model"
```

Salida:

```
{  
  "LanguageCode": "language-code",  
  "BaseModelName": "base-model-name",  
  "ModelName": "cli-clm-example",  
  "InputDataConfig": {  
    "S3Uri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/",  
    "TuningDataS3Uri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/",  
    "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-with-  
permissions-create-a-custom-language-model"  
  },  
  "ModelStatus": "IN_PROGRESS"  
}
```

Para obtener más información, consulte [Mejora de la precisión de la transcripción específica de dominios con modelos de idiomas personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 2: Para crear un modelo de lenguaje personalizado utilizando únicamente datos de entrenamiento.

En el siguiente ejemplo de `create-language-model` se transcribe un archivo de audio. Puede utilizar un modelo de lenguaje personalizado para mejorar el rendimiento de la transcripción en ámbitos como el jurídico, la hostelería, las finanzas y los seguros. Para el código de idioma, introduce un código de idioma válido. Para `base-model-name` ello, especifique el modelo base que mejor se adapte a la frecuencia de muestreo del audio que desea transcribir con su modelo de idioma personalizado. En `model-name`, especifique el nombre con el que quiere llamar al modelo de lenguaje personalizado.

```
aws transcribe create-language-model \  
  --language-code en-US \  
  --base-model-name base-model-name \  
  --model-name cli-clm-example \  
  --input-data-config S3Uri="s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix-For-  
Training-Data",DataAccessRoleArn="arn:aws:iam::AWS-account-number:role/IAM-role-  
with-permissions-to-create-a-custom-language-model"
```

Salida:

```
{  
  "LanguageCode": "en-US",  
  "BaseModelName": "base-model-name",  
  "ModelName": "cli-clm-example",  
  "InputDataConfig": {  
    "S3Uri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix-For-Training-Data/",  
    "DataAccessRoleArn": "arn:aws:iam::your-AWS-account-number:role/IAM-role-  
with-permissions-to-create-a-custom-language-model"  
  },  
  "ModelStatus": "IN_PROGRESS"  
}
```

Para obtener más información, consulte [Mejora de la precisión de la transcripción específica de dominios con modelos de idiomas personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte la Referencia [CreateLanguageModel](#) de AWS CLI comandos.

## create-medical-vocabulary

En el siguiente ejemplo de código, se muestra cómo usar `create-medical-vocabulary`.

### AWS CLI

Para crear un vocabulario médico personalizado

En el siguiente ejemplo de `create-medical-vocabulary` se crea un vocabulario personalizado. Para crear un vocabulario personalizado, debe haber creado un archivo de texto con todos los términos que desee transcribir con mayor precisión. Para `vocabulary-file-uri`, especifique el Amazon Simple Storage Service (Amazon URI S3) de ese archivo de texto. En `language-code`, especifique un código de idioma correspondiente al idioma del vocabulario personalizado. En `vocabulary-name`, especifique cómo desea llamar al vocabulario personalizado.

```
aws transcribe create-medical-vocabulary \
  --vocabulary-name cli-medical-vocab-example \
  --language-code language-code \
  --vocabulary-file-uri https://DOC-EXAMPLE-BUCKET.AWS-Region.amazonaws.com/the-
  text-file-for-the-medical-custom-vocabulary.txt
```

Salida:

```
{
  "VocabularyName": "cli-medical-vocab-example",
  "LanguageCode": "language-code",
  "VocabularyState": "PENDING"
}
```

Para obtener más información, consulte [Vocabularios médicos personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [CreateMedicalVocabulary](#) la Referencia de AWS CLI comandos.

## create-vocabulary-filter

En el siguiente ejemplo de código, se muestra cómo usar `create-vocabulary-filter`.

## AWS CLI

Para crear un filtro de vocabulario

En el siguiente `create-vocabulary-filter` ejemplo, se crea un filtro de vocabulario que utiliza un archivo de texto que contiene una lista de palabras que no desea que aparezcan en una transcripción. Para el código de idioma, especifique el código de idioma correspondiente al idioma del filtro de vocabulario. Para `vocabulary-filter-file-uri`, especifique el Amazon Simple Storage Service (Amazon URI S3) del archivo de texto. Para `vocabulary-filter-name`, especifique el nombre de su filtro de vocabulario.

```
aws transcribe create-vocabulary-filter \  
  --language-code language-code \  
  --vocabulary-filter-file-uri s3://DOC-EXAMPLE-BUCKET/vocabulary-filter.txt \  
  --vocabulary-filter-name cli-vocabulary-filter-example
```

Salida:

```
{  
  "VocabularyFilterName": "cli-vocabulary-filter-example",  
  "LanguageCode": "language-code"  
}
```

Para obtener más información, consulte [Filtrar palabras no deseadas](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [CreateVocabularyFilter](#) la Referencia de AWS CLI comandos.

## **create-vocabulary**

En el siguiente ejemplo de código, se muestra cómo usar `create-vocabulary`.

## AWS CLI

Para crear un vocabulario personalizado

En el siguiente ejemplo de `create-vocabulary` se crea un vocabulario personalizado. Para crear un vocabulario personalizado, debe haber creado un archivo de texto con todos los términos que desee transcribir con mayor precisión. Para `vocabulary-file-uri`, especifique el



Amazon Simple Storage Service (Amazon URI S3) de ese archivo de texto. En `language-code`, especifique un código de idioma correspondiente al idioma del vocabulario personalizado. En `vocabulary-name`, especifique cómo desea llamar al vocabulario personalizado.

```
aws transcribe create-vocabulary \  
  --language-code language-code \  
  --vocabulary-name cli-vocab-example \  
  --vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/the-text-file-  
  for-the-custom-vocabulary.txt
```

Salida:

```
{  
  "VocabularyName": "cli-vocab-example",  
  "LanguageCode": "language-code",  
  "VocabularyState": "PENDING"  
}
```

Para obtener más información, consulte [Vocabularios personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [CreateVocabulary](#) la Referencia de AWS CLI comandos.

## **delete-language-model**

En el siguiente ejemplo de código, se muestra cómo usar `delete-language-model`.

AWS CLI

Para eliminar un modelo de idioma personalizado

En el siguiente `delete-language-model` ejemplo, se elimina un modelo de idioma personalizado.

```
aws transcribe delete-language-model \  
  --model-name model-name
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Mejora de la precisión de la transcripción específica de dominios con modelos de idiomas personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [DeleteLanguageModel](#) la Referencia de AWS CLI comandos.

## **delete-medical-transcription-job**

En el siguiente ejemplo de código, se muestra cómo usar `delete-medical-transcription-job`.

### AWS CLI

Para eliminar un trabajo de transcripción médica

En el siguiente ejemplo de `delete-medical-transcription-job` se elimina un trabajo de transcripción médica.

```
aws transcribe delete-medical-transcription-job \  
  --medical-transcription-job-name medical-transcription-job-name
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteMedicalTranscriptionJob](#) la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [DeleteMedicalTranscriptionJob](#) la Referencia de AWS CLI comandos.

## **delete-medical-vocabulary**

En el siguiente ejemplo de código, se muestra cómo usar `delete-medical-vocabulary`.

### AWS CLI

Para eliminar un vocabulario médico personalizado

En el siguiente `delete-medical-vocabulary` ejemplo, se elimina un vocabulario médico personalizado. Para el nombre del vocabulario, especifique el nombre del vocabulario médico personalizado.

```
aws transcribe delete-vocabulary \  
  --vocabulary-name medical-custom-vocabulary-name
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Vocabularios médicos personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte la Referencia de [DeleteMedicalVocabulary](#) comandos AWS CLI .

## delete-transcription-job

En el siguiente ejemplo de código, se muestra cómo usar delete-transcription-job.

AWS CLI

Para eliminar uno de los trabajos de transcripción

En el siguiente ejemplo de delete-transcription-job se elimina uno de los trabajos de transcripción.

```
aws transcribe delete-transcription-job \  
  --transcription-job-name your-transcription-job
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [DeleteTranscriptionJob](#) la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [DeleteTranscriptionJob](#) la Referencia de AWS CLI comandos.

## delete-vocabulary-filter

En el siguiente ejemplo de código, se muestra cómo usar delete-vocabulary-filter.

AWS CLI

Para eliminar un filtro de vocabulario

En el siguiente delete-vocabulary-filter ejemplo, se elimina un filtro de vocabulario.

```
aws transcribe delete-vocabulary-filter \  
  --vocabulary-filter-name vocabulary-filter-name
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Filtrar palabras no deseadas](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [DeleteVocabularyFilter](#) la Referencia de AWS CLI comandos.

## delete-vocabulary

En el siguiente ejemplo de código, se muestra cómo usar delete-vocabulary.

### AWS CLI

Para eliminar un vocabulario personalizado

En el siguiente ejemplo de delete-vocabulary se elimina un vocabulario personalizado.

```
aws transcribe delete-vocabulary \  
  --vocabulary-name vocabulary-name
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Vocabularios personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [DeleteVocabulary](#) la Referencia de AWS CLI comandos.

## describe-language-model

En el siguiente ejemplo de código, se muestra cómo usar describe-language-model.

### AWS CLI

Para obtener información sobre un modelo de lenguaje personalizado específico

En el siguiente describe-language-model ejemplo, se obtiene información sobre un modelo de idioma personalizado específico. Por ejemplo, en esta sección BaseModelName puede ver

si su modelo está entrenado con un WideBand modelo NarrowBand o. Los modelos de lenguaje personalizados con un modelo NarrowBand base pueden transcribir audio con una frecuencia de muestreo inferior a 16kHz. Los modelos de lenguaje que utilizan un modelo WideBand base pueden transcribir audio con una frecuencia de muestreo superior a 16. kHz El parámetro S3Uri indica el prefijo Amazon S3 que ha utilizado para acceder a los datos de entrenamiento para crear el modelo de idioma personalizado.

```
aws transcribe describe-language-model \  
  --model-name cli-clm-example
```

Salida:

```
{  
  "LanguageModel": {  
    "ModelName": "cli-clm-example",  
    "CreateTime": "2020-09-25T17:57:38.504000+00:00",  
    "LastModifiedTime": "2020-09-25T17:57:48.585000+00:00",  
    "LanguageCode": "language-code",  
    "BaseModelName": "base-model-name",  
    "ModelStatus": "IN_PROGRESS",  
    "UpgradeAvailability": false,  
    "InputDataConfig": {  
      "S3Uri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/",  
      "TuningDataS3Uri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/",  
      "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-with-permissions-to-create-a-custom-language-model"  
    }  
  }  
}
```

Para obtener más información, consulte [Mejora de la precisión de la transcripción específica de dominios con modelos de idiomas personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte la Referencia de [DescribeLanguageModel](#) comandos AWS CLI .

## **get-medical-transcription-job**

En el siguiente ejemplo de código, se muestra cómo usar `get-medical-transcription-job`.

## AWS CLI

Para obtener información sobre un trabajo específico de transcripción médica

En el siguiente `get-medical-transcription-job` ejemplo se obtiene información sobre un trabajo de transcripción médica específico. Para acceder a los resultados de la transcripción, utilice el `TranscriptFileUri` parámetro. Si has activado funciones adicionales para el trabajo de transcripción, puedes verlas en el objeto de configuración. El parámetro `Especialidad` muestra la especialidad médica del proveedor. El parámetro `Type` indica si el discurso del trabajo de transcripción corresponde a una conversación médica o a un dictado médico.

```
aws transcribe get-medical-transcription-job \
  --medical-transcription-job-name vocabulary-dictation-medical-transcription-job
```

Salida:

```
{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "vocabulary-dictation-medical-transcription-
job",
    "TranscriptionJobStatus": "COMPLETED",
    "LanguageCode": "en-US",
    "MediaSampleRateHertz": 48000,
    "MediaFormat": "mp4",
    "Media": {
      "MediaFileUri": "s3://Amazon-S3-Prefix/your-audio-file.file-extension"
    },
    "Transcript": {
      "TranscriptFileUri": "https://s3.Region.amazonaws.com/Amazon-S3-Prefix/
vocabulary-dictation-medical-transcription-job.json"
    },
    "StartTime": "2020-09-21T21:17:27.045000+00:00",
    "CreationTime": "2020-09-21T21:17:27.016000+00:00",
    "CompletionTime": "2020-09-21T21:17:59.561000+00:00",
    "Settings": {
      "ChannelIdentification": false,
      "ShowAlternatives": false,
      "VocabularyName": "cli-medical-vocab-example"
    },
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}
```

```
}
```

Para obtener más información, consulte [Batch Transcription](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [GetMedicalTranscriptionJob](#) la Referencia de AWS CLI comandos.

## get-medical-vocabulary

En el siguiente ejemplo de código, se muestra cómo usar `get-medical-vocabulary`.

### AWS CLI

Para obtener información sobre un vocabulario médico personalizado

En el siguiente `get-medical-vocabulary` ejemplo se obtiene información sobre un vocabulario médico personalizado. Puede usar el `VocabularyState` parámetro para ver el estado de procesamiento del vocabulario. Si es así `READY`, puedes usarlo en la `StartMedicalTranscriptionJob` operación. :

```
aws transcribe get-medical-vocabulary \  
  --vocabulary-name medical-vocab-example
```

Salida:

```
{  
  "VocabularyName": "medical-vocab-example",  
  "LanguageCode": "en-US",  
  "VocabularyState": "READY",  
  "LastModifiedTime": "2020-09-19T23:59:04.349000+00:00",  
  "DownloadUri": "https://link-to-download-the-text-file-used-to-create-your-  
  medical-custom-vocabulary"  
}
```

Para obtener más información, consulte [Vocabularios médicos personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [GetMedicalVocabulary](#) la Referencia de AWS CLI comandos.

## get-transcription-job

En el siguiente ejemplo de código, se muestra cómo usar `get-transcription-job`.

### AWS CLI

Para obtener información acerca de un determinado trabajo de transcripción

En el siguiente ejemplo de `get-transcription-job` se obtiene información sobre un determinado trabajo de transcripción. Para acceder a los resultados de la transcripción, utilice el `TranscriptFileUri` parámetro. Utilice el `MediaFileUri` parámetro para ver qué archivo de audio ha transcrito con este trabajo. Puede usar el objeto `Settings` para ver las características opcionales que ha habilitado en el trabajo de transcripción.

```
aws transcribe get-transcription-job \  
  --transcription-job-name your-transcription-job
```

Salida:

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "your-transcription-job",  
    "TranscriptionJobStatus": "COMPLETED",  
    "LanguageCode": "language-code",  
    "MediaSampleRateHertz": 48000,  
    "MediaFormat": "mp4",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.file-extension"  
    },  
    "Transcript": {  
      "TranscriptFileUri": "https://Amazon-S3-file-location-of-transcription-  
output"  
    },  
    "StartTime": "2020-09-18T22:27:23.970000+00:00",  
    "CreationTime": "2020-09-18T22:27:23.948000+00:00",  
    "CompletionTime": "2020-09-18T22:28:21.197000+00:00",  
    "Settings": {  
      "ChannelIdentification": false,  
      "ShowAlternatives": false  
    },  
    "IdentifyLanguage": true,  
    "IdentifiedLanguageScore": 0.8672199249267578  
  }  
}
```



```
}
```

Para obtener más información, consulte [Introducción \(interfaz de línea de AWS comandos\)](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [GetTranscriptionJob](#) la Referencia de AWS CLI comandos.

## get-vocabulary-filter

En el siguiente ejemplo de código, se muestra cómo usar `get-vocabulary-filter`.

### AWS CLI

Para obtener información sobre un filtro de vocabulario

En el siguiente `get-vocabulary-filter` ejemplo, se obtiene información sobre un filtro de vocabulario. Puede utilizar el `DownloadUri` parámetro para obtener la lista de palabras que utilizó para crear el filtro de vocabulario.

```
aws transcribe get-vocabulary-filter \  
  --vocabulary-filter-name testFilter
```

Salida:

```
{  
  "VocabularyFilterName": "testFilter",  
  "LanguageCode": "language-code",  
  "LastModifiedTime": "2020-05-07T22:39:32.147000+00:00",  
  "DownloadUri": "https://Amazon-S3-location-to-download-your-vocabulary-filter"  
}
```

Para obtener más información, consulte [Filtrar palabras no deseadas](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [GetVocabularyFilter](#) la Referencia de AWS CLI comandos.

## get-vocabulary

En el siguiente ejemplo de código, se muestra cómo usar `get-vocabulary`.

## AWS CLI

Para obtener información sobre un vocabulario personalizado

En el siguiente ejemplo de `get-vocabulary` se obtiene información sobre un vocabulario personalizado creado anteriormente.

```
aws transcribe get-vocabulary \  
  --vocabulary-name cli-vocab-1
```

Salida:

```
{  
  "VocabularyName": "cli-vocab-1",  
  "LanguageCode": "language-code",  
  "VocabularyState": "READY",  
  "LastModifiedTime": "2020-09-19T23:22:32.836000+00:00",  
  "DownloadUri": "https://link-to-download-the-text-file-used-to-create-your-  
  custom-vocabulary"  
}
```

Para obtener más información, consulte [Vocabularios personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [GetVocabulary](#) la Referencia de AWS CLI comandos.

## list-language-models

En el siguiente ejemplo de código, se muestra cómo usar `list-language-models`.

## AWS CLI

Para enumerar sus modelos de lenguaje personalizados

En el siguiente `list-language-models` ejemplo, se enumeran los modelos de idioma personalizados asociados a su AWS cuenta y región. Puede utilizar los `TuningDataS3Uri` parámetros `S3Uri` y para buscar los prefijos de Amazon S3 que ha utilizado como datos de entrenamiento o datos de ajuste. Esto `BaseModelName` le indica si ha utilizado un modelo de lenguaje o un `NarrowBand WideBand` modelo para crear un modelo de lenguaje personalizado.

Puede transcribir audio con una frecuencia de muestreo inferior a 16 kHz con un modelo de lenguaje personalizado utilizando un modelo NarrowBand base. Puede transcribir 16 kHz o más audios con un modelo de idioma personalizado utilizando un modelo WideBand base. El `ModelStatus` parámetro muestra si puede utilizar el modelo de idioma personalizado en un trabajo de transcripción. Si el valor es `COMPLETED`, puede usarlo en un trabajo de transcripción.

```
aws transcribe list-language-models
```

Salida:

```
{
  "Models": [
    {
      "ModelName": "cli-clm-2",
      "CreateTime": "2020-09-25T17:57:38.504000+00:00",
      "LastModifiedTime": "2020-09-25T17:57:48.585000+00:00",
      "LanguageCode": "language-code",
      "BaseModelName": "WideBand",
      "ModelStatus": "IN_PROGRESS",
      "UpgradeAvailability": false,
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/clm-training-data/",
        "TuningDataS3Uri": "s3://DOC-EXAMPLE-BUCKET/clm-tuning-data/",
        "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-used-to-create-the-custom-language-model"
      }
    },
    {
      "ModelName": "cli-clm-1",
      "CreateTime": "2020-09-25T17:16:01.835000+00:00",
      "LastModifiedTime": "2020-09-25T17:16:15.555000+00:00",
      "LanguageCode": "language-code",
      "BaseModelName": "WideBand",
      "ModelStatus": "IN_PROGRESS",
      "UpgradeAvailability": false,
      "InputDataConfig": {
        "S3Uri": "s3://DOC-EXAMPLE-BUCKET/clm-training-data/",
        "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-used-to-create-the-custom-language-model"
      }
    },
    {
```

```
"ModelName": "clm-console-1",
"CreateTime": "2020-09-24T19:26:28.076000+00:00",
"LastModifiedTime": "2020-09-25T04:25:22.271000+00:00",
"LanguageCode": "language-code",
"BaseModelName": "NarrowBand",
"ModelStatus": "COMPLETED",
"UpgradeAvailability": false,
"InputDataConfig": {
  "S3Uri": "s3://DOC-EXAMPLE-BUCKET/clm-training-data/",
  "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-used-to-create-the-custom-language-model"
}
]
}
```

Para obtener más información, consulte [Mejora de la precisión de la transcripción específica de dominios con modelos de idiomas personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [ListLanguageModels](#) la Referencia de AWS CLI comandos.

## list-medical-transcription-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-medical-transcription-jobs`.

### AWS CLI

Para enumerar los trabajos de transcripción médica

En el siguiente `list-medical-transcription-jobs` ejemplo, se enumeran los trabajos de transcripción médica asociados a su AWS cuenta y región. Para obtener más información sobre un trabajo de transcripción concreto, copie el valor de un `MedicalTranscriptionJobName` parámetro en el resultado de la transcripción y especifique ese valor en la `MedicalTranscriptionJobName` opción del `get-medical-transcription-job` comando. Para ver más trabajos de transcripción, copie el valor del `NextToken` parámetro, vuelva a ejecutar el `list-medical-transcription-jobs` comando y especifique ese valor en la `--next-token` opción.

```
aws transcribe list-medical-transcription-jobs
```

## Salida:

```
{
  "NextToken": "3/PblzkiGhzjER3KHuQt2fmbPLF7cDYafjFMEoGn440N/
gsuUSTIkGyanvRE6WMXfd/ZTEc2EZj+P9eii/
z102FDY1i6RLI0WoRX4RwMisVrh9G0Kie0Y8ikBCdtqLZB10Wa9McC+eb01
+LaDtZPC4u6ttoHLRlEfzqstHXSgapXg3tEBtm9piIaPB6M0M5BB6t86+qtmocTR/
qrteHZBBudhTfbCwhsxaqujHiiUvFdm3BQbKKWIW06yV9b+4f38oD2lVIan
+vfUs3gBYA15VTDmXXzQPBQ0HPjtwmFI+IWX15nSUjWuN3TUy1HgPWzDaYT8qBtu0Z+3UG4V6b
+K2CC0XszXg5rBq9hYgNzy4XoFh/6s5DoSznzq49Q9xHgHdT2yBADFmvFK7myZBsJ75+2vQZ0SVpWUPy3WT/32zFAcoEL
+mFYfUjtTZ8n/jq7aQEjQ42A
+X/7K6Jg0cdVPtEg8P1Dr5kgYYG3q30mYXX37U3FZuJmnTI63VtIXsNn0U5eGoY0btpk00Nq9UkzGsjxqj84ZD5n
+S0EGy9ZUYBJRRcGeYUM3Q4DbSJfUwSAqcFdLIWZdp8qIREMQIBWy7BLwSdyqsQo2vRrd53hm5aWM7SVf6pPq6X/
IXR5+1eU00D8/coaTT4ES2DerbV6RkV4o0VT1d0SdVX/
MmtkNG8nYj8PqU07w7988quh1ZP6D80veJS1q73tUUR9MjnGernW2tAnvnLNhdefBcD
+sZVfYq3iBMFY7wTy1P1G6NqW9GrYDY0X3tTPW1D7phpbVSYkrh/
PdYrps5UxnsGoA1b7L/FfAXDfUoGrGUB4N3JsPYXX9D++g+6gV1qBBs/
WFF934aKqfD6UTggm/zV3GA0WiBpfvAZRvEb924i6yGHYMC7y5401ZAwSBupmI
+FFd13CaP04kN1vJlth6aM5vUPXg4BpyUhtbRhwd/KxCvf9K0tLJGyL1A==",
  "MedicalTranscriptionJobSummaries": [
    {
      "MedicalTranscriptionJobName": "vocabulary-dictation-medical-
transcription-job",
      "CreationTime": "2020-09-21T21:17:27.016000+00:00",
      "StartTime": "2020-09-21T21:17:27.045000+00:00",
      "CompletionTime": "2020-09-21T21:17:59.561000+00:00",
      "LanguageCode": "en-US",
      "TranscriptionJobStatus": "COMPLETED",
      "OutputLocationType": "CUSTOMER_BUCKET",
      "Specialty": "PRIMARYCARE",
      "Type": "DICTATION"
    },
    {
      "MedicalTranscriptionJobName": "alternatives-dictation-medical-
transcription-job",
      "CreationTime": "2020-09-21T21:01:14.569000+00:00",
      "StartTime": "2020-09-21T21:01:14.592000+00:00",
      "CompletionTime": "2020-09-21T21:01:43.606000+00:00",
      "LanguageCode": "en-US",
      "TranscriptionJobStatus": "COMPLETED",
      "OutputLocationType": "CUSTOMER_BUCKET",
      "Specialty": "PRIMARYCARE",
      "Type": "DICTATION"
    }
  ],
}
```

```
{
  "MedicalTranscriptionJobName": "alternatives-conversation-medical-
transcription-job",
  "CreationTime": "2020-09-21T19:09:18.171000+00:00",
  "StartTime": "2020-09-21T19:09:18.199000+00:00",
  "CompletionTime": "2020-09-21T19:10:22.516000+00:00",
  "LanguageCode": "en-US",
  "TranscriptionJobStatus": "COMPLETED",
  "OutputLocationType": "CUSTOMER_BUCKET",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION"
},
{
  "MedicalTranscriptionJobName": "speaker-id-conversation-medical-
transcription-job",
  "CreationTime": "2020-09-21T18:43:37.157000+00:00",
  "StartTime": "2020-09-21T18:43:37.265000+00:00",
  "CompletionTime": "2020-09-21T18:44:21.192000+00:00",
  "LanguageCode": "en-US",
  "TranscriptionJobStatus": "COMPLETED",
  "OutputLocationType": "CUSTOMER_BUCKET",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION"
},
{
  "MedicalTranscriptionJobName": "multichannel-conversation-medical-
transcription-job",
  "CreationTime": "2020-09-20T23:46:44.053000+00:00",
  "StartTime": "2020-09-20T23:46:44.081000+00:00",
  "CompletionTime": "2020-09-20T23:47:35.851000+00:00",
  "LanguageCode": "en-US",
  "TranscriptionJobStatus": "COMPLETED",
  "OutputLocationType": "CUSTOMER_BUCKET",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION"
}
]
```

Para obtener más información, consulte <https://docs.aws.amazon.com/transcribe/latest/dg/batch-med-transcription.html> en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte la Referencia de comandos.

[ListMedicalTranscriptionJobs](#) AWS CLI

## list-medical-vocabularies

En el siguiente ejemplo de código, se muestra cómo usar `list-medical-vocabularies`.

### AWS CLI

Para enumerar sus vocabularios médicos personalizados

En el siguiente `list-medical-vocabularies` ejemplo, se enumeran los vocabularios médicos personalizados asociados a tu AWS cuenta y región. Para obtener más información sobre un trabajo de transcripción concreto, copie el valor de un `MedicalTranscriptionJobName` parámetro en el resultado de la transcripción y especifique ese valor para la `MedicalTranscriptionJobName` opción del `get-medical-transcription-job` comando. Para ver más trabajos de transcripción, copie el valor del `NextToken` parámetro, vuelva a ejecutar el `list-medical-transcription-jobs` comando y especifique ese valor en la `--next-token` opción.

```
aws transcribe list-medical-vocabularies
```

Salida:

```
{
  "Vocabularies": [
    {
      "VocabularyName": "cli-medical-vocab-2",
      "LanguageCode": "en-US",
      "LastModifiedTime": "2020-09-21T21:44:59.521000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "cli-medical-vocab-1",
      "LanguageCode": "en-US",
      "LastModifiedTime": "2020-09-19T23:59:04.349000+00:00",
      "VocabularyState": "READY"
    }
  ]
}
```

Para obtener más información, consulte [Vocabularios médicos personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [ListMedicalVocabularies](#) la Referencia de AWS CLI comandos.

## list-transcription-jobs

En el siguiente ejemplo de código, se muestra cómo usar `list-transcription-jobs`.

### AWS CLI

Para enumerar los trabajos de transcripción

En el siguiente `list-transcription-jobs` ejemplo, se enumeran los trabajos de transcripción asociados a su AWS cuenta y región.

```
aws transcribe list-transcription-jobs
```

Salida:

```
{
  "NextToken": "NextToken",
  "TranscriptionJobSummaries": [
    {
      "TranscriptionJobName": "speak-id-job-1",
      "CreationTime": "2020-08-17T21:06:15.391000+00:00",
      "StartTime": "2020-08-17T21:06:15.416000+00:00",
      "CompletionTime": "2020-08-17T21:07:05.098000+00:00",
      "LanguageCode": "language-code",
      "TranscriptionJobStatus": "COMPLETED",
      "OutputLocationType": "SERVICE_BUCKET"
    },
    {
      "TranscriptionJobName": "job-1",
      "CreationTime": "2020-08-17T20:50:24.207000+00:00",
      "StartTime": "2020-08-17T20:50:24.230000+00:00",
      "CompletionTime": "2020-08-17T20:52:18.737000+00:00",
      "LanguageCode": "language-code",
      "TranscriptionJobStatus": "COMPLETED",
      "OutputLocationType": "SERVICE_BUCKET"
    },
    {
      "TranscriptionJobName": "sdk-test-job-4",
      "CreationTime": "2020-08-17T20:32:27.917000+00:00",
```



```

    "StartTime": "2020-08-17T20:32:27.956000+00:00",
    "CompletionTime": "2020-08-17T20:33:15.126000+00:00",
    "LanguageCode": "language-code",
    "TranscriptionJobStatus": "COMPLETED",
    "OutputLocationType": "SERVICE_BUCKET"
  },
  {
    "TranscriptionJobName": "Diarization-speak-id",
    "CreationTime": "2020-08-10T22:10:09.066000+00:00",
    "StartTime": "2020-08-10T22:10:09.116000+00:00",
    "CompletionTime": "2020-08-10T22:26:48.172000+00:00",
    "LanguageCode": "language-code",
    "TranscriptionJobStatus": "COMPLETED",
    "OutputLocationType": "SERVICE_BUCKET"
  },
  {
    "TranscriptionJobName": "your-transcription-job-name",
    "CreationTime": "2020-07-29T17:45:09.791000+00:00",
    "StartTime": "2020-07-29T17:45:09.826000+00:00",
    "CompletionTime": "2020-07-29T17:46:20.831000+00:00",
    "LanguageCode": "language-code",
    "TranscriptionJobStatus": "COMPLETED",
    "OutputLocationType": "SERVICE_BUCKET"
  }
]
}

```

Para obtener más información, consulte [Introducción \(interfaz de línea de AWS comandos\)](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [ListTranscriptionJobs](#) la Referencia de AWS CLI comandos.

## list-vocabularies

En el siguiente ejemplo de código, se muestra cómo usar `list-vocabularies`.

### AWS CLI

Para enumerar los vocabularios personalizados

En el siguiente `list-vocabularies` ejemplo, se enumeran los vocabularios personalizados asociados a su AWS cuenta y región.

**aws transcribe list-vocabularies**

Salida:

```
{
  "NextToken": "NextToken",
  "Vocabularies": [
    {
      "VocabularyName": "ards-test-1",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-27T22:00:27.330000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "sample-test",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-24T23:04:11.044000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "CRLF-to-LF-test-3-1",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-24T22:12:22.277000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "CRLF-to-LF-test-2",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-24T21:53:50.455000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "CRLF-to-LF-1-1",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-24T21:39:33.356000+00:00",
      "VocabularyState": "READY"
    }
  ]
}
```

Para obtener más información, consulte [Vocabularios personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [ListVocabularies](#) la Referencia de AWS CLI comandos.

## list-vocabulary-filters

En el siguiente ejemplo de código, se muestra cómo usar `list-vocabulary-filters`.

AWS CLI

Para enumerar tus filtros de vocabulario

En el siguiente `list-vocabulary-filters` ejemplo, se enumeran los filtros de vocabulario asociados a tu AWS cuenta y región.

```
aws transcribe list-vocabulary-filters
```

Salida:

```
{
  "NextToken": "NextToken": [
    {
      "VocabularyFilterName": "testFilter",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-05-07T22:39:32.147000+00:00"
    },
    {
      "VocabularyFilterName": "testFilter2",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-05-21T23:29:35.174000+00:00"
    },
    {
      "VocabularyFilterName": "filter2",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-05-08T20:18:26.426000+00:00"
    },
    {
      "VocabularyFilterName": "filter-review",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-06-03T18:52:30.448000+00:00"
    },
    {
      "VocabularyFilterName": "crlf-filt",
```

```

        "LanguageCode": "language-code",
        "LastModifiedTime": "2020-05-22T19:42:42.737000+00:00"
    }
]
}

```

Para obtener más información, consulte [Filtrar palabras no deseadas](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [ListVocabularyFilters](#) la Referencia de AWS CLI comandos.

## start-medical-transcription-job

En el siguiente ejemplo de código, se muestra cómo usar `start-medical-transcription-job`.

### AWS CLI

Ejemplo 1: para transcribir un dictado médico almacenado como un archivo de audio

En el siguiente ejemplo de `start-medical-transcription-job` se transcribe un archivo de audio. Especifique la ubicación del resultado de la transcripción en el parámetro `OutputBucketName`.

```

aws transcribe start-medical-transcription-job \
  --cli-input-json file://myfile.json

```

Contenidos de `myfile.json`:

```

{
  "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "DICTATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  }
}

```

Salida:

```
{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-20T00:35:22.256000+00:00",
    "CreationTime": "2020-09-20T00:35:22.218000+00:00",
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}
```

Para obtener más información, consulte [Información general de la transcripción por lotes](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 2: para transcribir un diálogo entre el médico y el paciente almacenado como un archivo de audio

En el siguiente ejemplo de `start-medical-transcription-job` se transcribe un archivo de audio que contiene un diálogo entre el médico y el paciente. La ubicación del resultado de la transcripción se especifica en el `OutputBucketName` parámetro.

```
aws transcribe start-medical-transcription-job \
  --cli-input-json file://mysecondfile.json
```

Contenidos de `mysecondfile.json`:

```
{
  "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  }
}
```

**Salida:**

```
{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "simple-conversation-medical-transcription-
job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-20T23:19:49.965000+00:00",
    "CreationTime": "2020-09-20T23:19:49.941000+00:00",
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  }
}
```

Para obtener más información, consulte [Información general de la transcripción por lotes](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 3: para transcribir un archivo de audio multicanal de un diálogo entre el médico y el paciente

En el siguiente ejemplo de `start-medical-transcription-job` se transcribe el audio de cada canal del archivo de audio y se combinan las transcripciones independientes de cada canal en un único resultado de transcripción. Especifique la ubicación del resultado de la transcripción en el parámetro `OutputBucketName`.

```
aws transcribe start-medical-transcription-job \
  --cli-input-json file:///mythirdfile.json
```

Contenidos de `mythirdfile.json`:

```
{
  "MedicalTranscriptionJobName": "multichannel-conversation-medical-transcription-
job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
```

```

    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "Settings":{
      "ChannelIdentification": true
    }
  }
}

```

Salida:

```

{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "multichannel-conversation-medical-
transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-20T23:46:44.081000+00:00",
    "CreationTime": "2020-09-20T23:46:44.053000+00:00",
    "Settings": {
      "ChannelIdentification": true
    },
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  }
}

```

Para obtener más información, consulte [Identificación de canales](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 4: para transcribir un archivo de audio de un diálogo entre el médico y el paciente e identificar a los interlocutores en el resultado de la transcripción

En el siguiente ejemplo de `start-medical-transcription-job` se transcribe un archivo de audio y se etiqueta la voz de cada interlocutor en el resultado de la transcripción. Especifique la ubicación del resultado de la transcripción en el parámetro `OutputBucketName`.

```

aws transcribe start-medical-transcription-job \
  --cli-input-json file://myfourthfile.json

```

## Contenidos de myfourthfile.json:

```
{
  "MedicalTranscriptionJobName": "speaker-id-conversation-medical-transcription-
job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  },
  "Settings": {
    "ShowSpeakerLabels": true,
    "MaxSpeakerLabels": 2
  }
}
```

## Salida:

```
{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "speaker-id-conversation-medical-
transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-21T18:43:37.265000+00:00",
    "CreationTime": "2020-09-21T18:43:37.157000+00:00",
    "Settings": {
      "ShowSpeakerLabels": true,
      "MaxSpeakerLabels": 2
    },
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  }
}
```

Para obtener más información, consulte [Identificación de interlocutores](#) en la Guía para desarrolladores de Amazon Transcribe.



Ejemplo 5: para transcribir una conversación médica almacenada como un archivo de audio con hasta dos alternativas de transcripción

En el siguiente ejemplo de `start-medical-transcription-job` crea hasta dos transcripciones alternativas a partir de un único archivo de audio. Cada transcripción tiene un nivel de confianza asociado. De forma predeterminada, Amazon Transcribe devuelve la transcripción con el nivel de confianza más alto. Puede especificar que Amazon Transcribe devuelva otras transcripciones con niveles de confianza más bajos. Especifique la ubicación del resultado de la transcripción en el parámetro `OutputBucketName`.

```
aws transcribe start-medical-transcription-job \  
  --cli-input-json file://myfifthfile.json
```

Contenidos de `myfifthfile.json`:

```
{  
  "MedicalTranscriptionJobName": "alternatives-conversation-medical-transcription-  
job",  
  "LanguageCode": "language-code",  
  "Specialty": "PRIMARYCARE",  
  "Type": "CONVERSATION",  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
  },  
  "Settings": {  
    "ShowAlternatives": true,  
    "MaxAlternatives": 2  
  }  
}
```

Salida:

```
{  
  "MedicalTranscriptionJob": {  
    "MedicalTranscriptionJobName": "alternatives-conversation-medical-  
transcription-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "language-code",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
    }  
  }  
}
```

```

    },
    "StartTime": "2020-09-21T19:09:18.199000+00:00",
    "CreationTime": "2020-09-21T19:09:18.171000+00:00",
    "Settings": {
      "ShowAlternatives": true,
      "MaxAlternatives": 2
    },
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  }
}

```

Para obtener más información, consulte [Transcripciones alternativas](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 6: para transcribir un archivo de audio de un dictado médico con hasta dos transcripciones alternativas

En el siguiente ejemplo de `start-medical-transcription-job` se transcribe un archivo de audio y se utiliza un filtro de vocabulario para ocultar las palabras no deseadas. La ubicación del resultado de la transcripción se especifica en el `OutputBucketName` parámetro.

```

aws transcribe start-medical-transcription-job \
  --cli-input-json file://mysixthfile.json

```

Contenidos de `mysixthfile.json`:

```

{
  "MedicalTranscriptionJobName": "alternatives-conversation-medical-transcription-
job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "DICTATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  },
  "Settings": {
    "ShowAlternatives": true,
    "MaxAlternatives": 2
  }
}

```

**Salida:**

```
{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "alternatives-dictation-medical-
transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-21T21:01:14.592000+00:00",
    "CreationTime": "2020-09-21T21:01:14.569000+00:00",
    "Settings": {
      "ShowAlternatives": true,
      "MaxAlternatives": 2
    },
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}
```

Para obtener más información, consulte [Transcripciones alternativas](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 7: para transcribir un archivo de audio de un dictado médico con mayor precisión mediante un vocabulario personalizado

En el siguiente ejemplo de `start-medical-transcription-job` se transcribe un archivo de audio y se utiliza un vocabulario médico personalizado que haya creado anteriormente para aumentar la precisión de la transcripción. Especifique la ubicación del resultado de la transcripción en el parámetro `OutputBucketName`.

```
aws transcribe start-transcription-job \
  --cli-input-json file://myseventhfile.json
```

Contenidos de `mysixthfile.json`:

```
{
  "MedicalTranscriptionJobName": "vocabulary-dictation-medical-transcription-job",
  "LanguageCode": "language-code",
```

```

    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION",
    "OutputBucketName": "DOC-EXAMPLE-BUCKET",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "Settings": {
      "VocabularyName": "cli-medical-vocab-1"
    }
  }
}

```

**Salida:**

```

{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "vocabulary-dictation-medical-transcription-
job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-21T21:17:27.045000+00:00",
    "CreationTime": "2020-09-21T21:17:27.016000+00:00",
    "Settings": {
      "VocabularyName": "cli-medical-vocab-1"
    },
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}

```

Para obtener más información, consulte [Vocabularios médicos personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [StartMedicalTranscriptionJob](#) la Referencia de AWS CLI comandos.

**start-transcription-job**

En el siguiente ejemplo de código, se muestra cómo usar `start-transcription-job`.

## AWS CLI

Ejemplo 1: para transcribir un archivo de audio

En el siguiente ejemplo de `start-transcription-job` se transcribe un archivo de audio.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://myfile.json
```

Contenidos de `myfile.json`:

```
{  
  "TranscriptionJobName": "cli-simple-transcription-job",  
  "LanguageCode": "the-language-of-your-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-  
name.file-extension"  
  }  
}
```

Para obtener más información, consulte [Introducción \(interfaz de línea de AWS comandos\)](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 2: para transcribir un archivo de audio multicanal

En el siguiente ejemplo de `start-transcription-job` se transcribe un archivo de audio multicanal.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://mysecondfile.json
```

Contenidos de `mysecondfile.json`:

```
{  
  "TranscriptionJobName": "cli-channelid-job",  
  "LanguageCode": "the-language-of-your-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-  
name.file-extension"  
  },  
}
```

```

    "Settings":{
      "ChannelIdentification":true
    }
  }
}

```

Salida:

```

{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-channelid-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "StartTime": "2020-09-17T16:07:56.817000+00:00",
    "CreationTime": "2020-09-17T16:07:56.784000+00:00",
    "Settings": {
      "ChannelIdentification": true
    }
  }
}

```

Para obtener más información, consulte [Transcripción de audio multicanal](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 3: para transcribir un archivo de audio e identificar a los distintos interlocutores

En el siguiente ejemplo de `start-transcription-job` se transcribe un archivo de audio y se identifica la voz de los interlocutores en el resultado de la transcripción.

```

aws transcribe start-transcription-job \
  --cli-input-json file://mythirdfile.json

```

Contenidos de `mythirdfile.json`:

```

{
  "TranscriptionJobName": "cli-speakerid-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {

```

```

    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-
name.file-extension"
  },
  "Settings":{
    "ShowSpeakerLabels": true,
    "MaxSpeakerLabels": 2
  }
}

```

Salida:

```

{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-speakerid-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "StartTime": "2020-09-17T16:22:59.696000+00:00",
    "CreationTime": "2020-09-17T16:22:59.676000+00:00",
    "Settings": {
      "ShowSpeakerLabels": true,
      "MaxSpeakerLabels": 2
    }
  }
}

```

Para obtener más información, consulte [Identificación de interlocutores](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 4: para transcribir un archivo de audio y ocultar las palabras no deseadas en el resultado de la transcripción

En el siguiente ejemplo de `start-transcription-job` se transcribe un archivo de audio y se utiliza un filtro de vocabulario que ha creado anteriormente para ocultar las palabras no deseadas.

```

aws transcribe start-transcription-job \
  --cli-input-json file://myfourthfile.json

```

Contenidos de `myfourthfile.json`:

```
{
  "TranscriptionJobName": "cli-filter-mask-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-
name.file-extension"
  },
  "Settings":{
    "VocabularyFilterName": "your-vocabulary-filter",
    "VocabularyFilterMethod": "mask"
  }
}
```

Salida:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-filter-mask-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
      "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
    },
    "StartTime": "2020-09-18T16:36:18.568000+00:00",
    "CreationTime": "2020-09-18T16:36:18.547000+00:00",
    "Settings": {
      "VocabularyFilterName": "your-vocabulary-filter",
      "VocabularyFilterMethod": "mask"
    }
  }
}
```

Para obtener más información, consulte [Filtrado de transcripciones](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 5: para transcribir un archivo de audio y eliminar las palabras no deseadas en el resultado de la transcripción

En el siguiente ejemplo de `start-transcription-job` se transcribe un archivo de audio y se utiliza un filtro de vocabulario que ha creado anteriormente para ocultar las palabras no deseadas.

```
aws transcribe start-transcription-job \
```



```
--cli-input-json file://myfifthfile.json
```

Contenidos de myfifthfile.json:

```
{
  "TranscriptionJobName": "cli-filter-remove-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-
name.file-extension"
  },
  "Settings":{
    "VocabularyFilterName": "your-vocabulary-filter",
    "VocabularyFilterMethod": "remove"
  }
}
```

Salida:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-filter-remove-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "StartTime": "2020-09-18T16:36:18.568000+00:00",
    "CreationTime": "2020-09-18T16:36:18.547000+00:00",
    "Settings": {
      "VocabularyFilterName": "your-vocabulary-filter",
      "VocabularyFilterMethod": "remove"
    }
  }
}
```

Para obtener más información, consulte [Filtrado de transcripciones](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 6: para transcribir un archivo de audio con mayor precisión mediante un vocabulario personalizado

En el siguiente ejemplo de `start-transcription-job` se transcribe un archivo de audio y se utiliza un filtro de vocabulario que ha creado anteriormente para ocultar las palabras no deseadas.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://mysixthfile.json
```

Contenidos de `mysixthfile.json`:

```
{  
  "TranscriptionJobName": "cli-vocab-job",  
  "LanguageCode": "the-language-of-your-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-  
name.file-extension"  
  },  
  "Settings": {  
    "VocabularyName": "your-vocabulary"  
  }  
}
```

Salida:

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "cli-vocab-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "the-language-of-your-transcription-job",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-  
file-name.file-extension"  
    },  
    "StartTime": "2020-09-18T16:36:18.568000+00:00",  
    "CreationTime": "2020-09-18T16:36:18.547000+00:00",  
    "Settings": {  
      "VocabularyName": "your-vocabulary"  
    }  
  }  
}
```

Para obtener más información, consulte [Filtrado de transcripciones](#) en la Guía para desarrolladores de Amazon Transcribe.

## Ejemplo 7: para identificar el idioma de un archivo de audio y transcribirlo

En el siguiente ejemplo de `start-transcription-job` se transcribe un archivo de audio y se utiliza un filtro de vocabulario que ha creado anteriormente para ocultar las palabras no deseadas.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://myseventhfile.json
```

Contenidos de `myseventhfile.json`:

```
{  
  "TranscriptionJobName": "cli-identify-language-transcription-job",  
  "IdentifyLanguage": true,  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-  
name.file-extension"  
  }  
}
```

Salida:

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "cli-identify-language-transcription-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-  
file-name.file-extension"  
    },  
    "StartTime": "2020-09-18T22:27:23.970000+00:00",  
    "CreationTime": "2020-09-18T22:27:23.948000+00:00",  
    "IdentifyLanguage": true  
  }  
}
```

Para obtener más información, consulte [Identificación del idioma](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 8: para transcribir un archivo de audio con información de identificación personal redactada

En el siguiente ejemplo de `start-transcription-job` se transcribe un archivo de audio y se redacta la información de identificación personal en el resultado de la transcripción.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://myeighthfile.json
```

Contenidos de `myeighthfile.json`:

```
{  
  "TranscriptionJobName": "cli-redaction-job",  
  "LanguageCode": "language-code",  
  "Media": {  
    "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"  
  },  
  "ContentRedaction": {  
    "RedactionOutput": "redacted",  
    "RedactionType": "PII"  
  }  
}
```

Salida:

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "cli-redaction-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "language-code",  
    "Media": {  
      "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"  
    },  
    "StartTime": "2020-09-25T23:49:13.195000+00:00",  
    "CreationTime": "2020-09-25T23:49:13.176000+00:00",  
    "ContentRedaction": {  
      "RedactionType": "PII",  
      "RedactionOutput": "redacted"  
    }  
  }  
}
```

Para obtener más información, consulte [Redacción automática de contenido](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 9: Para generar una transcripción con información de identificación personal (PII) redactada y una transcripción sin editar

En el siguiente ejemplo de `start-transcription-job` se generan dos transcripciones del archivo de audio, una con la información de identificación personal redactada y la otra sin ninguna redacción.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://myninthfile.json
```

Contenidos de `myninthfile.json`:

```
{  
  "TranscriptionJobName": "cli-redaction-job-with-unredacted-transcript",  
  "LanguageCode": "language-code",  
  "Media": {  
    "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"  
  },  
  "ContentRedaction": {  
    "RedactionOutput": "redacted_and_unredacted",  
    "RedactionType": "PII"  
  }  
}
```

Salida:

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "cli-redaction-job-with-unredacted-transcript",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "language-code",  
    "Media": {  
      "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"  
    },  
    "StartTime": "2020-09-25T23:59:47.677000+00:00",  
    "CreationTime": "2020-09-25T23:59:47.653000+00:00",  
    "ContentRedaction": {  
      "RedactionType": "PII",  
      "RedactionOutput": "redacted_and_unredacted"  
    }  
  }  
}
```

```
}
```

Para obtener más información, consulte [Redacción automática de contenido](#) en la Guía para desarrolladores de Amazon Transcribe.

Ejemplo 10: para usar un modelo de idioma personalizado que haya creado previamente para transcribir un archivo de audio

En el siguiente ejemplo de `start-transcription-job` se transcribe un archivo de audio con un modelo de idioma personalizado que haya creado anteriormente.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://mytenthfile.json
```

Contenidos de `mytenthfile.json`:

```
{  
  "TranscriptionJobName": "cli-clm-2-job-1",  
  "LanguageCode": "language-code",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.file-extension"  
  },  
  "ModelSettings": {  
    "LanguageModelName": "cli-clm-2"  
  }  
}
```

Salida:

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "cli-clm-2-job-1",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "language-code",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.file-extension"  
    },  
    "StartTime": "2020-09-28T17:56:01.835000+00:00",  
    "CreationTime": "2020-09-28T17:56:01.801000+00:00",  
    "ModelSettings": {  
      "LanguageModelName": "cli-clm-2"  
    }  
  }  
}
```

```
}  
}
```

Para obtener más información, consulte [Mejora de la precisión de la transcripción específica de dominios con modelos de idiomas personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para obtener API más información, consulte la Referencia de comandos. [StartTranscriptionJob](#) AWS CLI

## update-medical-vocabulary

En el siguiente ejemplo de código, se muestra cómo usar `update-medical-vocabulary`.

### AWS CLI

Para actualizar un vocabulario médico personalizado con nuevos términos.

En el siguiente `update-medical-vocabulary` ejemplo, se sustituyen los términos utilizados en un vocabulario médico personalizado por otros nuevos. Requisito previo: para sustituir los términos de un vocabulario médico personalizado, se necesita un archivo con términos nuevos.

```
aws transcribe update-medical-vocabulary \  
  --vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/medical-custom-  
vocabulary.txt \  
  --vocabulary-name medical-custom-vocabulary \  
  --language-code language
```

Salida:

```
{  
  "VocabularyName": "medical-custom-vocabulary",  
  "LanguageCode": "en-US",  
  "VocabularyState": "PENDING"  
}
```

Para obtener más información, consulte [Vocabularios médicos personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [UpdateMedicalVocabulary](#) la Referencia de AWS CLI comandos.

## update-vocabulary-filter

En el siguiente ejemplo de código, se muestra cómo usar `update-vocabulary-filter`.

### AWS CLI

Para reemplazar las palabras de un filtro de vocabulario

En el siguiente `update-vocabulary-filter` ejemplo, se sustituyen las palabras de un filtro de vocabulario por otras nuevas. Requisito previo: para actualizar un filtro de vocabulario con las nuevas palabras, debe guardarlas como un archivo de texto.

```
aws transcribe update-vocabulary-filter \  
  --vocabulary-filter-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/your-text-  
file-to-update-your-vocabulary-filter.txt \  
  --vocabulary-filter-name vocabulary-filter-name
```

Salida:

```
{  
  "VocabularyFilterName": "vocabulary-filter-name",  
  "LanguageCode": "language-code",  
  "LastModifiedTime": "2020-09-23T18:40:35.139000+00:00"  
}
```

Para obtener más información, consulte [Filtrar palabras no deseadas](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [UpdateVocabularyFilter](#) la Referencia de AWS CLI comandos.

## update-vocabulary

En el siguiente ejemplo de código, se muestra cómo usar `update-vocabulary`.

### AWS CLI

Para actualizar un vocabulario personalizado con términos nuevos

En el siguiente ejemplo de `update-vocabulary` se sobrescriben los términos utilizados para crear un vocabulario personalizado con los nuevos que proporcione. Requisito previo: para sustituir los términos de un vocabulario personalizado, necesita un archivo con términos nuevos.



```
aws transcribe update-vocabulary \  
  --vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/custom-  
vocabulary.txt \  
  --vocabulary-name custom-vocabulary \  
  --language-code language-code
```

Salida:

```
{  
  "VocabularyName": "custom-vocabulary",  
  "LanguageCode": "language",  
  "VocabularyState": "PENDING"  
}
```

Para obtener más información, consulte [Vocabularios personalizados](#) en la Guía para desarrolladores de Amazon Transcribe.

- Para API obtener más información, consulte [UpdateVocabulary](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon Translate utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes AWS Command Line Interface mediante Amazon Translate.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **import-terminology**

En el siguiente ejemplo de código, se muestra cómo usar `import-terminology`.

## AWS CLI

Para importar una terminología personalizada desde un archivo

En el `import-terminology` ejemplo siguiente se crea una terminología llamada `MyTestTerminology` desde el `test-terminology.csv` archivo:

```
aws translate import-terminology \  
  --name MyTestTerminology \  
  --description "Creating a test terminology in AWS Translate" \  
  --merge-strategy OVERWRITE \  
  --data-file fileb://test-terminology.csv \  
  --terminology-data Format=CSV
```

Contenidos de `test-terminology.csv`:

en, fr, es, zh ¡Hola mundo! , Bonjour tout le monde! , ¡Hola Mundo! ,???? Amazon, Amazon, Amazon

Salida:

```
{  
  "TerminologyProperties": {  
    "SourceLanguageCode": "en",  
    "Name": "MyTestTerminology",  
    "TargetLanguageCodes": [  
      "fr",  
      "es",  
      "zh"  
    ],  
    "SizeBytes": 97,  
    "LastUpdatedAt": 1571089500.851,  
    "CreatedAt": 1571089500.851,  
    "TermCount": 6,  
    "Arn": "arn:aws:translate:us-west-2:123456789012:terminology/  
MyTestTerminology/LATEST",  
    "Description": "Creating a test terminology in AWS Translate"  
  }  
}
```

- Para API obtener más información, consulte [ImportTerminology](#) la Referencia de AWS CLI comandos.

## Trusted Advisor ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with Trusted Advisor.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **get-organization-recommendation**

En el siguiente ejemplo de código, se muestra cómo usar `get-organization-recommendation`.

AWS CLI

Para obtener una recomendación de la organización

En el siguiente `get-organization-recommendation` ejemplo, se obtiene una recomendación de organización por su identificador.

```
aws trustedadvisor get-organization-recommendation \  
  --organization-recommendation-identifier arn:aws:trustedadvisor::organization-  
  recommendation/9534ec9b-bf3a-44e8-8213-2ed68b39d9d5
```

Salida:

```
{  
  "organizationRecommendation": {  
    "arn": "arn:aws:trustedadvisor::organization-recommendation/9534ec9b-  
    bf3a-44e8-8213-2ed68b39d9d5",  
    "name": "Lambda Runtime Deprecation Warning",
```

```

    "description": "One or more lambdas are using a deprecated runtime",
    "awsServices": [
      "lambda"
    ],
    "checkArn": "arn:aws:trustedadvisor:::check/L4dfs2Q4C5",
    "id": "9534ec9b-bf3a-44e8-8213-2ed68b39d9d5",
    "lifecycleStage": "resolved",
    "pillars": [
      "security"
    ],
    "resourcesAggregates": {
      "errorCount": 0,
      "okCount": 0,
      "warningCount": 0
    },
    "source": "ta_check",
    "status": "warning",
    "type": "priority"
  }
}

```

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

- Para API obtener más información, consulte [GetOrganizationRecommendation](#) la Referencia de AWS CLI comandos.

## get-recommendation

En el siguiente ejemplo de código, se muestra cómo usar `get-recommendation`.

### AWS CLI

Para obtener una recomendación

En el siguiente `get-recommendation` ejemplo, se obtiene una recomendación por su identificador.

```

aws trustedadvisor get-recommendation \
  --recommendation-
  identifier arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578

```

**Salida:**

```
{
  "recommendation": {
    "arn": "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
    "name": "MFA Recommendation",
    "description": "Enable multi-factor authentication",
    "awsServices": [
      "iam"
    ],
    "checkArn": "arn:aws:trustedadvisor:::check/7DAFEemoDos",
    "id": "55fa4d2e-bbb7-491a-833b-5773e9589578",
    "lastUpdatedAt": "2023-11-01T15:57:58.673Z",
    "pillarSpecificAggregates": {
      "costOptimizing": {
        "estimatedMonthlySavings": 0.0,
        "estimatedPercentMonthlySavings": 0.0
      }
    },
    "pillars": [
      "security"
    ],
    "resourcesAggregates": {
      "errorCount": 1,
      "okCount": 0,
      "warningCount": 0
    },
    "source": "ta_check",
    "status": "error",
    "type": "standard"
  }
}
```

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

- Para API obtener más información, consulte [GetRecommendation](#) la Referencia de AWS CLI comandos.

**list-checks**

En el siguiente ejemplo de código, se muestra cómo usar `list-checks`.

## AWS CLI

Para enumerar las comprobaciones de Trusted Advisor

En el siguiente `list-checks` ejemplo, se enumeran todas las comprobaciones de Trusted Advisor.

```
aws trustedadvisor list-checks
```

Salida:

```
{
  "checkSummaries": [
    {
      "arn": "arn:aws:trustedadvisor:::check/1iG5NDGVre",
      "awsServices": [
        "EC2"
      ],
      "description": "Checks security groups for rules that allow unrestricted
access to a resource. Unrestricted access increases opportunities for malicious
activity (hacking, denial-of-service attacks, loss of data)",
      "id": "1iG5NDGVre",
      "metadata": {
        "0": "Region",
        "1": "Security Group Name",
        "2": "Security Group ID",
        "3": "Protocol",
        "4": "Port",
        "5": "Status",
        "6": "IP Range"
      },
      "name": "Security Groups - Unrestricted Access",
      "pillars": [
        "security"
      ],
      "source": "ta_check"
    },
    {
      "arn": "arn:aws:trustedadvisor:::check/1qazXsw23e",
      "awsServices": [
        "RDS"
      ],

```

```

    "description": "Checks your usage of RDS and provides recommendations
on purchase of Reserved Instances to help reduce costs incurred from using RDS
On-Demand. AWS generates these recommendations by analyzing your On-Demand usage
for the past 30 days. We then simulate every combination of reservations in the
generated category of usage in order to identify the best number of each type
of Reserved Instance to purchase to maximize your savings. This check covers
recommendations based on partial upfront payment option with 1-year or 3-year
commitment. This check is not available to accounts linked in Consolidated Billing.
Recommendations are only available for the Paying Account.",
    "id": "1qazXsw23e",
    "metadata": {
        "0": "Region",
        "1": "Family",
        "2": "Instance Type",
        "3": "License Model",
        "4": "Database Edition",
        "5": "Database Engine",
        "6": "Deployment Option",
        "7": "Recommended number of Reserved Instances to purchase",
        "8": "Expected Average Reserved Instance Utilization",
        "9": "Estimated Savings with Recommendation (monthly)"
        "10": "Upfront Cost of Reserved Instances",
        "11": "Estimated cost of Reserved Instances (monthly)",
        "12": "Estimated On-Demand Cost Post Recommended Reserved Instance
Purchase (monthly)",
        "13": "Estimated Break Even (months)",
        "14": "Lookback Period (days)",
        "15": "Term (years)"
    },
    "name": "Amazon Relational Database Service (RDS) Reserved Instance
Optimization",
    "pillars": [
        "cost_optimizing"
    ],
    "source": "ta_check"
},
{
    "arn": "arn:aws:trustedadvisor:::check/1qw23er45t",
    "awsServices": [
        "Redshift"
    ],
    "description": "Checks your usage of Redshift and provides
recommendations on purchase of Reserved Nodes to help reduce costs incurred from
using Redshift On-Demand. AWS generates these recommendations by analyzing your

```

On-Demand usage for the past 30 days. We then simulate every combination of reservations in the generated category of usage in order to identify the best number of each type of Reserved Nodes to purchase to maximize your savings. This check covers recommendations based on partial upfront payment option with 1-year or 3-year commitment. This check is not available to accounts linked in Consolidated Billing. Recommendations are only available for the Paying Account.",

```

    "id": "1qw23er45t",
    "metadata": {
      "0": "Region",
      "1": "Family",
      "2": "Node Type",
      "3": "Recommended number of Reserved Nodes to purchase",
      "4": "Expected Average Reserved Node Utilization",
      "5": "Estimated Savings with Recommendation (monthly)",
      "6": "Upfront Cost of Reserved Nodes",
      "7": "Estimated cost of Reserved Nodes (monthly)",
      "8": "Estimated On-Demand Cost Post Recommended Reserved Nodes
Purchase (monthly)",
      "9": "Estimated Break Even (months)",
      "10": "Lookback Period (days)",
      "11": "Term (years)",
    },
    "name": "Amazon Redshift Reserved Node Optimization",
    "pillars": [
      "cost_optimizing"
    ],
    "source": "ta_check"
  },
],
"nextToken": "REDACTED"
}

```

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

- Para API obtener más información, consulte [ListChecks](#) la Referencia de AWS CLI comandos.

## **list-organization-recommendation-accounts**

En el siguiente ejemplo de código, se muestra cómo usar `list-organization-recommendation-accounts`.



## AWS CLI

Para enumerar las cuentas de recomendación de la organización

En el siguiente `list-organization-recommendation-accounts` ejemplo, se enumeran todos los resúmenes de las recomendaciones de cuentas de una organización por su identificador.

```
aws trustedadvisor list-organization-recommendation-accounts \
  --organization-recommendation-identifier arn:aws:trustedadvisor::organization-recommendation/9534ec9b-bf3a-44e8-8213-2ed68b39d9d5
```

Salida:

```
{
  "accountRecommendationLifecycleSummaries": [{
    "accountId": "000000000000",
    "accountRecommendationArn":
    "arn:aws:trustedadvisor::000000000000:recommendation/9534ec9b-
    bf3a-44e8-8213-2ed68b39d9d5",
    "lifecycleStage": "resolved",
    "updateReason": "Resolved issue",
    "updateReasonCode": "valid_business_case",
    "lastUpdatedAt": "2023-01-17T18:25:44.552Z"
  }],
  "nextToken": "REDACTED"
}
```

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

- Para API obtener más información, consulte [ListOrganizationRecommendationAccounts](#) la Referencia de AWS CLI comandos.

## `list-organization-recommendation-resources`

En el siguiente ejemplo de código, se muestra cómo usar `list-organization-recommendation-resources`.

## AWS CLI

Para enumerar los recursos de recomendación de la organización

En el siguiente `list-organization-recommendation-resources` ejemplo, se enumeran todos los recursos de una recomendación de organización por su identificador.

```
aws trustedadvisor list-organization-recommendation-resources \
  --organization-recommendation-identifier arn:aws:trustedadvisor::organization-recommendation/5a694939-2e54-45a2-ae72-730598fa89d0
```

Salida:

```
{
  "organizationRecommendationResourceSummaries": [
    {
      "arn": "arn:aws:trustedadvisor::000000000000:recommendation-resource/5a694939-2e54-45a2-ae72-730598fa89d0/bb38affc0ce0681d9a6cd13f30238ba03a8f63dfe7a379dc403c619119d86af",
      "awsResourceId": "database-1-instance-1",
      "id": "bb38affc0ce0681d9a6cd13f302383ba03a8f63dfe7a379dc403c619119d86af",
      "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
      "metadata": {
        "0": "14",
        "1": "208.79999999999998",
        "2": "database-1-instance-1",
        "3": "db.r5.large",
        "4": "false",
        "5": "us-west-2",
        "6": "arn:aws:rds:us-west-2:000000000000:db:database-1-instance-1",
        "7": "1"
      },
      "recommendationArn": "arn:aws:trustedadvisor::organization-recommendation/5a694939-2e54-45a2-ae72-730598fa89d0",
      "regionCode": "us-west-2",
      "status": "warning"
    },
    {
      "arn": "arn:aws:trustedadvisor::000000000000:recommendation-resource/5a694939-2e54-45a2-ae72-730598fa89d0/51fded4d7a3278818df9cfe344ff5762cec46c095a6763d1ba1ba53bd0e1b0e6",
      "awsResourceId": "database-1",
      "id": "51fded4d7a3278818df9cfe344ff5762cec46c095a6763d1ba1ba53bd0e1b0e6",
      "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
      "metadata": {
```

```
        "0": "14",
        "1": "31.679999999999996",
        "2": "database-1",
        "3": "db.t3.small",
        "4": "false",
        "5": "us-west-2",
        "6": "arn:aws:rds:us-west-2:000000000000:db:database-1",
        "7": "20"
    },
    "recommendationArn": "arn:aws:trustedadvisor::organization-
recommendation/5a694939-2e54-45a2-ae72-730598fa89d0",
    "regionCode": "us-west-2",
    "status": "warning"
},
{
    "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
resource/5a694939-2e54-45a2-ae72-730598fa89d0/
f4d01bd20f4cd5372062aafc8786c489e48f0ead7cdab121463bf9f89e40a36b",
    "awsResourceId": "database-2-instance-1-us-west-2a",
    "id":
"f4d01bd20f4cd5372062aafc8786c489e48f0ead7cdab121463bf9f89e40a36b",
    "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
    "metadata": {
        "0": "14",
        "1": "187.200000000000002",
        "2": "database-2-instance-1-us-west-2a",
        "3": "db.r6g.large",
        "4": "true",
        "5": "us-west-2",
        "6": "arn:aws:rds:us-west-2:000000000000:db:database-2-instance-1-
us-west-2a",
        "7": "1"
    },
    "recommendationArn": "arn:aws:trustedadvisor::organization-
recommendation/5a694939-2e54-45a2-ae72-730598fa89d0",
    "regionCode": "us-west-2",
    "status": "warning"
},
],
"nextToken": "REDACTED"
}
```

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

- Para API obtener más información, consulte [ListOrganizationRecommendationResources](#) la Referencia de AWS CLI comandos.

## list-organization-recommendations

En el siguiente ejemplo de código, se muestra cómo usar `list-organization-recommendations`.

### AWS CLI

Ejemplo 1: Para enumerar las recomendaciones de la organización

En el siguiente `list-organization-recommendations` ejemplo se enumeran todas las recomendaciones de la organización y no se incluye ningún filtro.

```
aws trustedadvisor list-organization-recommendations
```

Salida:

```
{
  "organizationRecommendationSummaries": [
    {
      "arn": "arn:aws:trustedadvisor::organization-recommendation/9534ec9b-
bf3a-44e8-8213-2ed68b39d9d5",
      "name": "Lambda Runtime Deprecation Warning",
      "awsServices": [
        "lambda"
      ],
      "checkArn": "arn:aws:trustedadvisor::check/L4dfs2Q4C5",
      "id": "9534ec9b-bf3a-44e8-8213-2ed68b39d9d5",
      "lifecycleStage": "resolved",
      "pillars": [
        "security"
      ],
      "resourcesAggregates": {
        "errorCount": 0,
        "okCount": 0,
        "warningCount": 0
      }
    }
  ],
}
```

```

        "source": "ta_check",
        "status": "warning",
        "type": "priority"
    },
    {
        "arn": "arn:aws:trustedadvisor::organization-
recommendation/4ecff4d4-1bc1-4c99-a5b8-0fff9ee500d6",
        "name": "Lambda Runtime Deprecation Warning",
        "awsServices": [
            "lambda"
        ],
        "checkArn": "arn:aws:trustedadvisor::check/L4dfs2Q4C5",
        "id": "4ecff4d4-1bc1-4c99-a5b8-0fff9ee500d6",
        "lifecycleStage": "resolved",
        "pillars": [
            "security"
        ],
        "resourcesAggregates": {
            "errorCount": 0,
            "okCount": 0,
            "warningCount": 0
        },
        "source": "ta_check",
        "status": "warning",
        "type": "priority"
    },
],
"nextToken": "REDACTED"
}

```

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

Ejemplo 2: Para enumerar las recomendaciones de la organización con un filtro

El siguiente `list-organization-recommendations` ejemplo filtra y devuelve como máximo una recomendación de la organización que forma parte del pilar de la «seguridad».

```

aws trustedadvisor list-organization-recommendations \
  --pillar security \
  --max-items 100

```

Salida:

```
{
  "organizationRecommendationSummaries": [{
    "arn": "arn:aws:trustedadvisor::organization-recommendation/9534ec9b-
bf3a-44e8-8213-2ed68b39d9d5",
    "name": "Lambda Runtime Deprecation Warning",
    "awsServices": [
      "lambda"
    ],
    "checkArn": "arn:aws:trustedadvisor::check/L4dfs2Q4C5",
    "id": "9534ec9b-bf3a-44e8-8213-2ed68b39d9d5",
    "lifecycleStage": "resolved",
    "pillars": [
      "security"
    ],
    "resourcesAggregates": {
      "errorCount": 0,
      "okCount": 0,
      "warningCount": 0
    },
    "source": "ta_check",
    "status": "warning",
    "type": "priority"
  }],
  "nextToken": "REDACTED"
}
```

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

Ejemplo 3: Para enumerar las recomendaciones de la organización con un token de paginación

En el siguiente `list-organization-recommendations` ejemplo, se utiliza el `nextToken` «" devuelto por una solicitud anterior para buscar la siguiente página de recomendaciones de la organización.

```
aws trustedadvisor list-organization-recommendations \
  --pillar security \
  --max-items 100 \
  --starting-token <next-token>
```

Salida:

```
{
  "organizationRecommendationSummaries": [{
    "arn": "arn:aws:trustedadvisor::organization-
recommendation/4ecff4d4-1bc1-4c99-a5b8-0fff9ee500d6",
    "name": "Lambda Runtime Deprecation Warning",
    "awsServices": [
      "lambda"
    ],
    "checkArn": "arn:aws:trustedadvisor::check/L4dfs2Q4C5",
    "id": "4ecff4d4-1bc1-4c99-a5b8-0fff9ee500d6",
    "lifecycleStage": "resolved",
    "pillars": [
      "security"
    ],
    "resourcesAggregates": {
      "errorCount": 0,
      "okCount": 0,
      "warningCount": 0
    },
    "source": "ta_check",
    "status": "warning",
    "type": "priority"
  }]
}
```

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

- Para API obtener más información, consulte [ListOrganizationRecommendations](#) la Referencia de AWS CLI comandos.

## **list-recommendation-resources**

En el siguiente ejemplo de código, se muestra cómo usar `list-recommendation-resources`.

### AWS CLI

Para enumerar los recursos de recomendaciones

En el siguiente `list-recommendation-resources` ejemplo, se enumeran todos los recursos de una recomendación por su identificador.

```
aws trustedadvisor list-recommendation-resources \
  --recommendation-
  identifier arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
  bbb7-491a-833b-5773e9589578
```

Salida:

```
{
  "recommendationResourceSummaries": [
    {
      "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
      resource/55fa4d2e-
      bbb7-491a-833b-5773e9589578/18959a1f1973cff8e706e9d9bde28bba36cd602a6b2cb86c8b61252835236010",
      "id":
      "18959a1f1973cff8e706e9d9bde28bba36cd602a6b2cb86c8b61252835236010",
      "awsResourceId": "webcms-dev-01",
      "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
      "metadata": {
        "0": "14",
        "1": "123.120000000000002",
        "2": "webcms-dev-01",
        "3": "db.m6i.large",
        "4": "false",
        "5": "us-east-1",
        "6": "arn:aws:rds:us-east-1:000000000000:db:webcms-dev-01",
        "7": "20"
      },
      "recommendationArn":
      "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
      bbb7-491a-833b-5773e9589578",
      "regionCode": "us-east-1",
      "status": "warning"
    },
    {
      "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
      resource/55fa4d2e-bbb7-491a-833b-5773e9589578/
      e6367ff500ac90db8e4adeb4892e39ee9c36bbf812dcbce4b9e4fefcec9eb63e",
      "id":
      "e6367ff500ac90db8e4adeb4892e39ee9c36bbf812dcbce4b9e4fefcec9eb63e",
      "awsResourceId": "aws-dev-db-stack-instance-1",
      "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
      "metadata": {
        "0": "14",
```



```

        "1": "29.52",
        "2": "aws-dev-db-stack-instance-1",
        "3": "db.t2.small",
        "4": "false",
        "5": "us-east-1",
        "6": "arn:aws:rds:us-east-1:000000000000:db:aws-dev-db-stack-
instance-1",
        "7": "1"
    },
    "recommendationArn":
"arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
    "regionCode": "us-east-1",
    "status": "warning"
},
{
    "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
resource/55fa4d2e-
bbb7-491a-833b-5773e9589578/31aa78ba050a5015d2d38cca7f5f1ce88f70857c4e1c3ad03f8f9fd95dad7459",
    "id":
"31aa78ba050a5015d2d38cca7f5f1ce88f70857c4e1c3ad03f8f9fd95dad7459",
    "awsResourceId": "aws-awesome-apps-stack-db",
    "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
    "metadata": {
        "0": "14",
        "1": "114.48000000000002",
        "2": "aws-awesome-apps-stack-db",
        "3": "db.m6g.large",
        "4": "false",
        "5": "us-east-1",
        "6": "arn:aws:rds:us-east-1:000000000000:db:aws-awesome-apps-stack-
db",
        "7": "100"
    },
    "recommendationArn":
"arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
    "regionCode": "us-east-1",
    "status": "warning"
}
],
"nextToken": "REDACTED"
}

```

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

- Para API obtener más información, consulte [ListRecommendationResources](#) la Referencia de AWS CLI comandos.

## list-recommendations

En el siguiente ejemplo de código, se muestra cómo usar `list-recommendations`.

### AWS CLI

Ejemplo 1: Para enumerar las recomendaciones

En el `list-recommendations` ejemplo siguiente se enumeran todas las recomendaciones y no se incluye ningún filtro.

```
aws trustedadvisor list-recommendations
```

Salida:

```
{
  "recommendationSummaries": [
    {
      "arn": "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
      "name": "MFA Recommendation",
      "awsServices": [
        "iam"
      ],
      "checkArn": "arn:aws:trustedadvisor:::check/7DAFEmoDos",
      "id": "55fa4d2e-bbb7-491a-833b-5773e9589578",
      "lastUpdatedAt": "2023-11-01T15:57:58.673Z",
      "pillarSpecificAggregates": {
        "costOptimizing": {
          "estimatedMonthlySavings": 0.0,
          "estimatedPercentMonthlySavings": 0.0
        }
      },
      "pillars": [
        "security"
      ],
      "resourcesAggregates": {
```

```
        "errorCount": 1,
        "okCount": 0,
        "warningCount": 0
    },
    "source": "ta_check",
    "status": "error",
    "type": "standard"
},
{
    "arn":
"arn:aws:trustedadvisor::000000000000:recommendation/8b602b6f-452d-4cb2-8a9e-
c7650955d9cd",
    "name": "RDS clusters quota warning",
    "awsServices": [
        "rds"
    ],
    "checkArn": "arn:aws:trustedadvisor:::check/gjqMBn6pjz",
    "id": "8b602b6f-452d-4cb2-8a9e-c7650955d9cd",
    "lastUpdatedAt": "2023-11-01T15:58:17.397Z",
    "pillarSpecificAggregates": {
        "costOptimizing": {
            "estimatedMonthlySavings": 0.0,
            "estimatedPercentMonthlySavings": 0.0
        }
    },
    "pillars": [
        "service_limits"
    ],
    "resourcesAggregates": {
        "errorCount": 0,
        "okCount": 3,
        "warningCount": 6
    },
    "source": "ta_check",
    "status": "warning",
    "type": "standard"
}
],
"nextToken": "REDACTED"
}
```

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

## Ejemplo 2: Para enumerar las recomendaciones con un filtro

En el siguiente `list-recommendations` ejemplo, se enumeran las recomendaciones e incluye un filtro.

```
aws trustedadvisor list-recommendations \  
  --aws-service iam \  
  --max-items 100
```

Salida:

```
{  
  "recommendationSummaries": [{  
    "arn": "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-  
bbb7-491a-833b-5773e9589578",  
    "name": "MFA Recommendation",  
    "awsServices": [  
      "iam"  
    ],  
    "checkArn": "arn:aws:trustedadvisor:::check/7DAFEemoDos",  
    "id": "55fa4d2e-bbb7-491a-833b-5773e9589578",  
    "lastUpdatedAt": "2023-11-01T15:57:58.673Z",  
    "pillarSpecificAggregates": {  
      "costOptimizing": {  
        "estimatedMonthlySavings": 0.0,  
        "estimatedPercentMonthlySavings": 0.0  
      }  
    },  
    "pillars": [  
      "security"  
    ],  
    "resourcesAggregates": {  
      "errorCount": 1,  
      "okCount": 0,  
      "warningCount": 0  
    },  
    "source": "ta_check",  
    "status": "error",  
    "type": "standard"  
  }],  
  "nextToken": "REDACTED"  
}
```

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

Ejemplo 3: Para enumerar las recomendaciones con un token de paginación

En el siguiente `list-recommendations` ejemplo, se utiliza el `nextToken` «" devuelto por una solicitud anterior para buscar la siguiente página de recomendaciones filtradas.

```
aws trustedadvisor list-recommendations \  
  --aws-service rds \  
  --max-items 100 \  
  --starting-token <next-token>
```

Salida:

```
{  
  "recommendationSummaries": [{  
    "arn":  
    "arn:aws:trustedadvisor::000000000000:recommendation/8b602b6f-452d-4cb2-8a9e-  
c7650955d9cd",  
    "name": "RDS clusters quota warning",  
    "awsServices": [  
      "rds"  
    ],  
    "checkArn": "arn:aws:trustedadvisor:::check/gjqMBn6pjz",  
    "id": "8b602b6f-452d-4cb2-8a9e-c7650955d9cd",  
    "lastUpdatedAt": "2023-11-01T15:58:17.397Z",  
    "pillarSpecificAggregates": {  
      "costOptimizing": {  
        "estimatedMonthlySavings": 0.0,  
        "estimatedPercentMonthlySavings": 0.0  
      }  
    },  
    "pillars": [  
      "service_limits"  
    ],  
    "resourcesAggregates": {  
      "errorCount": 0,  
      "okCount": 3,  
      "warningCount": 6  
    },  
    "source": "ta_check",  
    "status": "warning",
```

```
    "type": "standard"  
  }]  
}
```

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

- Para API obtener más información, consulte [ListRecommendations](#) la Referencia de AWS CLI comandos.

## update-organization-recommendation-lifecycle

En el siguiente ejemplo de código, se muestra cómo usar `update-organization-recommendation-lifecycle`.

### AWS CLI

Para actualizar el ciclo de vida de las recomendaciones de una organización

En el siguiente `update-organization-recommendation-lifecycle` ejemplo, se actualiza el ciclo de vida de una recomendación de la organización mediante su identificador.

```
aws trustedadvisor update-organization-recommendation-lifecycle \  
  --organization-recommendation-identifier arn:aws:trustedadvisor::organization-  
recommendation/96b5e5ca-7930-444c-90c6-06d386128100 \  
  --lifecycle-stage dismissed \  
  --update-reason-code not_applicable
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

- Para API obtener más información, consulte [UpdateOrganizationRecommendationLifecycle](#) la Referencia de AWS CLI comandos.

## update-recommendation-lifecycle

En el siguiente ejemplo de código, se muestra cómo usar `update-recommendation-lifecycle`.

## AWS CLI

Para actualizar el ciclo de vida de una recomendación

En el siguiente `update-recommendation-lifecycle` ejemplo, se actualiza el ciclo de vida de una recomendación por su identificador.

```
aws trustedadvisor update-recommendation-lifecycle \  
  --recommendation-  
  identifier arn:aws:trustedadvisor::000000000000:recommendation/861c9c6e-  
  f169-405a-8b59-537a8cacc7a \  
  --lifecycle-stage resolved \  
  --update-reason-code valid_business_case
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Comenzar con el Trusted Advisor API](#) en la Guía del usuario de AWS Trusted Advisor.

- Para API obtener más información, consulte [UpdateRecommendationLifecycle](#) la Referencia de AWS CLI comandos.

## Ejemplos de permisos verificados mediante AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante el uso de permisos AWS Command Line Interface verificados.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **create-identity-source**

En el siguiente ejemplo de código, se muestra cómo usar `create-identity-source`.

#### AWS CLI

Para crear una fuente de identidad

El siguiente `create-identity-source` ejemplo crea una fuente de identidad que le permite hacer referencia a las identidades almacenadas en el grupo de usuarios de Amazon Cognito especificado. Esas identidades están disponibles en los permisos verificados como entidades de tipo `User`.

```
aws verifiedpermissions create-identity-source \  
  --configuration file://config.txt \  
  --principal-entity-type "User" \  
  --policy-store-id PSEXAMPLEabcdefgh111111
```

Contenidos de `config.txt`:

```
{  
  "cognitoUserPoolConfiguration": {  
    "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-west-2_1a2b3c4d5",  
    "clientIds":["a1b2c3d4e5f6g7h8i9j0kalbmc"]  
  }  
}
```

Salida:

```
{  
  "createdDate": "2023-05-19T20:30:28.214829+00:00",  
  "identitySourceId": "ISEXAMPLEabcdefgh111111",  
  "lastUpdatedDate": "2023-05-19T20:30:28.214829+00:00",  
  "policyStoreId": "PSEXAMPLEabcdefgh111111"  
}
```

Para obtener más información sobre las fuentes de identidad, consulte [Uso de permisos verificados de Amazon con proveedores de identidad](#) en la Guía del usuario de permisos verificados de Amazon.



- Para API obtener más información, consulte [CreateIdentitySource](#) la Referencia de AWS CLI comandos.

## create-policy-store

En el siguiente ejemplo de código, se muestra cómo usar `create-policy-store`.

### AWS CLI

Para crear un almacén de políticas

En el siguiente `create-policy-store` ejemplo, se crea un almacén de políticas en la AWS región actual.

```
aws verifiedpermissions create-policy-store \  
  --validation-settings "mode=STRICT"
```

Salida:

```
{  
  "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/  
PSEXAMPLEEabcdefg111111",  
  "createdDate": "2023-05-16T17:41:29.103459+00:00",  
  "lastUpdatedDate": "2023-05-16T17:41:29.103459+00:00",  
  "policyStoreId": "PSEXAMPLEEabcdefg111111"  
}
```

Para obtener más información sobre los almacenes de políticas, consulta los [almacenes de políticas de permisos verificados](#) de Amazon en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [CreatePolicyStore](#) la Referencia de AWS CLI comandos.

## create-policy-template

En el siguiente ejemplo de código, se muestra cómo usar `create-policy-template`.

### AWS CLI

Ejemplo 1: Para crear una plantilla de políticas

En el siguiente `create-policy-template` ejemplo, se crea una plantilla de política con una declaración que contiene un marcador de posición para el director.

```
aws verifiedpermissions create-policy-template \  
  --definition file://template1.txt \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

Contenido del archivo `template1.txt`:

```
permit(  
  principal in ?principal,  
  action == Action::"view",  
  resource == Photo::"VacationPhoto94.jpg"  
);
```

Salida:

```
{  
  "createdDate": "2023-06-12T20:47:42.804511+00:00",  
  "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",  
  "policyStoreId": "PSEXAMPLEabcdefg111111",  
  "policyTemplateId": "PTEXAMPLEabcdefg111111"  
}
```

Para obtener más información sobre las plantillas de políticas, consulte las [plantillas de políticas de permisos verificados](#) de Amazon en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [CreatePolicyTemplate](#) la Referencia de AWS CLI comandos.

## **create-policy**

En el siguiente ejemplo de código, se muestra cómo usar `create-policy`.

### AWS CLI

Ejemplo 1: Para crear una política estática

El siguiente `create-policy` ejemplo crea una política estática con un alcance de política que especifica tanto un principal como un recurso.

```
aws verifiedpermissions create-policy \  
  --definition file://definition1.txt \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

Contenido del archivo definition1.txt:

```
{  
  "static": {  
    "description": "Grant everyone of janeFriends UserGroup access to the  
vacationFolder Album",  
    "statement": "permit(principal in UserGroup::\"janeFriends\", action,  
resource in Album::\"vacationFolder\" );"  
  }  
}
```

Salida:

```
{  
  "createdDate": "2023-06-12T20:33:37.382907+00:00",  
  "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",  
  "policyId": "SPEXAMPLEabcdefg111111",  
  "policyStoreId": "PSEXAMPLEabcdefg111111",  
  "policyType": "STATIC",  
  "principal": {  
    "entityId": "janeFriends",  
    "entityType": "UserGroup"  
  },  
  "resource": {  
    "entityId": "vacationFolder",  
    "entityType": "Album"  
  }  
}
```

Ejemplo 2: Crear una política estática que conceda acceso a un recurso a todos

El siguiente create-policy ejemplo crea una política estática con un alcance de política que especifica solo un recurso.

```
aws verifiedpermissions create-policy \  
  --definition file://definition2.txt \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

### Contenido del archivo definition2.txt:

```
{
  "static": {
    "description": "Grant everyone access to the publicFolder Album",
    "statement": "permit(principal, action, resource in Album:\""publicFolder
  \");"
  }
}
```

### Salida:

```
{
  "createdDate": "2023-06-12T20:39:44.975897+00:00",
  "lastUpdatedDate": "2023-06-12T20:39:44.975897+00:00",
  "policyId": "PbfR73F8oh5MMfr9uRtFDB",
  "policyStoreId": "PSEXAMPLEEabcdefg222222",
  "policyType": "STATIC",
  "resource": {
    "entityId": "publicFolder",
    "entityType": "Album"
  }
}
```

Ejemplo 3: Para crear una política vinculada a una plantilla que esté asociada a la plantilla especificada

En el siguiente `create-policy` ejemplo, se crea una política vinculada a una plantilla mediante la plantilla de política especificada y se asocia el principio especificado que se va a utilizar con la nueva política vinculada a la plantilla.

```
aws verifiedpermissions create-policy \
  --definition file://definition.txt \
  --policy-store-id PSEXAMPLEEabcdefg111111
```

### Contenidos de definition.txt:

```
{
  "templateLinked": {
    "policyTemplateId": "PTEXAMPLEEabcdefg111111",
    "principal": {
      "entityType": "User",
```

```

        "entityId": "alice"
      }
    }
  }
}

```

Salida:

```

{
  "createdDate": "2023-06-12T20:49:51.490211+00:00",
  "lastUpdatedDate": "2023-06-12T20:49:51.490211+00:00",
  "policyId": "TPEXAMPLEabcdefg111111",
  "policyStoreId": "PSEXAMPLEabcdefg111111",
  "policyType": "TEMPLATE_LINKED",
  "principal": {
    "entityId": "alice",
    "entityType": "User"
  },
  "resource": {
    "entityId": "VacationPhoto94.jpg",
    "entityType": "Photo"
  }
}

```

Para obtener más información sobre las políticas, consulta las políticas de [permisos verificados de Amazon en la Guía del](#) usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [CreatePolicy](#) la Referencia de AWS CLI comandos.

## delete-identity-source

En el siguiente ejemplo de código, se muestra cómo usar delete-identity-source.

### AWS CLI

Para eliminar una fuente de identidad

En el siguiente delete-identity-source ejemplo, se elimina la fuente de identidad que tiene el identificador especificado.

```

aws verifiedpermissions delete-identity-source \
  --identity-source-id ISEXAMPLEabcdefg111111 \
  --policy-store-id PSEXAMPLEabcdefg111111

```

Este comando no genera ninguna salida.

Para obtener más información sobre las fuentes de identidad, consulte [Uso de permisos verificados de Amazon con proveedores de identidad](#) en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [DeleteIdentitySource](#) la Referencia de AWS CLI comandos.

## **delete-policy-store**

En el siguiente ejemplo de código, se muestra cómo usar `delete-policy-store`.

### AWS CLI

Para eliminar un almacén de políticas

En el siguiente `delete-policy-store` ejemplo, se elimina el almacén de políticas que tiene el identificador especificado.

```
aws verifiedpermissions delete-policy-store \  
--policy-store-id PSEXAMPLEabcdefg111111
```

Este comando no genera ninguna salida.

Para obtener más información sobre los almacenes de políticas, consulta los [almacenes de políticas de permisos verificados](#) de Amazon en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [DeletePolicyStore](#) la Referencia de AWS CLI comandos.

## **delete-policy-template**

En el siguiente ejemplo de código, se muestra cómo usar `delete-policy-template`.

### AWS CLI

Para eliminar una plantilla de política

En el siguiente `delete-policy-template` ejemplo, se elimina la plantilla de política que tiene el identificador especificado.

```
aws verifiedpermissions delete-policy \  
  --policy-template-id PEXAMPLEabcdefg111111 \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

Este comando no genera ninguna salida.

Para obtener más información sobre las plantillas de políticas, consulte las [plantillas de políticas de permisos verificados](#) de Amazon en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [DeletePolicyTemplate](#) la Referencia de AWS CLI comandos.

## delete-policy

En el siguiente ejemplo de código, se muestra cómo usar delete-policy.

### AWS CLI

Para eliminar una política estática o vinculada a una plantilla

En el siguiente delete-policy ejemplo, se elimina la política que tiene el identificador especificado.

```
aws verifiedpermissions delete-policy \  
  --policy-id SPEXAMPLEabcdefg111111 \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

Este comando no genera ninguna salida.

Para obtener más información sobre las políticas, consulta las políticas de [permisos verificados de Amazon en la Guía del](#) usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [DeletePolicy](#) la Referencia de AWS CLI comandos.

## get-identity-source

En el siguiente ejemplo de código, se muestra cómo usar get-identity-source.

### AWS CLI

Para recuperar detalles sobre una fuente de identidad

En el siguiente `get-identity-source` ejemplo, se muestran los detalles de la fuente de identidad con el identificador especificado.

```
aws verifiedpermissions get-identity-source \  
  --identity-source IEXAMPLEabcdefgh111111 \  
  --policy-store-id PEXAMPLEabcdefgh111111
```

Salida:

```
{  
  "createdDate": "2023-06-12T22:27:49.150035+00:00",  
  "details": {  
    "clientIds": [ "a1b2c3d4e5f6g7h8i9j0kalbmc" ],  
    "discoveryUrl": "https://cognito-idp.us-west-2.amazonaws.com/us-west-2_1a2b3c4d5",  
    "openIdIssuer": "COGNITO",  
    "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-west-2_1a2b3c4d5"  
  },  
  "identitySourceId": "IEXAMPLEabcdefgh111111",  
  "lastUpdatedDate": "2023-06-12T22:27:49.150035+00:00",  
  "policyStoreId": "PEXAMPLEabcdefgh111111",  
  "principalEntityType": "User"  
}
```

Para obtener más información sobre las fuentes de identidad, consulte [Uso de permisos verificados de Amazon con proveedores de identidad](#) en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [GetIdentitySource](#) la Referencia de AWS CLI comandos.

## get-policy-store

En el siguiente ejemplo de código, se muestra cómo usar `get-policy-store`.

### AWS CLI

Para recuperar detalles sobre un almacén de políticas

En el siguiente `get-policy-store` ejemplo, se muestran los detalles del almacén de políticas con el identificador especificado.



```
aws verifiedpermissions get-policy-store \  
  --policy-store-id PSEXAMPLEEabcdefg111111
```

Salida:

```
{  
  "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/  
PSEXAMPLEEabcdefg111111",  
  "createdDate": "2023-06-05T20:16:46.225598+00:00",  
  "lastUpdatedDate": "2023-06-08T20:40:23.173691+00:00",  
  "policyStoreId": "PSEXAMPLEEabcdefg111111",  
  "validationSettings": { "mode": "OFF" }  
}
```

Para obtener más información sobre los almacenes de políticas, consulta los [almacenes de políticas de permisos verificados](#) de Amazon en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [GetPolicyStore](#) la Referencia de AWS CLI comandos.

## get-policy-template

En el siguiente ejemplo de código, se muestra cómo usar `get-policy-template`.

AWS CLI

Para recuperar detalles sobre una plantilla de política

En el siguiente `get-policy-template` ejemplo, se muestran los detalles de la plantilla de política con el ID especificado.

```
aws verifiedpermissions get-policy-template \  
  --policy-template-id PTEXAMPLEEabcdefg111111 \  
  --policy-store-id PSEXAMPLEEabcdefg111111
```

Salida:

```
{  
  "createdDate": "2023-06-12T20:47:42.804511+00:00",
```

```

    "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",
    "policyStoreId": "PSEXAMPLEEabcdefg111111",
    "policyTemplateId": "PTEXAMPLEEabcdefg111111",
    "statement": "permit(\n    principal in ?principal,\n    action == Action::\n    \\"view\\",\n    resource == Photo::\\"VacationPhoto94.jpg\\"\\n);"
  }

```

Para obtener más información sobre las plantillas de políticas, consulte las [plantillas de políticas de permisos verificados](#) de Amazon en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [GetPolicyTemplate](#) la Referencia de AWS CLI comandos.

## get-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-policy`.

### AWS CLI

Para recuperar detalles sobre una política

En el siguiente `get-policy` ejemplo, se muestran los detalles de la política con el ID especificado.

```

aws verifiedpermissions get-policy \
  --policy-id PSEXAMPLEEabcdefg111111 \
  --policy-store-id PSEXAMPLEEabcdefg111111

```

Salida:

```

{
  "createdDate": "2023-06-12T20:33:37.382907+00:00",
  "definition": {
    "static": {
      "description": "Grant everyone of janeFriends UserGroup access to the
vacationFolder Album",
      "statement": "permit(principal in UserGroup::\\"janeFriends\\", action,
resource in Album::\\"vacationFolder\\" );"
    }
  },
  "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
  "policyId": "SPEXAMPLEEabcdefg111111",

```

```

    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyType": "STATIC",
    "principal": {
      "entityId": "janeFriends",
      "entityType": "UserGroup"
    },
    "resource": {
      "entityId": "vacationFolder",
      "entityType": "Album"
    }
  }
}

```

Para obtener más información sobre las políticas, consulta las políticas de [permisos verificados de Amazon en la Guía del](#) usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [GetPolicy](#) la Referencia de AWS CLI comandos.

## get-schema

En el siguiente ejemplo de código, se muestra cómo usar `get-schema`.

### AWS CLI

Para recuperar el esquema en un almacén de políticas

En el siguiente `get-schema` ejemplo, se muestran los detalles del esquema en el almacén de políticas especificado.

```

aws verifiedpermissions get-schema \
  --policy-store-id PSEXAMPLEabcdefg111111

```

Salida:

```

{
  "policyStoreId": "PSEXAMPLEabcdefg111111",
  "schema": "{\"MySampleNamespace\":{\"entityTypes\":{\"Employee\":{\"shape\":"
  "\":{\"attributes\":{\"jobLevel\":{\"type\":\"Long\"},\"name\":{\"type\":\"String\":"
  "\"}\",\"type\":\"Record\"}}},\"actions\":{\"remoteAccess\":{\"appliesTo\":{"
  "\"principalTypes\":[\"Employee\"]}}}}}",
  "createdDate": "2023-06-14T17:47:13.999885+00:00",
  "lastUpdatedDate": "2023-06-14T17:47:13.999885+00:00"
}

```

Para obtener más información sobre el esquema, consulte el [esquema del almacén de políticas](#) en la Guía del usuario de Amazon Verified Permissions.

- Para API obtener más información, consulte [GetSchema](#) la Referencia de AWS CLI comandos.

## **is-authorized-with-token**

En el siguiente ejemplo de código, se muestra cómo usar `is-authorized-with-token`.

### AWS CLI

Ejemplo 1: solicitar una decisión de autorización para una solicitud de usuario (permitir)

En el siguiente `is-authorized-with-token` ejemplo, se solicita una decisión de autorización para un usuario autenticado por Amazon Cognito. La solicitud utiliza el token de identidad proporcionado por Cognito en lugar del token de acceso. En este ejemplo, el almacén de información especificado está configurado para devolver los principales como entidades de tipo `CognitoUser`

```
aws verifiedpermissions is-authorized-with-token \  
  --action actionId="View",actionType="Action" \  
  --resource entityId="vacationPhoto94.jpg",entityType="Photo" \  
  --policy-store-id PSEXAMPLEabcdefg111111 \  
  --identity-token "AbCdE12345...long.string...54321EdCbA"
```

El almacén de políticas contiene una política con la siguiente declaración que acepta identidades del grupo de usuarios y del identificador de aplicación de Cognito especificados.

```
permit(  
  principal == CognitoUser::"us-east-1_1a2b3c4d5|a1b2c3d4e5f6g7h8i9j0kalbmc",  
  action,  
  resource == Photo::"VacationPhoto94.jpg"  
);
```

Salida:

```
{  
  "decision": "Allow",  
  "determiningPolicies": [  
    {  
      "determiningPolicyId": "SPEXAMPLEabcdefg111111"  
    }  
  ]  
}
```

```

    }
  ],
  "errors": []
}

```

Para obtener más información sobre el uso de identidades de un grupo de usuarios de Cognito, consulte Uso de [permisos verificados de Amazon con proveedores de identidad](#) en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [IsAuthorizedWithToken](#) la Referencia de AWS CLI comandos.

## is-authorized

En el siguiente ejemplo de código, se muestra cómo usar `is-authorized`.

### AWS CLI

Ejemplo 1: solicitar una decisión de autorización para una solicitud de usuario (permitir)

El siguiente `is-authorized` ejemplo solicita una decisión de autorización para un principal del tipo `User` denominado `Alice`, que desea realizar la `updatePhoto` operación, en un recurso del tipo `Photo` denominado `VacationPhoto94.jpg`.

La respuesta muestra que la solicitud está permitida por una política.

```

aws verifiedpermissions is-authorized \
  --principal entityType=User,entityId=alice \
  --action actionType=Action,actionId=view \
  --resource entityType=Photo,entityId=VactionPhoto94.jpg \
  --policy-store-id PSEXAMPLEabcdefg111111

```

Salida:

```

{
  "decision": "ALLOW",
  "determiningPolicies": [
    {
      "policyId": "SPEXAMPLEabcdefg111111"
    }
  ],
  "errors": []
}

```

```
}
```

Ejemplo 2: solicitar una decisión de autorización para una solicitud de un usuario (denegar)

El siguiente ejemplo es el mismo que el anterior, excepto que el principal sí lo es `User::"Bob"`. El almacén de políticas no contiene ninguna política que permita a ese usuario acceder a `Album::"alice_folder"`.

El resultado indica que Deny estaba implícito porque la lista de `DeterminingPolicies` está vacía.

```
aws verifiedpermissions create-policy \  
  --definition file://definition2.txt \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

Salida:

```
{  
  "decision": "DENY",  
  "determiningPolicies": [],  
  "errors": []  
}
```

Para obtener más información, consulta la [Guía del usuario de permisos verificados de Amazon](#).

- Para API obtener más información, consulte [IsAuthorized](#) la Referencia de AWS CLI comandos.

## list-identity-sources

En el siguiente ejemplo de código, se muestra cómo usar `list-identity-sources`.

AWS CLI

Para enumerar las fuentes de identidad disponibles

En el siguiente `list-identity-sources` ejemplo, se enumeran todas las fuentes de identidad del almacén de políticas especificado.

```
aws verifiedpermissions list-identity-sources \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

**Salida:**

```
{
  "identitySources": [
    {
      "createdDate": "2023-06-12T22:27:49.150035+00:00",
      "details": {
        "clientIds": [ "a1b2c3d4e5f6g7h8i9j0kalbmc" ],
        "discoveryUrl": "https://cognito-idp.us-west-2.amazonaws.com/us-west-2_1a2b3c4d5",
        "openIdIssuer": "COGNITO",
        "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-west-2_1a2b3c4d5"
      },
      "identitySourceId": "ISEXAMPLEabcdefg111111",
      "lastUpdatedDate": "2023-06-12T22:27:49.150035+00:00",
      "policyStoreId": "PSEXAMPLEabcdefg111111",
      "principalEntityType": "User"
    }
  ]
}
```

Para obtener más información sobre las fuentes de identidad, consulte [Uso de permisos verificados de Amazon con proveedores de identidad](#) en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [ListIdentitySources](#) la Referencia de AWS CLI comandos.

**list-policies**

En el siguiente ejemplo de código, se muestra cómo usar `list-policies`.

**AWS CLI**

Para ver una lista de las políticas disponibles

En el siguiente `list-policies` ejemplo, se enumeran todas las políticas del almacén de políticas especificado.

```
aws verifiedpermissions list-policies \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

**Salida:**

```
{
  "policies": [
    {
      "createdDate": "2023-06-12T20:33:37.382907+00:00",
      "definition": {
        "static": {
          "description": "Grant everyone of janeFriends UserGroup access
to the vacationFolder Album"
        }
      },
      "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
      "policyId": "SPEXAMPLEabcdefghijklmnop111111",
      "policyStoreId": "PSEXAMPLEabcdefghijklmnop111111",
      "policyType": "STATIC",
      "principal": {
        "entityId": "janeFriends",
        "entityType": "UserGroup"
      },
      "resource": {
        "entityId": "vacationFolder",
        "entityType": "Album"
      }
    },
    {
      "createdDate": "2023-06-12T20:39:44.975897+00:00",
      "definition": {
        "static": {
          "description": "Grant everyone access to the publicFolder Album"
        }
      },
      "lastUpdatedDate": "2023-06-12T20:39:44.975897+00:00",
      "policyId": "SPEXAMPLEabcdefghijklmnop222222",
      "policyStoreId": "PSEXAMPLEabcdefghijklmnop111111",
      "policyType": "STATIC",
      "resource": {
        "entityId": "publicFolder",
        "entityType": "Album"
      }
    },
    {
      "createdDate": "2023-06-12T20:49:51.490211+00:00",
      "definition": {
```



```
        "templateLinked": {
            "policyTemplateId": "PTEXAMPLEabcdefghijklmnop111111"
        }
    },
    "lastUpdatedDate": "2023-06-12T20:49:51.490211+00:00",
    "policyId": "SPEXAMPLEabcdefghijklmnop333333",
    "policyStoreId": "PSEXAMPLEabcdefghijklmnop111111",
    "policyType": "TEMPLATE_LINKED",
    "principal": {
        "entityId": "alice",
        "entityType": "User"
    },
    "resource": {
        "entityId": "VacationPhoto94.jpg",
        "entityType": "Photo"
    }
}
]
```

Para obtener más información sobre las políticas, consulta las políticas de [permisos verificados de Amazon en la Guía del](#) usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [ListPolicies](#)la Referencia de AWS CLI comandos.

## list-policy-stores

En el siguiente ejemplo de código, se muestra cómo usar `list-policy-stores`.

### AWS CLI

Para ver una lista de los almacenes de políticas disponibles

En el siguiente `list-policy-stores` ejemplo, se enumeran todos los almacenes de políticas de la AWS región. Todos los comandos de permisos `create-policy-store` verificados, excepto los que `list-policy-stores` requieren que especifique el identificador del almacén de políticas con el que desea trabajar.

```
aws verifiedpermissions list-policy-stores
```

Salida:

```
{
  "policyStores": [
    {
      "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEEabcdefg111111",
      "createdDate": "2023-06-05T20:16:46.225598+00:00",
      "policyStoreId": "PSEXAMPLEEabcdefg111111"
    },
    {
      "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEEabcdefg222222",
      "createdDate": "2023-06-08T18:09:37.364356+00:00",
      "policyStoreId": "PSEXAMPLEEabcdefg222222"
    },
    {
      "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEEabcdefg333333",
      "createdDate": "2023-06-08T18:09:46.920600+00:00",
      "policyStoreId": "PSEXAMPLEEabcdefg333333"
    }
  ]
}
```

Para obtener más información sobre los almacenes de políticas, consulta los [almacenes de políticas de permisos verificados](#) de Amazon en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [ListPolicyStores](#) la Referencia de AWS CLI comandos.

## list-policy-templates

En el siguiente ejemplo de código, se muestra cómo usar `list-policy-templates`.

### AWS CLI

Para ver una lista de las plantillas de políticas disponibles

En el siguiente `list-policy-templates` ejemplo, se enumeran todas las plantillas de políticas del almacén de políticas especificado.

```
aws verifiedpermissions list-policy-templates \
```

```
--policy-store-id PSEXAMPLEEabcdefg111111
```

Salida:

```
{
  "policyTemplates": [
    {
      "createdDate": "2023-06-12T20:47:42.804511+00:00",
      "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",
      "policyStoreId": "PSEXAMPLEEabcdefg111111",
      "policyTemplateId": "PTEXAMPLEEabcdefg111111"
    }
  ]
}
```

Para obtener más información sobre las plantillas de políticas, consulte las [plantillas de políticas de permisos verificados](#) de Amazon en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [ListPolicyTemplates](#) la Referencia de AWS CLI comandos.

## put -schema

En el siguiente ejemplo de código, se muestra cómo usar put -schema.

### AWS CLI

Para guardar un esquema en un almacén de políticas

En el siguiente put -schema ejemplo, se crea o reemplaza el esquema en el almacén de políticas especificado.

El cedarJson parámetro del archivo de entrada toma una representación en cadena de un JSON objeto. Contiene comillas incrustadas («») dentro del par de comillas más externas. Para ello, debe convertir la JSON cadena en una cadena poniendo una barra invertida (") delante de todas las comillas incrustadas y combinando todas las líneas en una sola línea de texto sin saltos de línea.

Las cadenas de ejemplo se pueden mostrar agrupadas entre varias líneas para facilitar la lectura, pero la operación requiere que los parámetros se envíen como cadenas de una sola línea.

```
aws verifiedpermissions put-schema --fichero de definición: //schema.txt -- policy-store-id
PSEXAMPLEabcdefg111111
```

Contenidos de `schema.txt`:

```
{
  "cedarJson": "{\"MySampleNamespace\": {\"actions\": {\"remoteAccess\": {
    \"appliesTo\": {\"principalTypes\": [\"Employee\"]}},\"entityTypes\": {
    \"Employee\": {\"shape\": {\"attributes\": {\"jobLevel\": {\"type\":
    \"Long\"}},\"name\": {\"type\": \"String\"}},\"type\": \"Record\"}}}}}"
}
```

Salida:

```
{
  "policyStoreId": "PSEXAMPLEabcdefg111111",
  "namespaces": [
    "MySampleNamespace"
  ],
  "createdDate": "2023-06-14T17:47:13.999885+00:00",
  "lastUpdatedDate": "2023-06-14T17:47:13.999885+00:00"
}
```

Para obtener más información sobre el esquema, consulte el [esquema del almacén de políticas](#) en la Guía del usuario de Amazon Verified Permissions.

- Para API obtener más información, consulte [PutSchema](#) la Referencia de AWS CLI comandos.

## update-identity-source

En el siguiente ejemplo de código, se muestra cómo usar `update-identity-source`.

### AWS CLI

Para actualizar una fuente de identidad

El siguiente `update-identity-source` ejemplo modifica la fuente de identidad especificada proporcionando una nueva configuración de grupo de usuarios de Cognito y cambiando el tipo de entidad devuelto por la fuente de identidad.

```
aws verifiedpermissions update-identity-source
```

```
--identity-source-id ISEXAMPLEabcdefg111111 \  
--update-configuration file://config.txt \  
--principal-entity-type "Employee" \  
--policy-store-id PSEXAMPLEabcdefg111111
```

Contenidos de config.txt:

```
{  
  "cognitoUserPoolConfiguration": {  
    "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/  
us-west-2_1a2b3c4d5",  
    "clientIds":["a1b2c3d4e5f6g7h8i9j0kalbmc"]  
  }  
}
```

Salida:

```
{  
  "createdDate": "2023-05-19T20:30:28.214829+00:00",  
  "identitySourceId": "ISEXAMPLEabcdefg111111",  
  "lastUpdatedDate": "2023-05-19T20:30:28.214829+00:00",  
  "policyStoreId": "PSEXAMPLEabcdefg111111"  
}
```

Para obtener más información sobre las fuentes de identidad, consulte [Uso de permisos verificados de Amazon con proveedores de identidad](#) en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [UpdateIdentitySource](#) la Referencia de AWS CLI comandos.

## update-policy-store

En el siguiente ejemplo de código, se muestra cómo usar update-policy-store.

### AWS CLI

Para actualizar un almacén de políticas

En el siguiente update-policy-store ejemplo, se modifica un almacén de políticas cambiando su configuración de validación.

```
aws verifiedpermissions update-policy-store \  
  --validation-settings "mode=STRICT" \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

Salida:

```
{  
  "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/  
PSEXAMPLEabcdefg111111",  
  "createdDate": "2023-05-16T17:41:29.103459+00:00",  
  "lastUpdatedDate": "2023-05-16T17:41:29.103459+00:00",  
  "policyStoreId": "PSEXAMPLEabcdefg111111"  
}
```

Para obtener más información sobre los almacenes de políticas, consulta los [almacenes de políticas de permisos verificados](#) de Amazon en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [UpdatePolicyStore](#) la Referencia de AWS CLI comandos.

## update-policy-template

En el siguiente ejemplo de código, se muestra cómo usar `update-policy-template`.

AWS CLI

Ejemplo 1: Para actualizar una plantilla de política

El siguiente `update-policy-template` ejemplo modifica la política vinculada a la plantilla especificada para reemplazar su declaración de política.

```
aws verifiedpermissions update-policy-template \  
  --policy-template-id PTEXAMPLEabcdefg111111 \  
  --statement file://template1.txt \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

Contenido del archivo `template1.txt`:

```
permit(  
  principal in ?principal,
```

```
    action == Action::"view",
    resource == Photo::"VacationPhoto94.jpg"
);
```

Salida:

```
{
  "createdDate": "2023-06-12T20:47:42.804511+00:00",
  "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",
  "policyStoreId": "PSEXAMPLEabcdefg111111",
  "policyTemplateId": "PTEXAMPLEabcdefg111111"
}
```

Para obtener más información sobre las plantillas de políticas, consulte las [plantillas de políticas de permisos verificados](#) de Amazon en la Guía del usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [UpdatePolicyTemplate](#) la Referencia de AWS CLI comandos.

## update-policy

En el siguiente ejemplo de código, se muestra cómo usar `update-policy`.

### AWS CLI

Ejemplo 1: Para crear una política estática

El siguiente `create-policy` ejemplo crea una política estática con un alcance de política que especifica tanto un principal como un recurso.

```
aws verifiedpermissions create-policy \  
  --definition file://definition.txt \  
  --policy-store-id PSEXAMPLEabcdefg111111
```

El `statement` parámetro toma una representación en cadena de un JSON objeto. Contiene comillas incrustadas («) dentro del par de comillas más externas. Para ello, debe convertir la JSON cadena en una cadena poniendo una barra invertida (") delante de todas las comillas incrustadas y combinando todas las líneas en una sola línea de texto sin saltos de línea.

Las cadenas de ejemplo se pueden mostrar agrupadas entre varias líneas para facilitar la lectura, pero la operación requiere que los parámetros se envíen como cadenas de una sola línea.

### Contenido del archivo `definition.txt`:

```
{
  "static": {
    "description": "Grant everyone of janeFriends UserGroup access to the
vacationFolder Album",
    "statement": "permit(principal in UserGroup::\"janeFriends\", action,
resource in Album::\"vacationFolder\" );"
  }
}
```

### Salida:

```
{
  "createdDate": "2023-06-12T20:33:37.382907+00:00",
  "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
  "policyId": "SPEXAMPLEabcdefg111111",
  "policyStoreId": "PSEXAMPLEabcdefg111111",
  "policyType": "STATIC",
  "principal": {
    "entityId": "janeFriends",
    "entityType": "UserGroup"
  },
  "resource": {
    "entityId": "vacationFolder",
    "entityType": "Album"
  }
}
```

### Ejemplo 2: Crear una política estática que conceda acceso a un recurso a todos

El siguiente `create-policy` ejemplo crea una política estática con un alcance de política que especifica solo un recurso.

```
aws verifiedpermissions create-policy \
  --definition file://definition2.txt \
  --policy-store-id PSEXAMPLEabcdefg111111
```

### Contenido del archivo `definition2.txt`:

```
{
```



```

    "static": {
      "description": "Grant everyone access to the publicFolder Album",
      "statement": "permit(principal, action, resource in Album:\""publicFolder
\");"
    }
  }
}

```

Salida:

```

{
  "createdDate": "2023-06-12T20:39:44.975897+00:00",
  "lastUpdatedDate": "2023-06-12T20:39:44.975897+00:00",
  "policyId": "PbfR73F8oh5MMfr9uRtFDB",
  "policyStoreId": "PSEXAMPLEEabcdefg222222",
  "policyType": "STATIC",
  "resource": {
    "entityId": "publicFolder",
    "entityType": "Album"
  }
}

```

Ejemplo 3: Para crear una política vinculada a una plantilla que esté asociada a la plantilla especificada

En el siguiente `create-policy` ejemplo, se crea una política vinculada a una plantilla mediante la plantilla de política especificada y se asocia el principio especificado que se va a utilizar con la nueva política vinculada a la plantilla.

```

aws verifiedpermissions create-policy \
  --definition file://definition2.txt \
  --policy-store-id PSEXAMPLEEabcdefg111111

```

Contenido del archivo `definition3.txt`:

```

{
  "templateLinked": {
    "policyTemplateId": "PTEXAMPLEEabcdefg111111",
    "principal": {
      "entityType": "User",
      "entityId": "alice"
    }
  }
}

```

```
}  
}
```

Salida:

```
{  
  "createdDate": "2023-06-12T20:49:51.490211+00:00",  
  "lastUpdatedDate": "2023-06-12T20:49:51.490211+00:00",  
  "policyId": "TPEXAMPLEabcdefg111111",  
  "policyStoreId": "PSEXAMPLEabcdefg111111",  
  "policyType": "TEMPLATE_LINKED",  
  "principal": {  
    "entityId": "alice",  
    "entityType": "User"  
  },  
  "resource": {  
    "entityId": "VacationPhoto94.jpg",  
    "entityType": "Photo"  
  }  
}
```

Para obtener más información sobre las políticas, consulta las políticas de [permisos verificados de Amazon en la Guía del](#) usuario de permisos verificados de Amazon.

- Para API obtener más información, consulte [UpdatePolicy](#) la Referencia de AWS CLI comandos.

## VPCEjemplos de celosía utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface VPC Lattice.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **create-listener**

En el siguiente ejemplo de código, se muestra cómo usar `create-listener`.

#### AWS CLI

Para crear un oyente

El siguiente `create-listener` ejemplo crea un agente de HTTPS escucha con una regla predeterminada que reenvía el tráfico al grupo objetivo de VPC Lattice especificado.

```
aws vpc-lattice create-listener \  
  --name my-service-listener \  
  --protocol HTTPS \  
  --port 443 \  
  --service-identifier svc-0285b53b2eEXAMPLE \  
  --default-action file://listener-config.json
```

Contenidos de `listener-config.json`:

```
{  
  "forward": {  
    "targetGroups": [  
      {  
        "targetGroupIdentifier": "tg-0eaa4b9ab4EXAMPLE"  
      }  
    ]  
  }  
}
```

Salida:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/  
svc-0285b53b2eEXAMPLE/listener/listener-07cc7fb0abEXAMPLE",  
  "defaultAction": {  
    "forward": {  
      "targetGroups": [  
        {  
          "targetGroupIdentifier": "tg-0eaa4b9ab4EXAMPLE",
```

```

        "weight": 100
      }
    ]
  },
  "id": "listener-07cc7fb0abEXAMPLE",
  "name": "my-service-listener",
  "port": 443,
  "protocol": "HTTPS",
  "serviceArn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
  "serviceId": "svc-0285b53b2eEXAMPLE"
}

```

Para obtener más información, consulte [Listeners](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte la Referencia [CreateListener](#) de AWS CLI comandos.

## create-service-network-service-association

En el siguiente ejemplo de código, se muestra cómo usar `create-service-network-service-association`.

### AWS CLI

Para crear una asociación de servicios

El siguiente `create-service-network-service-association` ejemplo asocia el servicio especificado a la red de servicios especificada.

```

aws vpc-lattice create-service-network-service-association \
  --service-identifier svc-0285b53b2eEXAMPLE \
  --service-network-identifier sn-080ec7dc93EXAMPLE

```

Salida:

```

{
  "arn": "arn:aws:vpc-lattice:us-
east-2:123456789012:servicenetworkserviceassociation/snsa-0e16955a8cEXAMPLE",
  "createdBy": "123456789012",
  "dnsEntry": {

```

```

    "domainName": "my-lattice-service-0285b53b2eEXAMPLE.7d67968.vpc-lattice-
svcs.us-east-2.on.aws",
    "hostedZoneId": "Z09127221KTH2CEXAMPLE"
  },
  "id": "snsa-0e16955a8cEXAMPLE",
  "status": "CREATE_IN_PROGRESS"
}

```

Para obtener más información, consulte [Administrar asociaciones de servicios](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [CreateServiceNetworkServiceAssociation](#) la Referencia de AWS CLI comandos.

## create-service-network-vpc-association

En el siguiente ejemplo de código, se muestra cómo usar `create-service-network-vpc-association`.

### AWS CLI

Para crear una VPC asociación

El siguiente `create-service-network-vpc-association` ejemplo asocia la vpc especificada a la red de servicio especificada. El grupo de seguridad especificado controla qué recursos de la red VPC pueden acceder a la red de servicios y a sus servicios.

```

aws vpc-lattice create-service-network-vpc-association \
  --vpc-identifier vpc-0a1b2c3d4eEXAMPLE \
  --service-network-identifier sn-080ec7dc93EXAMPLE \
  --security-group-ids sg-0aee16bc6cEXAMPLE

```

Salida:

```

{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetworkvpcassociation/
snva-0821fc8631EXAMPLE",
  "createdBy": "123456789012",
  "id": "snva-0821fc8631EXAMPLE",
  "securityGroupIds": [
    "sg-0aee16bc6cEXAMPLE"
  ],
}

```

```
"status": "CREATE_IN_PROGRESS"
}
```

Para obtener más información, consulte [Administrar VPC asociaciones](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [CreateServiceNetworkVpcAssociation](#) la Referencia de AWS CLI comandos.

## create-service-network

En el siguiente ejemplo de código, se muestra cómo usar `create-service-network`.

### AWS CLI

Para crear una red de servicios

En el siguiente `create-service-network` ejemplo, se crea una red de servicios con el nombre especificado.

```
aws vpc-lattice create-service-network \
  --name my-service-network
```

Salida:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/
sn-080ec7dc93EXAMPLE",
  "authType": "NONE",
  "id": "sn-080ec7dc93EXAMPLE",
  "name": "my-service-network"
}
```

Para obtener más información, consulte [Redes de servicios](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [CreateServiceNetwork](#) la Referencia de AWS CLI comandos.

## create-service

En el siguiente ejemplo de código, se muestra cómo usar `create-service`.

## AWS CLI

Para crear un servicio

En el siguiente `create-service` ejemplo, se crea un servicio con el nombre especificado.

```
aws vpc-lattice create-service \  
  --name my-lattice-service
```

Salida:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/  
svc-0285b53b2eEXAMPLE",  
  "authType": "NONE",  
  "dnsEntry": {  
    "domainName": "my-lattice-service-0285b53b2eEXAMPLE.1a2b3c4.vpc-lattice-  
svcs.us-east-2.on.aws",  
    "hostedZoneId": "Z09127221KTH2CEXAMPLE"  
  },  
  "id": "svc-0285b53b2eEXAMPLE",  
  "name": "my-lattice-service",  
  "status": "CREATE_IN_PROGRESS"  
}
```

Para obtener más información, consulte [Servicios en VPC Lattice](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte la Referencia [CreateService](#) de AWS CLI comandos.

## `create-target-group`

En el siguiente ejemplo de código, se muestra cómo usar `create-target-group`.

## AWS CLI

Ejemplo 1: Para crear un grupo objetivo del tipo INSTANCE

El siguiente `create-target-group` ejemplo crea un grupo objetivo con el nombre, el tipo y la configuración especificados.

```
aws vpc-lattice create-target-group \  
  --name my-lattice-target-group-instance \  
  --type INSTANCE \  
  --config file://tg-config.json
```

Contenidos de tg-config.json:

```
{  
  "port": 443,  
  "protocol": "HTTPS",  
  "protocolVersion": "HTTP1",  
  "vpcIdentifier": "vpc-f1663d9868EXAMPLE"  
}
```

Salida:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/  
tg-0eaa4b9ab4EXAMPLE",  
  "config": {  
    "healthCheck": {  
      "enabled": true,  
      "healthCheckIntervalSeconds": 30,  
      "healthCheckTimeoutSeconds": 5,  
      "healthyThresholdCount": 5,  
      "matcher": {  
        "httpCode": "200"  
      },  
      "path": "/",  
      "protocol": "HTTPS",  
      "protocolVersion": "HTTP1",  
      "unhealthyThresholdCount": 2  
    },  
    "port": 443,  
    "protocol": "HTTPS",  
    "protocolVersion": "HTTP1",  
    "vpcIdentifier": "vpc-f1663d9868EXAMPLE"  
  },  
  "id": "tg-0eaa4b9ab4EXAMPLE",  
  "name": "my-lattice-target-group-instance",  
  "status": "CREATE_IN_PROGRESS",  
  "type": "INSTANCE"
```



```
}
```

Ejemplo 2: Para crear un grupo objetivo de tipo IP

El siguiente `create-target-group` ejemplo crea un grupo objetivo con el nombre, el tipo y la configuración especificados.

```
aws vpc-lattice create-target-group \  
  --name my-lattice-target-group-ip \  
  --type IP \  
  --config file://tg-config.json
```

Contenidos de `tg-config.json`:

```
{  
  "ipAddressType": "IPV4",  
  "port": 443,  
  "protocol": "HTTPS",  
  "protocolVersion": "HTTP1",  
  "vpcIdentifier": "vpc-f1663d9868EXAMPLE"  
}
```

Salida:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/  
tg-0eaa4b9ab4EXAMPLE",  
  "config": {  
    "healthCheck": {  
      "enabled": true,  
      "healthCheckIntervalSeconds": 30,  
      "healthCheckTimeoutSeconds": 5,  
      "healthyThresholdCount": 5,  
      "matcher": {  
        "httpCode": "200"  
      },  
      "path": "/",  
      "protocol": "HTTPS",  
      "protocolVersion": "HTTP1",  
      "unhealthyThresholdCount": 2  
    },  
    "ipAddressType": "IPV4",
```

```

    "port": 443,
    "protocol": "HTTPS",
    "protocolVersion": "HTTP1",
    "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
  },
  "id": "tg-0eaa4b9ab4EXAMPLE",
  "name": "my-lattice-target-group-ip",
  "status": "CREATE_IN_PROGRESS",
  "type": "IP"
}

```

### Ejemplo 3: Para crear un grupo objetivo del tipo LAMBDA

El siguiente `create-target-group` ejemplo crea un grupo objetivo con el nombre, el tipo y la configuración especificados.

```

aws vpc-lattice create-target-group \
  --name my-lattice-target-group-lambda \
  --type LAMBDA

```

Salida:

```

{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-0eaa4b9ab4EXAMPLE",
  "id": "tg-0eaa4b9ab4EXAMPLE",
  "name": "my-lattice-target-group-lambda",
  "status": "CREATE_IN_PROGRESS",
  "type": "LAMBDA"
}

```

### Ejemplo 4: Para crear un grupo objetivo del tipo ALB

El siguiente `create-target-group` ejemplo crea un grupo objetivo con el nombre, el tipo y la configuración especificados.

```

aws vpc-lattice create-target-group \
  --name my-lattice-target-group-alb \
  --type ALB \
  --config file://tg-config.json

```

Contenidos de `tg-config.json`:

```
{
  "port": 443,
  "protocol": "HTTPS",
  "protocolVersion": "HTTP1",
  "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
}
```

Salida:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-0eaa4b9ab4EXAMPLE",
  "config": {
    "port": 443,
    "protocol": "HTTPS",
    "protocolVersion": "HTTP1",
    "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
  },
  "id": "tg-0eaa4b9ab4EXAMPLE",
  "name": "my-lattice-target-group-alb",
  "status": "CREATE_IN_PROGRESS",
  "type": "ALB"
}
```

Para obtener más información, consulte [Grupos objetivo](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [CreateTargetGroup](#) la Referencia de AWS CLI comandos.

## delete-auth-policy

En el siguiente ejemplo de código, se muestra cómo usar delete-auth-policy.

### AWS CLI

Para eliminar una política de autenticación

En el siguiente delete-auth-policy ejemplo, se elimina la política de autenticación del servicio especificado.

```
aws vpc-lattice delete-auth-policy \
```

```
--resource-identifier svc-0285b53b2eEXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [las políticas de autenticación](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte la Referencia [DeleteAuthPolicy](#) de AWS CLI comandos.

## **delete-listener**

En el siguiente ejemplo de código, se muestra cómo usar `delete-listener`.

AWS CLI

Para eliminar un oyente

En el siguiente `delete-listener` ejemplo, se elimina el oyente especificado.

```
aws vpc-lattice delete-listener \  
  --listener-identifier listener-07cc7fb0abEXAMPLE \  
  --service-identifier svc-0285b53b2eEXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Listeners](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte la Referencia [DeleteListener](#) de AWS CLI comandos.

## **delete-service-network-service-association**

En el siguiente ejemplo de código, se muestra cómo usar `delete-service-network-service-association`.

AWS CLI

Para eliminar una asociación de servicios

El siguiente `delete-service-network-service-association` ejemplo desasocia la asociación de servicios especificada.

```
aws vpc-lattice delete-service-network-service-association \  
--service-network-service-association-identifier snsa-031fabb4d8EXAMPLE
```

Salida:

```
{  
  "arn": "arn:aws:vpc-lattice:us-  
east-2:123456789012:servicenetworkserviceassociation/snsa-031fabb4d8EXAMPLE",  
  "id": "snsa-031fabb4d8EXAMPLE",  
  "status": "DELETE_IN_PROGRESS"  
}
```

Para obtener más información, consulte [Administrar asociaciones de servicios](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [DeleteServiceNetworkServiceAssociation](#) la Referencia de AWS CLI comandos.

## **delete-service-network-vpc-association**

En el siguiente ejemplo de código, se muestra cómo usar `delete-service-network-vpc-association`.

### AWS CLI

Para eliminar una VPC asociación

En el siguiente `delete-service-network-vpc-association` ejemplo, se desasocia la VPC asociación especificada.

```
aws vpc-lattice delete-service-network-vpc-association \  
--service-network-vpc-association-identifier snva-0821fc8631EXAMPLE
```

Salida:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetworkvpcassociation/  
snva-0821fc8631EXAMPLE",  
  "id": "snva-0821fc8631EXAMPLE",  
  "status": "DELETE_IN_PROGRESS"  
}
```

Para obtener más información, consulte [Administrar VPC asociaciones](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [DeleteServiceNetworkVpcAssociation](#) la Referencia de AWS CLI comandos.

## **delete-service-network**

En el siguiente ejemplo de código, se muestra cómo usar `delete-service-network`.

### AWS CLI

Para eliminar una red de servicio

En el siguiente `delete-service-network` ejemplo, se elimina la red de servicio especificada.

```
aws vpc-lattice delete-service-network \  
  --service-network-identifier sn-080ec7dc93EXAMPLE
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Redes de servicios](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [DeleteServiceNetwork](#) la Referencia de AWS CLI comandos.

## **delete-service**

En el siguiente ejemplo de código, se muestra cómo usar `delete-service`.

### AWS CLI

Eliminación de un servicio

En el siguiente `delete-service` ejemplo, se elimina el servicio especificado.

```
aws vpc-lattice delete-service \  
  --service-identifier svc-0285b53b2eEXAMPLE
```

Salida:

```
{
  "arn": "arn:aws:vpc-lattice:us-west-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
  "id": "svc-0285b53b2eEXAMPLE",
  "name": "my-lattice-service",
  "status": "DELETE_IN_PROGRESS"
}
```

Para obtener más información, consulte [Servicios en VPC Lattice](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte la Referencia [DeleteService](#) de AWS CLI comandos.

## delete-target-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-target-group`.

AWS CLI

Eliminación de un grupo de destino

En el siguiente ejemplo de `delete-target-group` se elimina el grupo de destino especificado.

```
aws vpc-lattice delete-target-group \
  --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Salida:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-0eaa4b9ab4EXAMPLE",
  "id": "tg-0eaa4b9ab4EXAMPLE",
  "status": "DELETE_IN_PROGRESS"
}
```

Para obtener más información, consulte [Grupos objetivo](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [DeleteTargetGroup](#) la Referencia de AWS CLI comandos.

## deregister-targets

En el siguiente ejemplo de código, se muestra cómo usar `deregister-targets`.

### AWS CLI

Para anular el registro de un objetivo

En el siguiente `deregister-targets` ejemplo, se anula el registro del objetivo especificado del grupo de objetivos especificado.

```
aws vpc-lattice deregister-targets \  
  --targets i-07dd579bc5EXAMPLE \  
  --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Salida:

```
{  
  "successful": [  
    {  
      "id": "i-07dd579bc5EXAMPLE",  
      "port": 443  
    }  
  ],  
  "unsuccessful": []  
}
```

Para obtener más información, consulte [Registrar objetivos](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [DeregisterTargets](#) la Referencia de AWS CLI comandos.

## get-auth-policy

En el siguiente ejemplo de código, se muestra cómo usar `get-auth-policy`.

### AWS CLI

Para obtener información sobre una política de autenticación

En el siguiente `get-auth-policy` ejemplo, se obtiene información sobre la política de autenticación del servicio especificado.



```
aws vpc-lattice get-auth-policy \  
  --resource-identifier svc-0285b53b2eEXAMPLE
```

Salida:

```
{  
  "createdAt": "2023-06-07T03:51:20.266Z",  
  "lastUpdatedAt": "2023-06-07T04:39:27.082Z",  
  "policy": "{\n\"Version\": \"2012-10-17\", \"Statement\": [\n{\n\"Effect\": \"Allow\", \"Principal\": {\n\"AWS\": \"arn:aws:iam::123456789012:role/my-clients\"},\n\"Action\": \"vpc-lattice-svcs:Invoke\", \"Resource\": \"arn:aws:vpc-lattice:us-east-2:123456789012:service/svc-0285b53b2eEXAMPLE\"}]}",  
  "state": "Active"  
}
```

Para obtener más información, consulte [las políticas de autenticación](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte la Referencia [GetAuthPolicy](#) de AWS CLI comandos.

## get-listener

En el siguiente ejemplo de código, se muestra cómo usar `get-listener`.

AWS CLI

Para obtener información sobre un agente de escucha de servicios

En el siguiente `get-listener` ejemplo, se obtiene información sobre el agente de escucha especificado para el servicio especificado.

```
aws vpc-lattice get-listener \  
  --listener-identifier listener-0ccf55918cEXAMPLE \  
  --service-identifier svc-0285b53b2eEXAMPLE
```

Salida:

```
{  
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/  
  svc-0285b53b2eEXAMPLE/listener/listener-0ccf55918cEXAMPLE",
```

```
"createdAt": "2023-05-07T05:08:45.192Z",
"defaultAction": {
  "forward": {
    "targetGroups": [
      {
        "targetGroupIdentifier": "tg-0ff213abb6EXAMPLE",
        "weight": 1
      }
    ]
  }
},
"id": "listener-0ccf55918cEXAMPLE",
"lastUpdatedAt": "2023-05-07T05:08:45.192Z",
"name": "http-80",
"port": 80,
"protocol": "HTTP",
"serviceArn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
"serviceId": "svc-0285b53b2eEXAMPLE"
}
```

Para obtener más información, consulte [Definir el enrutamiento](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [GetListener](#) la Referencia de AWS CLI comandos.

## get-service-network-service-association

En el siguiente ejemplo de código, se muestra cómo usar `get-service-network-service-association`.

### AWS CLI

Para obtener información sobre una asociación de servicios

En el siguiente `get-service-network-service-association` ejemplo, se obtiene información sobre la asociación de servicios especificada.

```
aws vpc-lattice get-service-network-service-association \
  --service-network-service-association-identifier snsa-031fabb4d8EXAMPLE
```

Salida:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetworkserviceassociation/snsa-031fabb4d8EXAMPLE",
  "createdAt": "2023-05-05T21:48:16.076Z",
  "createdBy": "123456789012",
  "dnsEntry": {
    "domainName": "my-lattice-service-0285b53b2eEXAMPLE.7d67968.vpc-lattice-svcs.us-east-2.on.aws",
    "hostedZoneId": "Z09127221KTH2CEXAMPLE"
  },
  "id": "snsa-031fabb4d8EXAMPLE",
  "serviceArn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/svc-0285b53b2eEXAMPLE",
  "serviceId": "svc-0285b53b2eEXAMPLE",
  "serviceName": "my-lattice-service",
  "serviceNetworkArn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/sn-080ec7dc93EXAMPLE",
  "serviceNetworkId": "sn-080ec7dc93EXAMPLE",
  "serviceNetworkName": "my-service-network",
  "status": "ACTIVE"
}
```

Para obtener más información, consulte [Administrar asociaciones de servicios](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [GetServiceNetworkServiceAssociation](#) la Referencia de AWS CLI comandos.

## get-service-network-vpc-association

En el siguiente ejemplo de código, se muestra cómo usar `get-service-network-vpc-association`.

### AWS CLI

Para obtener información sobre una VPC asociación

En el siguiente `get-service-network-vpc-association` ejemplo, se obtiene información sobre la VPC asociación especificada.

```
aws vpc-lattice get-service-network-vpc-association \
  --service-network-vpc-association-identifier snva-0821fc8631EXAMPLE
```

Salida:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetworkvpcassociation/
snva-0821fc8631EXAMPLE",
  "createdAt": "2023-06-06T23:41:08.421Z",
  "createdBy": "123456789012",
  "id": "snva-0c5dcb60d6EXAMPLE",
  "lastUpdatedAt": "2023-06-06T23:41:08.421Z",
  "securityGroupIds": [
    "sg-0aee16bc6cEXAMPLE"
  ],
  "serviceNetworkArn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/
sn-080ec7dc93EXAMPLE",
  "serviceNetworkId": "sn-080ec7dc93EXAMPLE",
  "serviceNetworkName": "my-service-network",
  "status": "ACTIVE",
  "vpcId": "vpc-0a1b2c3d4eEXAMPLE"
}
```

Para obtener más información, consulte [Administrar VPC asociaciones](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [GetServiceNetworkVpcAssociation](#) la Referencia de AWS CLI comandos.

## get-service-network

En el siguiente ejemplo de código, se muestra cómo usar `get-service-network`.

AWS CLI

Para obtener información sobre una red de servicios

En el siguiente `get-service-network` ejemplo, se obtiene información sobre la red de servicios especificada.

```
aws vpc-lattice get-service-network \
  --service-network-identifier sn-080ec7dc93EXAMPLE
```

Salida:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/
sn-080ec7dc93EXAMPLE",
  "authType": "AWS_IAM",
  "createdAt": "2023-05-05T15:26:08.417Z",
  "id": "sn-080ec7dc93EXAMPLE",
  "lastUpdatedAt": "2023-05-05T15:26:08.417Z",
  "name": "my-service-network",
  "numberOfAssociatedServices": 2,
  "numberOfAssociatedVPCs": 3
}
```

Para obtener más información, consulte [Redes de servicios](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [GetServiceNetwork](#) la Referencia de AWS CLI comandos.

## get-service

En el siguiente ejemplo de código, se muestra cómo usar `get-service`.

### AWS CLI

Para obtener información sobre un servicio

En el siguiente `get-service` ejemplo, se obtiene información sobre el servicio especificado.

```
aws vpc-lattice get-service \
  --service-identifier svc-0285b53b2eEXAMPLE
```

Salida:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
  "authType": "AWS_IAM",
  "createdAt": "2023-05-05T21:35:29.339Z",
  "dnsEntry": {
    "domainName": "my-lattice-service-0285b53b2eEXAMPLE.7d67968.vpc-lattice-
svcs.us-east-2.on.aws",
    "hostedZoneId": "Z09127221KTH2CFU0HIZH"
  }
}
```

```
  },
  "id": "svc-0285b53b2eEXAMPLE",
  "lastUpdatedAt": "2023-05-05T21:35:29.339Z",
  "name": "my-lattice-service",
  "status": "ACTIVE"
}
```

Para obtener más información, consulte [Servicios](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [GetService](#) la Referencia de AWS CLI comandos.

## get-target-group

En el siguiente ejemplo de código, se muestra cómo usar `get-target-group`.

### AWS CLI

Para obtener información sobre un grupo objetivo

En el siguiente `get-target-group` ejemplo se obtiene información sobre el grupo objetivo especificado, que tiene un tipo de objetivo de `INSTANCE`.

```
aws vpc-lattice get-target-group \
  --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Salida:

```
{
  "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-0eaa4b9ab4EXAMPLE",
  "config": {
    "healthCheck": {
      "enabled": true,
      "healthCheckIntervalSeconds": 30,
      "healthCheckTimeoutSeconds": 5,
      "healthyThresholdCount": 5,
      "matcher": {
        "httpCode": "200"
      }
    },
    "path": "/",
    "protocol": "HTTPS",
    "protocolVersion": "HTTP1",
    "unhealthyThresholdCount": 2
  }
}
```

```

    },
    "port": 443,
    "protocol": "HTTPS",
    "protocolVersion": "HTTP1",
    "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
  },
  "createdAt": "2023-05-06T04:41:04.122Z",
  "id": "tg-0eaa4b9ab4EXAMPLE",
  "lastUpdatedAt": "2023-05-06T04:41:04.122Z",
  "name": "my-target-group",
  "serviceArns": [
    "arn:aws:vpc-lattice:us-east-2:123456789012:service/svc-0285b53b2eEXAMPLE"
  ],
  "status": "ACTIVE",
  "type": "INSTANCE"
}

```

Para obtener más información, consulte [Grupos objetivo](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [GetTargetGroup](#) la Referencia de AWS CLI comandos.

## list-listeners

En el siguiente ejemplo de código, se muestra cómo usar `list-listeners`.

### AWS CLI

Para enumerar los oyentes del servicio

En el siguiente `list-listeners` ejemplo, se enumeran los oyentes del servicio especificado.

```
aws vpc-lattice list-listeners \
  --service-identifier svc-0285b53b2eEXAMPLE
```

Salida:

```

{
  "items": [
    {
      "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE/listener/listener-0ccf55918cEXAMPLE",

```

```
    "createdAt": "2023-05-07T05:08:45.192Z",
    "id": "listener-0ccf55918cEXAMPLE",
    "lastUpdatedAt": "2023-05-07T05:08:45.192Z",
    "name": "http-80",
    "port": 80,
    "protocol": "HTTP"
  }
]
```

Para obtener más información, consulte [Definir el enrutamiento](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [ListListeners](#) la Referencia de AWS CLI comandos.

## **list-service-network-service-associations**

En el siguiente ejemplo de código, se muestra cómo usar `list-service-network-service-associations`.

### AWS CLI

Para enumerar las asociaciones de servicios

En el siguiente `list-service-network-service-associations` ejemplo, se enumeran las asociaciones de servicios de la red de servicios especificada. La `--query` opción limita el resultado a las asociaciones IDs de servicios.

```
aws vpc-lattice list-service-network-service-associations \
  --service-network-identifier sn-080ec7dc93EXAMPLE \
  --query items[*].id
```

Salida:

```
[
  "snsa-031fabb4d8EXAMPLE",
  "snsa-0e16955a8cEXAMPLE"
]
```

Para obtener más información, consulte [Administrar asociaciones de servicios](#) en la Guía del usuario de Amazon VPC Lattice.



- Para API obtener más información, consulte [ListServiceNetworkServiceAssociations](#) la Referencia de AWS CLI comandos.

## **list-service-network-vpc-associations**

En el siguiente ejemplo de código, se muestra cómo usar `list-service-network-vpc-associations`.

### AWS CLI

Para enumerar VPC las asociaciones

En el siguiente `list-service-network-vpc-associations` ejemplo, se enumeran las VPC asociaciones de la red de servicio especificada. La `--query` opción limita el resultado a IDs las VPC asociaciones.

```
aws vpc-lattice list-service-network-vpc-associations \  
  --service-network-identifier sn-080ec7dc93EXAMPLE \  
  --query items[*].id
```

Salida:

```
[  
  "snva-0821fc8631EXAMPLE",  
  "snva-0c5dcb60d6EXAMPLE"  
]
```

Para obtener más información, consulte [Administrar VPC asociaciones](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [ListServiceNetworkVpcAssociations](#) la Referencia de AWS CLI comandos.

## **list-service-networks**

En el siguiente ejemplo de código, se muestra cómo usar `list-service-networks`.

### AWS CLI

Para enumerar sus redes de servicio

En el siguiente `list-service-networks` ejemplo, se enumeran las redes de servicio que son propiedad de la cuenta que realiza la llamada o que se comparten con ella. La `--query` opción extiende los resultados a los nombres de recursos de Amazon (ARN) de las redes de servicio.

```
aws vpc-lattice list-service-networks \  
  --query items[*].arn
```

Salida:

```
[  
  "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/  
sn-080ec7dc93EXAMPLE",  
  "arn:aws:vpc-lattice:us-east-2:111122223333:servicenetwork/sn-0ec4d436cfEXAMPLE"  
]
```

Para obtener más información, consulte [Redes de servicios](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [ListServiceNetworks](#) la Referencia de AWS CLI comandos.

## **list-services**

En el siguiente ejemplo de código, se muestra cómo usar `list-services`.

AWS CLI

Para enumerar sus servicios

En el siguiente `list-services` ejemplo, se enumeran los servicios que son propiedad de la cuenta que realiza la llamada o que se comparten con ella. La `--query` opción establece el alcance de los resultados a los nombres de recursos de Amazon (ARN) de los servicios.

```
aws vpc-lattice list-services \  
  --query items[*].arn
```

Salida:

```
[  
  "arn:aws:vpc-lattice:us-east-2:123456789012:service/svc-0285b53b2eEXAMPLE",
```

```
"arn:aws:vpc-lattice:us-east-2:111122223333:service/svc-0b8ac96550EXAMPLE"  
]
```

Para obtener más información, consulte [Servicios](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [ListServices](#) la Referencia de AWS CLI comandos.

## list-target-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-target-groups`.

### AWS CLI

Para enumerar sus grupos objetivo

En el siguiente `list-target-groups` ejemplo, se enumeran los grupos objetivo con un tipo de objetivo de LAMBDA.

```
aws vpc-lattice list-target-groups \  
  --target-group-type LAMBDA
```

Salida:

```
{  
  "items": [  
    {  
      "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/  
tg-045c1b7d9dEXAMPLE",  
      "createdAt": "2023-05-06T05:22:16.637Z",  
      "id": "tg-045c1b7d9dEXAMPLE",  
      "lastUpdatedAt": "2023-05-06T05:22:16.637Z",  
      "name": "my-target-group-lam",  
      "serviceArns": [  
        "arn:aws:vpc-lattice:us-east-2:123456789012:service/  
svc-0285b53b2eEXAMPLE"  
      ],  
      "status": "ACTIVE",  
      "type": "LAMBDA"  
    }  
  ]  
}
```

Para obtener más información, consulte [Grupos objetivo](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [ListTargetGroups](#) la Referencia de AWS CLI comandos.

## list-targets

En el siguiente ejemplo de código, se muestra cómo usar `list-targets`.

### AWS CLI

Para enumerar los objetivos de un grupo objetivo

En el siguiente `list-targets` ejemplo, se enumeran los objetivos del grupo de objetivos especificado.

```
aws vpc-lattice list-targets \  
  --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Salida:

```
{  
  "items": [  
    {  
      "id": "i-07dd579bc5EXAMPLE",  
      "port": 443,  
      "status": "HEALTHY"  
    },  
    {  
      "id": "i-047b3c9078EXAMPLE",  
      "port": 443,  
      "reasonCode": "HealthCheckFailed",  
      "status": "UNHEALTHY"  
    }  
  ]  
}
```

Para obtener más información, consulte [Grupos objetivo](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [ListTargets](#) la Referencia de AWS CLI comandos.

## put-auth-policy

En el siguiente ejemplo de código, se muestra cómo usar `put-auth-policy`.

### AWS CLI

Para crear una política de autenticación para un servicio

El siguiente `put-auth-policy` ejemplo concede acceso a las solicitudes de cualquier principal autenticado que utilice la función especificada al IAM. El recurso es el ARN del servicio al que está asociada la política.

```
aws vpc-lattice put-auth-policy \
  --resource-identifier svc-0285b53b2eEXAMPLE \
  --policy file://auth-policy.json
```

Contenidos de `auth-policy.json`:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:role/my-clients"
      },
      "Action": "vpc-lattice-svcs:Invoke",
      "Resource": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE"
    }
  ]
}
```

Salida:

```
{
  "policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow\",
\", \"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:role/my-clients\"},
\", \"Action\":\"vpc-lattice-svcs:Invoke\", \"Resource\":\"arn:aws:vpc-lattice:us-
east-2:123456789012:service/svc-0285b53b2eEXAMPLE\"}]}",
  "state": "Active"
```

```
}
```

Para obtener más información, consulte [las políticas de autenticación](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte la Referencia [PutAuthPolicy](#) de AWS CLI comandos.

## register-targets

En el siguiente ejemplo de código, se muestra cómo usar `register-targets`.

### AWS CLI

Para registrar un objetivo

En el siguiente `register-targets` ejemplo, se registran los objetivos especificados con el grupo de objetivos especificado.

```
aws vpc-lattice register-targets \  
  --targets id=i-047b3c9078EXAMPLE id=i-07dd579bc5EXAMPLE \  
  --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Salida:

```
{  
  "successful": [  
    {  
      "id": "i-07dd579bc5EXAMPLE",  
      "port": 443  
    }  
  ],  
  "unsuccessful": [  
    {  
      "failureCode": "UnsupportedTarget",  
      "failureMessage": "Instance targets must be in the same VPC as their  
target group",  
      "id": "i-047b3c9078EXAMPLE",  
      "port": 443  
    }  
  ]  
}
```

Para obtener más información, consulte [Registrar objetivos](#) en la Guía del usuario de Amazon VPC Lattice.

- Para API obtener más información, consulte [RegisterTargets](#) la Referencia de AWS CLI comandos.

## AWS WAF Classic ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS WAF Classic.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **put-logging-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `put-logging-configuration`.

AWS CLI

Para crear una configuración de registro para la web ACL ARN con la transmisión de Kinesis Firehose especificada ARN

En el siguiente `put-logging-configuration` ejemplo, se muestra la configuración de registro para `withWAF`. CloudFront

```
aws waf put-logging-configuration \  
  --logging-configuration ResourceArn=arn:aws:waf::123456789012:webacl/3bffd3ed-  
fa2e-445e-869f-a6a7cf153fd3,LogDestinationConfigs=arn:aws:firehose:us-  
east-1:123456789012:deliverystream/aws-waf-logs-firehose-stream,RedactedFields=[]
```

Salida:

```
{
  "LoggingConfiguration": {
    "ResourceArn": "arn:aws:waf::123456789012:webacl/3bffd3ed-fa2e-445e-869f-
a6a7cf153fd3",
    "LogDestinationConfigs": [
      "arn:aws:firehose:us-east-1:123456789012:deliverystream/aws-waf-logs-
firehose-stream"
    ]
  }
}
```

- Para API obtener más información, consulte [PutLoggingConfiguration](#) la Referencia de AWS CLI comandos.

## update-byte-match-set

En el siguiente ejemplo de código, se muestra cómo usar `update-byte-match-set`.

AWS CLI

Para actualizar un conjunto de coincidencias de bytes

El siguiente `update-byte-match-set` comando elimina un `ByteMatchTuple` objeto (filtro) de un `ByteMatchSet`:

```
aws waf update-byte-match-set --byte-match-set-id a123fae4-
b567-8e90-1234-5ab67ac8ca90 --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --
updates
Action="DELETE",ByteMatchTuple={FieldToMatch={Type="HEADER",Data="referer"},TargetString="b
```

Para obtener más información, consulte [Trabajar con condiciones de coincidencia de cadenas en la guía para AWS WAF desarrolladores](#).

- Para API obtener más información, consulte [UpdateByteMatchSet](#) la Referencia de AWS CLI comandos.

## update-ip-set

En el siguiente ejemplo de código, se muestra cómo usar `update-ip-set`.



## AWS CLI

Para actualizar un conjunto de direcciones IP

El siguiente `update-ip-set` comando actualiza un IPSet con una IPv4 dirección y elimina una IPv6 dirección:

```
aws waf update-ip-set --ip-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90
--change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates
Action="INSERT",IPSetDescriptor={Type="IPV4",Value="12.34.56.78/16"},Action="DELETE",IPSetD
```

Como alternativa, puede usar un JSON archivo para especificar la entrada. Por ejemplo:

```
aws waf update-ip-set --ip-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 --change-
token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates file://change.json
```

Dónde está el contenido del JSON archivo:

```
[
{
  "Action": "INSERT",
  "IPSetDescriptor":
  {
    "Type": "IPV4",
    "Value": "12.34.56.78/16"
  }
},
{
  "Action": "DELETE",
  "IPSetDescriptor":
  {
    "Type": "IPV6",
    "Value": "1111:0000:0000:0000:0000:0000:0000:0111/128"
  }
}
]
```

Para obtener más información, consulte [Trabajar con condiciones de coincidencia de IP en la guía para AWS WAFdesarrolladores](#).

- Para API obtener más información, consulte [UpdateIpSet](#) la Referencia de AWS CLI comandos.

## update-rule

En el siguiente ejemplo de código, se muestra cómo usar `update-rule`.

### AWS CLI

Para actualizar una regla

El siguiente `update-rule` comando elimina un objeto Predicate de una regla:

```
aws waf update-rule --rule-id a123fae4-b567-8e90-1234-5ab67ac8ca90
--change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates
Action="DELETE",Predicate={Negated=false,Type="ByteMatch",DataId="MyByteMatchSetID"}
```

Para obtener más información, consulte [Trabajar con reglas](#) en la guía para AWS WAFdesarrolladores.

- Para API obtener más información, consulte [UpdateRule](#) la Referencia de AWS CLI comandos.

## update-size-constraint-set

En el siguiente ejemplo de código, se muestra cómo usar `update-size-constraint-set`.

### AWS CLI

Para actualizar un conjunto de restricciones de tamaño

El siguiente `update-size-constraint-set` comando elimina un SizeConstraint objeto (filtros) de un conjunto de restricciones de tamaño:

```
aws waf update-size-constraint-set --size-constraint-set-id a123fae4-
b567-8e90-1234-5ab67ac8ca90 --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --
updates
Action="DELETE",SizeConstraint={FieldToMatch={Type="QUERY_STRING"},TextTransformation="NONE"}
```

Para obtener más información, consulte [Trabajar con condiciones de restricciones de tamaño](#) en la AWS WAFguía para desarrolladores.

- Para API obtener más información, consulte [UpdateSizeConstraintSet](#) la Referencia de AWS CLI comandos.

## update-sql-injection-match-set

En el siguiente ejemplo de código, se muestra cómo usar `update-sql-injection-match-set`.

### AWS CLI

Para actualizar un conjunto SQL de combinaciones de inyección

El siguiente `update-sql-injection-match-set` comando elimina un `SqlInjectionMatchTuple` objeto (filtros) de un conjunto de coincidencias SQL por inyección:

```
aws waf update-sql-injection-match-set --sql-injection-  
match-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 --  
change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates  
Action="DELETE",SqlInjectionMatchTuple={FieldToMatch={Type="QUERY_STRING"},TextTransformati
```

Para obtener más información, consulte [Cómo trabajar con condiciones SQL de coincidencia de inyecciones en la guía para AWS WAFdesarrolladores](#).

- Para API obtener más información, consulte [UpdateSqlInjectionMatchSet](#) la Referencia de AWS CLI comandos.

## update-web-acl

En el siguiente ejemplo de código, se muestra cómo usar `update-web-acl`.

### AWS CLI

Para actualizar una web ACL

El siguiente `update-web-acl` comando elimina un `ActivatedRule` objeto de una `WebACL`.

```
aws waf update-web-acl -- web-acl-id a123fae4-b567-8e90-1234-5ab67ac8ca90 --change-token  
12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates Action=» «, =' {Priority=1, =» -1-Example»,  
Action= {Type=» «}, Type=» «} 'DELETEActivatedRuleRuleIdWAFRuleALLOWREGULAR
```

Salida:

```
{  
  "ChangeToken": "12cs345-67cd-890b-1cd2-c3a4567d89f1"  
}
```

Para obtener más información, consulte [Trabajar con la Web ACLs](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [UpdateWebAcl](#) la Referencia de AWS CLI comandos.

## update-xss-match-set

En el siguiente ejemplo de código, se muestra cómo usar `update-xss-match-set`.

### AWS CLI

Para actualizar un XSSMatchSet

El siguiente `update-xss-match-set` comando elimina un `XssMatchTuple` objeto (filtros) en un `XssMatchSet`:

```
aws waf update-xss-match-set --xss-match-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90
--change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates
Action="DELETE",XssMatchTuple={FieldToMatch={Type="QUERY_STRING"},TextTransformation="URL_D
```

Para obtener más información, consulte [Trabajar con condiciones de coincidencia de secuencias de comandos entre sitios](#) en la AWS WAFguía para desarrolladores.

- Para API obtener más información, consulte la Referencia [UpdateXssMatchSet](#) de AWS CLI comandos.

## AWS WAF Classic Regional ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS WAF Classic Regional.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### **associate-web-acl**

En el siguiente ejemplo de código, se muestra cómo usar `associate-web-acl`.

#### AWS CLI

Para asociar una web ACL a un recurso

El siguiente `associate-web-acl` comando asocia una webACL, especificada por `web-acl-id`, con un recurso, especificado por `resource-arn`. El recurso ARN puede hacer referencia a un balanceador de carga de aplicaciones o a una puerta de enlace: API

```
aws waf-regional associate-web-acl \  
  --web-acl-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \  
  --resource-arn 12cs345-67cd-890b-1cd2-c3a4567d89f1
```

Para obtener más información, consulte [Trabajar con la Web ACLs](#) en la Guía para AWS WAF desarrolladores.

- Para API obtener más información, consulte [AssociateWebAcl](#) la Referencia de AWS CLI comandos.

### **put-logging-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `put-logging-configuration`.

#### AWS CLI

Para crear una configuración de registro para la web ACL ARN con la transmisión de Kinesis Firehose especificada ARN

En el siguiente `put-logging-configuration` ejemplo, se muestra la configuración de registro para la WAF región conALB/APIGateway. us-east-1

```
aws waf-regional put-logging-configuration \  
  --logging-configuration ResourceArn=arn:aws:waf-  
regional:us-east-1:123456789012:webacl/3bffd3ed-fa2e-445e-869f-
```

```
a6a7cf153fd3,LogDestinationConfigs=arn:aws:firehose:us-east-1:123456789012:deliverystream/aws-waf-logs-firehose-stream,RedactedFields=[] \
--region us-east-1
```

Salida:

```
{
  "LoggingConfiguration": {
    "ResourceArn": "arn:aws:waf-regional:us-east-1:123456789012:webacl/3bffd3ed-fa2e-445e-869f-a6a7cf153fd3",
    "LogDestinationConfigs": [
      "arn:aws:firehose:us-east-1:123456789012:deliverystream/aws-waf-logs-firehose-stream"
    ]
  }
}
```

- Para API obtener más información, consulte [PutLoggingConfiguration](#) la Referencia de AWS CLI comandos.

## update-byte-match-set

En el siguiente ejemplo de código, se muestra cómo usar update-byte-match-set.

AWS CLI

Para actualizar un conjunto de coincidencias de bytes

El siguiente update-byte-match-set comando elimina un ByteMatchTuple objeto (filtro) en un ByteMatchSet. Como el updates valor tiene comillas dobles incrustadas, debe rodearlo entre comillas simples.

```
aws waf-regional update-byte-match-set \
--byte-match-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
--change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
--updates
'Action="DELETE",ByteMatchTuple={FieldToMatch={Type="HEADER",Data="referer"},TargetString="'
```

Para obtener más información, consulte Cómo [trabajar con condiciones de coincidencia de cadenas](#) en la Guía para AWS WAF desarrolladores.

- Para API obtener más información, consulte [UpdateByteMatchSet](#) la Referencia de AWS CLI comandos.

## update-ip-set

En el siguiente ejemplo de código, se muestra cómo usar `update-ip-set`.

### AWS CLI

Para actualizar un conjunto de IP

El siguiente `update-ip-set` comando actualiza an IPSet con una IPv4 dirección y elimina una IPv6 dirección. Obtenga el valor de `change-token` ejecutando el `get-change-token` comando. Como el valor de las actualizaciones incluye comillas dobles incrustadas, debe escribir el valor entre comillas simples.

```
aws waf update-ip-set \
  --ip-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
  --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
  --updates
  'Action="INSERT",IPSetDescriptor={Type="IPV4",Value="12.34.56.78/16"},Action="DELETE",IPSet'
```

Como alternativa, puede usar un JSON archivo para especificar la entrada. Por ejemplo:

```
aws waf-regional update-ip-set \
  --ip-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
  --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
  --updates file://change.json
```

Contenido del `change.json`

```
[
  {
    "Action": "INSERT",
    "IPSetDescriptor":
    {
      "Type": "IPV4",
      "Value": "12.34.56.78/16"
    }
  },
]
```

```

    {
      "Action": "DELETE",
      "IPSetDescriptor":
      {
        "Type": "IPV6",
        "Value": "1111:0000:0000:0000:0000:0000:0000:0111/128"
      }
    }
  ]

```

Para obtener más información, consulte [Cómo trabajar con condiciones de coincidencia de IP](#) en la Guía para AWS WAF desarrolladores.

- Para API obtener más información, consulte [UpdateIpSet](#) la Referencia de AWS CLI comandos.

## update-rule

En el siguiente ejemplo de código, se muestra cómo usar `update-rule`.

### AWS CLI

Para actualizar una regla

El siguiente `update-rule` comando elimina un `Predicate` objeto de una regla. Como el `updates` valor tiene comillas dobles incrustadas, debe escribir todo el valor entre comillas simples.

```

aws waf-regional update-rule \
  --rule-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
  --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
  --updates
'Action="DELETE",Predicate={Negated=false,Type="ByteMatch",DataId="MyByteMatchSetID"}'

```

Para obtener más información, consulte [Cómo trabajar con reglas](#) en la Guía para AWS WAF desarrolladores.

- Para API obtener más información, consulte [UpdateRule](#) la Referencia de AWS CLI comandos.

## update-size-constraint-set

En el siguiente ejemplo de código, se muestra cómo usar `update-size-constraint-set`.



## AWS CLI

Para actualizar un conjunto de restricciones de tamaño

El siguiente `update-size-constraint-set` comando elimina un objeto `SizeConstraint` (filtros) de un conjunto de restricciones de tamaño. Como el `updates` valor contiene comillas dobles incrustadas, debe escribir todo el valor entre comillas simples.

```
aws waf-regional update-size-constraint-set \
  --size-constraint-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
  --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
  --updates
  'Action="DELETE",SizeConstraint={FieldToMatch={Type="QUERY_STRING"},TextTransformation="NON"
```

Para obtener más información, consulte Cómo [trabajar con condiciones de restricciones de tamaño](#) en la Guía para AWS WAF desarrolladores.

- Para API obtener más información, consulte [UpdateSizeConstraintSet](#) la Referencia de AWS CLI comandos.

## update-sql-injection-match-set

En el siguiente ejemplo de código, se muestra cómo usar `update-sql-injection-match-set`.

### AWS CLI

Para actualizar un conjunto SQL de combinaciones de inyección

El siguiente `update-sql-injection-match-set` comando elimina un `SqlInjectionMatchTuple` objeto (filtros) de un conjunto de coincidencias SQL por inyección. Como el `updates` valor contiene comillas dobles incrustadas, debe escribir todo el valor entre comillas simples. :

```
aws waf-regional update-sql-injection-match-set --sql-injection-match-set -id a123fae4-b567-8e90-1234-5ab67ac8ca90 --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates 'Action=> <, = {= {Type=> _ <}, => _ <}'
'DELETESqlInjectionMatchTupleFieldToMatchQUERYSTRINGTextTransformationURLDECODE
```

Para obtener más información, [consulte Cómo trabajar con las condiciones de Inyección SQL Match en la guía para desarrolladores](#).AWS WAF

- Para API obtener más información, consulte [UpdateSqlInjectionMatchSet](#) la Referencia de AWS CLI comandos.

## update-web-acl

En el siguiente ejemplo de código, se muestra cómo usar `update-web-acl`.

### AWS CLI

Para actualizar una web ACL

El siguiente `update-web-acl` comando elimina un `ActivatedRule` objeto de una `WebACL`. Como el `updates` valor contiene comillas dobles incrustadas, debe escribir todo el valor entre comillas simples.

```
aws waf-regional update-web-acl \  
  --web-acl-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \  
  --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \  
  --updates Action="DELETE",ActivatedRule='{Priority=1,RuleId="WAFRule-1-  
Example",Action={Type="ALLOW"},Type="ALLOW"}'
```

Para obtener más información, consulte [Trabajar con la Web ACLs](#) en la Guía para AWS WAF desarrolladores.

- Para API obtener más información, consulte [UpdateWebAcl](#) la Referencia de AWS CLI comandos.

## update-xss-match-set

En el siguiente ejemplo de código, se muestra cómo usar `update-xss-match-set`.

### AWS CLI

Para actualizar un `XSSMatchSet`

El siguiente `update-xss-match-set` comando elimina un `XssMatchTuple` objeto (filtros) de un `XssMatchSet`. Como el `updates` valor contiene comillas dobles incrustadas, debe escribir todo el valor entre comillas simples.

```
aws waf-regional update-xss-match-set \  
  --xss-match-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \  
  --updates '{Type="ALLOW"}'
```

```
--change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \  
--updates  
'Action="DELETE",XssMatchTuple={FieldToMatch={Type="QUERY_STRING"},TextTransformation="URL_
```

Para obtener más información, consulte Cómo [trabajar con condiciones de coincidencia de secuencias de comandos entre sitios](#) en la guía AWS WAF para desarrolladores.

- Para API obtener más información, consulte la Referencia [UpdateXssMatchSet](#) de AWS CLI comandos.

## AWS WAFV2 ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with AWS WAFV2.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **associate-web-acl**

En el siguiente ejemplo de código, se muestra cómo usar `associate-web-acl`.

AWS CLI

Para asociar una web ACL a un AWS recurso regional

El siguiente `associate-web-acl` ejemplo asocia la web ACL especificada a un Application Load Balancer.

```
aws wafv2 associate-web-acl \  
--web-acl-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test-cli/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  

```

```
--resource-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/  
app/waf-cli-alb/1ea17125f8b25a2a \  
--region us-west-2
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Asociación o desasociación de una Web ACL a un AWS recurso](#) en la Guía avanzada para desarrolladores de AWS Firewall AWS WAF Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia de [AssociateWebAcl](#) comandos AWS CLI .

## check-capacity

En el siguiente ejemplo de código, se muestra cómo usar `check-capacity`.

### AWS CLI

Para obtener la capacidad utilizada por un conjunto de reglas

A continuación, se `check-capacity` recuperan los requisitos de capacidad de un conjunto de reglas que contiene una declaración de reglas basada en tasas y una declaración de AND reglas que contiene reglas anidadas.

```
aws wafv2 check-capacity \  
  --scope REGIONAL \  
  --rules file://waf-rule-list.json \  
  --region us-west-2
```

Contenido del archivo://.json: waf-rule-list

```
[  
  {  
    "Name":"basic-rule",  
    "Priority":0,  
    "Statement":{  
      "AndStatement":{  
        "Statements":[  
          {  
            "ByteMatchStatement":{  
              "SearchString":"example.com",
```

```
        "FieldToMatch":{
            "SingleHeader":{
                "Name":"host"
            }
        },
        "TextTransformations":[
            {
                "Priority":0,
                "Type":"LOWERCASE"
            }
        ],
        "PositionalConstraint":"EXACTLY"
    }
},
{
    "GeoMatchStatement":{
        "CountryCodes":[
            "US",
            "IN"
        ]
    }
}
]
}
},
"Action":{
    "Allow":{

    }
},
"VisibilityConfig":{
    "SampledRequestsEnabled":true,
    "CloudWatchMetricsEnabled":true,
    "MetricName":"basic-rule"
}
},
{
    "Name":"rate-rule",
    "Priority":1,
    "Statement":{
        "RateBasedStatement":{
            "Limit":1000,
            "AggregateKeyType":"IP"
        }
    }
}
```

```

    },
    "Action":{
      "Block":{

      }
    },
    "VisibilityConfig":{
      "SampledRequestsEnabled":true,
      "CloudWatchMetricsEnabled":true,
      "MetricName":"rate-rule"
    }
  }
]

```

Salida:

```

{
  "Capacity":15
}

```

Para obtener más información, consulte [Unidades de ACL capacidad AWS WAF web \(WCU\)](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [CheckCapacity](#) la Referencia de AWS CLI comandos.

## create-ip-set

En el siguiente ejemplo de código, se muestra cómo usar `create-ip-set`.

### AWS CLI

Para crear un conjunto de IP para usarlo en su web ACLs y en sus grupos de reglas

El siguiente `create-ip-set` comando crea un conjunto de direcciones IP con una única especificación de rango de direcciones.

```

aws wafv2 create-ip-set \
  --name testip \
  --scope REGIONAL \
  --ip-address-version IPV4 \
  --addresses 198.51.100.0/16

```

Salida:

```
{
  "Summary":{
    "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/ipset/testip/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Description":"",
    "Name":"testip",
    "LockToken":"447e55ac-0000-0000-0000-86b67c17f8b5",
    "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}
```

Para obtener más información, consulte Conjuntos de [IP y conjuntos de patrones de expresiones regulares](#) en la Guía AWS WAFavanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia [CreateIpSet](#) de AWS CLI comandos.

## create-regex-pattern-set

En el siguiente ejemplo de código, se muestra cómo usar create-regex-pattern-set.

### AWS CLI

Para crear un conjunto de patrones de expresiones regulares para su uso en la web ACLs y en sus grupos de reglas

El siguiente create-regex-pattern-set comando crea un conjunto de patrones de expresiones regulares con dos patrones de expresiones regulares especificados.

```
aws wafv2 create-regex-pattern-set \
  --name regexPatterSet01 \
  --scope REGIONAL \
  --description 'Test web-acl' \
  --regular-expression-list '["RegexString": "/[0-9]*/"],{"RegexString": "/[a-z]*/"}']'
```

Salida:

```
{
  "Summary":{
```

```

    "ARN": "arn:aws:wafv2:us-west-2:123456789012:regional/regexpatternset/
    regexPatterSet01/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Description": "Test web-acl",
    "Name": "regexPatterSet01",
    "LockToken": "0bc01e21-03c9-4b98-9433-6229cbf1ef1c",
    "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}

```

Para obtener más información, consulte Conjuntos de [IP y conjuntos de patrones de expresiones regulares](#) en la Guía AWS WAFavanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia [CreateRegexPatternSet](#) de AWS CLI comandos.

## create-rule-group

En el siguiente ejemplo de código, se muestra cómo usar `create-rule-group`.

### AWS CLI

Para crear un grupo de reglas personalizado para su uso en la web ACLs

El siguiente `create-rule-group` comando crea un grupo de reglas personalizado para uso regional. Las declaraciones de reglas del grupo se proporcionan en un archivo JSON con formato.

```

aws wafv2 create-rule-group \
  --name "TestRuleGroup" \
  --scope REGIONAL \
  --capacity 250 \
  --rules file://waf-rule.json \
  --visibility-
config SampledRequestsEnabled=true,CloudWatchMetricsEnabled=true,MetricName=TestRuleGroupMet
\
  --region us-west-2

```

Contenido del archivo: `//waf-rule.json`:

```

[
  {
    "Name": "basic-rule",

```



```
"Priority":0,
"Statement":{
  "AndStatement":{
    "Statements":[
      {
        "ByteMatchStatement":{
          "SearchString":"example.com",
          "FieldToMatch":{
            "SingleHeader":{
              "Name":"host"
            }
          },
          "TextTransformations":[
            {
              "Priority":0,
              "Type":"LOWERCASE"
            }
          ],
          "PositionalConstraint":"EXACTLY"
        }
      },
      {
        "GeoMatchStatement":{
          "CountryCodes":[
            "US",
            "IN"
          ]
        }
      }
    ]
  }
},
"Action":{
  "Allow":{
  }
},
"VisibilityConfig":{
  "SampledRequestsEnabled":true,
  "CloudWatchMetricsEnabled":true,
  "MetricName":"basic-rule"
}
}
```

```
]

```

Salida:

```
{
  "Summary":{
    "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/rulegroup/
TestRuleGroup/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Description":"",
    "Name":"TestRuleGroup",
    "LockToken":"7b3bcec2-374e-4c5a-b2b9-563bf47249f0",
    "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}
```

Para obtener más información, consulte [Administrar sus propios grupos de reglas](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [CreateRuleGroup](#) la Referencia de AWS CLI comandos.

## create-web-acl

En el siguiente ejemplo de código, se muestra cómo usar `create-web-acl`.

### AWS CLI

Para crear una web ACL

El siguiente `create-web-acl` comando crea una web ACL para uso regional. Las declaraciones de reglas para la web ACL se proporcionan en un archivo JSON con formato.

```
aws wafv2 create-web-acl \
  --name TestWebAcl \
  --scope REGIONAL \
  --default-action Allow={} \
  --visibility-
config SampledRequestsEnabled=true,CloudWatchMetricsEnabled=true,MetricName=TestWebAclMetric
\
  --rules file://waf-rule.json \
  --region us-west-2
```

## Contenido del archivo: //waf-rule.json:

```
[
  {
    "Name": "basic-rule",
    "Priority": 0,
    "Statement": {
      "AndStatement": {
        "Statements": [
          {
            "ByteMatchStatement": {
              "SearchString": "example.com",
              "FieldToMatch": {
                "SingleHeader": {
                  "Name": "host"
                }
              },
              "TextTransformations": [
                {
                  "Priority": 0,
                  "Type": "LOWERCASE"
                }
              ],
              "PositionalConstraint": "EXACTLY"
            },
            {
              "GeoMatchStatement": {
                "CountryCodes": [
                  "US",
                  "IN"
                ]
              }
            }
          ]
        }
      },
      "Action": {
        "Allow": {
        }
      },
      "VisibilityConfig": {
        "SampledRequestsEnabled": true,

```

```

        "CloudWatchMetricsEnabled":true,
        "MetricName":"basic-rule"
    }
}
]

```

Salida:

```

{
  "Summary":{
    "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/TestWebAcl/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Description":"",
    "Name":"TestWebAcl",
    "LockToken":"2294b3a1-eb60-4aa0-a86f-a3ae04329de9",
    "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}

```

Para obtener más información, consulte [Administración y uso de una lista de control de acceso a la web \(WebACL\)](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager and AWS Shield.

- Para API obtener más información, consulte [CreateWebAcl](#) la Referencia de AWS CLI comandos.

## delete-ip-set

En el siguiente ejemplo de código, se muestra cómo usar delete-ip-set.

### AWS CLI

Para eliminar un conjunto de direcciones IP

Lo siguiente delete-ip-set elimina el conjunto de IP especificado. Esta llamada requiere un identificador, que puede obtener de la llamada list-ip-sets, y un token de bloqueo, que puede obtener de las llamadas, list-ip-sets y get-ip-set.

```

aws wafv2 delete-ip-set \
  --name test1 \
  --scope REGIONAL \

```

```
--id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
--lock-token 46851772-db6f-459d-9385-49428812e357
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Conjuntos de [IP y conjuntos de patrones de expresiones regulares](#) en la Guía AWS WAFavanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia [DeleteIpSet](#) de AWS CLI comandos.

## **delete-logging-configuration**

En el siguiente ejemplo de código, se muestra cómo usar `delete-logging-configuration`.

### AWS CLI

Para deshabilitar el registro en una web ACL

Lo siguiente `delete-logging-configuration` elimina cualquier configuración de registro de la web especificadaACL.

```
aws wafv2 delete-logging-configuration \  
--resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Registro de información de ACL tráfico web](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [DeleteLoggingConfiguration](#) la Referencia de AWS CLI comandos.

## **delete-regex-pattern-set**

En el siguiente ejemplo de código, se muestra cómo usar `delete-regex-pattern-set`.

### AWS CLI

Para eliminar un conjunto de patrones de expresiones regulares

A continuación, se `delete-regex-pattern-set` actualiza la configuración del conjunto de patrones de expresiones regulares especificado. Esta llamada requiere un identificador, que puede obtener de la llamada `list-regex-pattern-sets`, y un identificador de bloqueo, que puede obtener de la llamada `list-regex-pattern-sets` o de la llamada `get-regex-pattern-set`

```
aws wafv2 delete-regex-pattern-set \  
  --name regexPatterSet01 \  
  --scope REGIONAL \  
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --lock-token 0bc01e21-03c9-4b98-9433-6229cbf1ef1c
```

Este comando no genera ninguna salida.

Para obtener más información, consulte Conjuntos de [IP y conjuntos de patrones de expresiones regulares](#) en la Guía AWS WAFavanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia [DeleteRegexPatternSet](#) de AWS CLI comandos.

## **delete-rule-group**

En el siguiente ejemplo de código, se muestra cómo usar `delete-rule-group`.

### AWS CLI

Para eliminar un grupo de reglas personalizado

Lo siguiente `delete-rule-group` elimina el grupo de reglas personalizado especificado. Esta llamada requiere un identificador, que puede obtener de la llamada `list-rule-groups`, y un token de bloqueo, que puede obtener de la llamada `list-rule-groups` o de la llamada `get-rule-group`.

```
aws wafv2 delete-rule-group \  
  --name TestRuleGroup \  
  --scope REGIONAL \  
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --lock-token 7b3bcec2-0000-0000-0000-563bf47249f0
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administrar sus propios grupos de reglas](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [DeleteRuleGroup](#) la Referencia de AWS CLI comandos.

## **delete-web-acl**

En el siguiente ejemplo de código, se muestra cómo usar `delete-web-acl`.

### AWS CLI

Para eliminar una web ACL

Lo siguiente `delete-web-acl` elimina la web especificada ACL de tu cuenta. Una web solo se ACL puede eliminar si no está asociada a ningún recurso. Esta llamada requiere un identificador, que puedes obtener de la llamada `list-web-acls`, y un token de bloqueo, que puedes obtener de la llamada `list-web-acls` o de la llamada `get-web-acl`.

```
aws wafv2 delete-web-acl \  
  --name test \  
  --scope REGIONAL \  
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --lock-token ebab4ed2-155e-4c9a-9efb-e4c45665b1f5
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Administración y uso de una lista de control de acceso a la web \(WebACL\)](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager and AWS Shield.

- Para API obtener más información, consulte [DeleteWebAcl](#) la Referencia de AWS CLI comandos.

## **describe-managed-rule-group**

En el siguiente ejemplo de código, se muestra cómo usar `describe-managed-rule-group`.

### AWS CLI

Para recuperar la descripción de un grupo de reglas administrado

A continuación, se describe-`managed-rule-group` recupera la descripción de un grupo de reglas AWS administrado.

```
aws wafv2 describe-managed-rule-group \  
  --vendor-name AWS \  
  --name AWSManagedRulesCommonRuleSet \  
  --scope REGIONAL
```

Salida:

```
{  
  "Capacity": 700,  
  "Rules": [  
    {  
      "Name": "NoUserAgent_HEADER",  
      "Action": {  
        "Block": {}  
      }  
    },  
    {  
      "Name": "UserAgent_BadBots_HEADER",  
      "Action": {  
        "Block": {}  
      }  
    },  
    {  
      "Name": "SizeRestrictions_QUERYSTRING",  
      "Action": {  
        "Block": {}  
      }  
    },  
    {  
      "Name": "SizeRestrictions_Cookie_HEADER",  
      "Action": {  
        "Block": {}  
      }  
    },  
    {  
      "Name": "SizeRestrictions_BODY",  
      "Action": {  
        "Block": {}  
      }  
    }  
  ],  
}
```



```
{
  "Name": "SizeRestrictions_URI_PATH",
  "Action": {
    "Block": {}
  }
},
{
  "Name": "EC2Meta-data-SSRF_BODY",
  "Action": {
    "Block": {}
  }
},
{
  "Name": "EC2Meta-data-SSRF_COOKIE",
  "Action": {
    "Block": {}
  }
},
{
  "Name": "EC2Meta-data-SSRF_URI_PATH",
  "Action": {
    "Block": {}
  }
},
{
  "Name": "EC2Meta-data-SSRF_QUERY_ARGUMENTS",
  "Action": {
    "Block": {}
  }
},
{
  "Name": "GenericLFI_QUERY_ARGUMENTS",
  "Action": {
    "Block": {}
  }
},
{
  "Name": "GenericLFI_URI_PATH",
  "Action": {
    "Block": {}
  }
},
{
```

```
    "Name": "GenericLFI_BODY",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "RestrictedExtensions_URI_PATH",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "RestrictedExtensions_QUERY_ARGUMENTS",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "GenericRFI_QUERY_ARGUMENTS",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "GenericRFI_BODY",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "GenericRFI_URI_PATH",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "CrossSiteScripting_COOKIE",
    "Action": {
      "Block": {}
    }
  },
  {
    "Name": "CrossSiteScripting_QUERY_ARGUMENTS",
    "Action": {
```

```

        "Block": {}
    },
    {
        "Name": "CrossSiteScripting_BODY",
        "Action": {
            "Block": {}
        }
    },
    {
        "Name": "CrossSiteScripting_URI_PATH",
        "Action": {
            "Block": {}
        }
    }
]
}

```

Para obtener más información, consulte [los grupos de reglas gestionados](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [DescribeManagedRuleGroup](#) la Referencia de AWS CLI comandos.

## disassociate-web-acl

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-web-acl`.

### AWS CLI

Para desasociar una web ACL de un recurso regional AWS

En el siguiente `disassociate-web-acl` ejemplo, se elimina cualquier ACL asociación web existente del Application Load Balancer especificado.

```

aws wafv2 disassociate-web-acl \
  --resource-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/  

app/waf-cli-alb/1ea17125f8b25a2a \
  --region us-west-2

```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Asociación o desasociación de una Web ACL a un AWS recurso](#) en la Guía avanzada para desarrolladores de AWS Firewall AWS WAF Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia de [DisassociateWebAcl](#) comandos AWS CLI .

## get-ip-set

En el siguiente ejemplo de código, se muestra cómo usar `get-ip-set`.

### AWS CLI

Para recuperar un conjunto de IP específico

A continuación, se `get-ip-set` recupera el conjunto de direcciones IP con el nombre, el alcance y el ID especificados. Puede obtener el ID de un conjunto de direcciones IP mediante los comandos `create-ip-set` y `list-ip-sets`.

```
aws wafv2 get-ip-set \
  --name testip \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{
  "IPSet":{
    "Description":"","
    "Name":"testip",
    "IPAddressVersion":"IPV4",
    "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE1111",
    "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/ipset/testip/
a1b2c3d4-5678-90ab-cdef-EXAMPLE1111",
    "Addresses":[
      "192.0.2.0/16"
    ]
  },
  "LockToken":"447e55ac-2396-4c6d-b9f9-86b67c17f8b5"
}
```

Para obtener más información, consulte Conjuntos de [IP y conjuntos de patrones de expresiones regulares](#) en la Guía AWS WAFavanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia [GetIpSet](#) de AWS CLI comandos.

## get-logging-configuration

En el siguiente ejemplo de código, se muestra cómo usar `get-logging-configuration`.

### AWS CLI

Para recuperar las configuraciones de registro de una web ACL

A continuación, se `get-logging-configuration` recupera la configuración de registro de la web ACL especificada.

```
aws wafv2 get-logging-configuration \  
  --resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222 \  
  --region us-west-2
```

Salida:

```
{  
  "LoggingConfiguration":{  
    "ResourceArn":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
    "RedactedFields":[  
      {  
        "Method":{  
          }  
      }  
    ],  
    "LogDestinationConfigs":[  
      "arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-logs-  
custom-transformation"  
    ]  
  }  
}
```

Para obtener más información, consulte [Registro de información de ACL tráfico web](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [GetLoggingConfiguration](#) la Referencia de AWS CLI comandos.

## get-rate-based-statement-managed-keys

En el siguiente ejemplo de código, se muestra cómo usar `get-rate-based-statement-managed-keys`.

### AWS CLI

Para recuperar una lista de direcciones IP que están bloqueadas por una regla basada en la velocidad

A continuación, se `get-rate-based-statement-managed-keys` recuperan las direcciones IP actualmente bloqueadas por una regla basada en tasas que se utiliza para una aplicación regional.

```
aws wafv2 get-rate-based-statement-managed-keys \
  --scope REGIONAL \
  --web-acl-name testwebacl2 \
  --web-acl-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --rule-name ratebasedtest
```

Salida:

```
{
  "ManagedKeysIPV4":{
    "IPAddressVersion":"IPV4",
    "Addresses":[
      "198.51.100.0/32"
    ]
  },
  "ManagedKeysIPV6":{
    "IPAddressVersion":"IPV6",
    "Addresses":[
    ]
  }
}
```

Para obtener más información, consulte la [Declaración de reglas basadas en la velocidad](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [GetRateBasedStatementManagedKeys](#) la Referencia de AWS CLI comandos.

## get-regex-pattern-set

En el siguiente ejemplo de código, se muestra cómo usar `get-regex-pattern-set`.

### AWS CLI

Para recuperar un conjunto de patrones de expresiones regulares específico

A continuación, se `get-regex-pattern-set` recupera el conjunto de patrones de expresiones regulares con el nombre, el ámbito, la región y el ID especificados. Puede obtener el ID de un conjunto de patrones de expresiones regulares mediante los comandos `y. create-regex-pattern-set list-regex-pattern-sets`

```
aws wafv2 get-regex-pattern-set \
  --name regexPatterSet01 \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --region us-west-2
```

Salida:

```
{
  "RegexPatternSet": {
    "Description": "Test web-acl",
    "RegularExpressionList": [
      {
        "RegexString": "/[0-9]*/"
      },
      {
        "RegexString": "/[a-z]*/"
      }
    ],
    "Name": "regexPatterSet01",
    "ARN": "arn:aws:wafv2:us-west-2:123456789012:regional/regexpatternset/regexPatterSet01/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  }
}
```

```
  },
  "LockToken":"c8abf33f-b6fc-46ae-846e-42f994d57b29"
}
```

Para obtener más información, consulte Conjuntos de [IP y conjuntos de patrones de expresiones regulares](#) en la Guía AWS WAFavanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia [GetRegexPatternSet](#) de AWS CLI comandos.

## get-rule-group

En el siguiente ejemplo de código, se muestra cómo usar `get-rule-group`.

### AWS CLI

Para recuperar un grupo de reglas personalizado específico

A continuación, se `get-rule-group` recupera el grupo de reglas personalizado con el nombre, el ámbito y el ID especificados. Puede obtener el ID de un grupo de reglas a partir de los comandos `create-rule-group` y `list-rule-groups`.

```
aws wafv2 get-rule-group \
  --name ff \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{
  "RuleGroup":{
    "Capacity":1,
    "Description":"","
    "Rules":[
      {
        "Priority":0,
        "Action":{
          "Block":{

```



```

    },
    "VisibilityConfig":{
      "SampledRequestsEnabled":true,
      "CloudWatchMetricsEnabled":true,
      "MetricName":"jj"
    },
    "Name":"jj",
    "Statement":{
      "SizeConstraintStatement":{
        "ComparisonOperator":"LE",
        "TextTransformations":[
          {
            "Priority":0,
            "Type":"NONE"
          }
        ],
        "FieldToMatch":{
          "UriPath":{

          }
        },
        "Size":7
      }
    }
  ],
  "VisibilityConfig":{
    "SampledRequestsEnabled":true,
    "CloudWatchMetricsEnabled":true,
    "MetricName":"ff"
  },
  "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/rulegroup/ff/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "Name":"ff"
},
"LockToken":"485458c9-1830-4234-af31-ec4d52ced1b3"
}

```

Para obtener más información, consulte [Administrar sus propios grupos de reglas](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [GetRuleGroup](#) la Referencia de AWS CLI comandos.

## get-sampled-requests

En el siguiente ejemplo de código, se muestra cómo usar `get-sampled-requests`.

### AWS CLI

Para recuperar una muestra de solicitudes web para una web ACL

A continuación, se `get-sampled-requests` recuperan las solicitudes web muestreadas para la webACL, la métrica de regla y el período de tiempo especificados.

```
aws wafv2 get-sampled-requests \  
  --web-acl-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test-cli/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --rule-metric-name AWS-AWSManagedRulesSQLiRuleSet \  
  --scope=REGIONAL \  
  --time-window StartTime=2020-02-12T20:00Z,EndTime=2020-02-12T21:10Z \  
  --max-items 100
```

Salida:

```
{  
  "TimeWindow": {  
    "EndTime": 1581541800.0,  
    "StartTime": 1581537600.0  
  },  
  "SampledRequests": [  
    {  
      "Action": "BLOCK",  
      "Timestamp": 1581541799.564,  
      "RuleNameWithinRuleGroup": "AWS#AWSManagedRulesSQLiRuleSet#SQLi_BODY",  
      "Request": {  
        "Country": "US",  
        "URI": "/",  
        "Headers": [  
          {  
            "Name": "Host",  
            "Value": "alb-test-1EXAMPLE1.us-east-1.elb.amazonaws.com"  
          },  
          {  
            "Name": "Content-Length",  
            "Value": "7456"  
          }  
        ]  
      }  
    }  
  ]  
}
```

```
        {
            "Name": "User-Agent",
            "Value": "curl/7.53.1"
        },
        {
            "Name": "Accept",
            "Value": "/"
        },
        {
            "Name": "Content-Type",
            "Value": "application/x-www-form-urlencoded"
        }
    ],
    "ClientIP": "198.51.100.08",
    "Method": "POST",
    "HTTPVersion": "HTTP/1.1"
},
"Weight": 1
},
{
    "Action": "BLOCK",
    "Timestamp": 1581541799.988,
    "RuleNameWithinRuleGroup": "AWS#AWSManagedRulesSQLiRuleSet#SQLi_BODY",
    "Request": {
        "Country": "US",
        "URI": "/",
        "Headers": [
            {
                "Name": "Host",
                "Value": "alb-test-1EXAMPLE1.us-east-1.elb.amazonaws.com"
            },
            {
                "Name": "Content-Length",
                "Value": "7456"
            },
            {
                "Name": "User-Agent",
                "Value": "curl/7.53.1"
            },
            {
                "Name": "Accept",
                "Value": "/"
            }
        ]
    }
}
```

```
        "Name": "Content-Type",
        "Value": "application/x-www-form-urlencoded"
    }
  ],
  "ClientIP": "198.51.100.08",
  "Method": "POST",
  "HTTPVersion": "HTTP/1.1"
},
"Weight": 3
},
{
  "Action": "BLOCK",
  "Timestamp": 1581541799.846,
  "RuleNameWithinRuleGroup": "AWS#AWSManagedRulesSQLiRuleSet#SQLi_BODY",
  "Request": {
    "Country": "US",
    "URI": "/",
    "Headers": [
      {
        "Name": "Host",
        "Value": "alb-test-1EXAMPLE1.us-east-1.elb.amazonaws.com"
      },
      {
        "Name": "Content-Length",
        "Value": "7456"
      },
      {
        "Name": "User-Agent",
        "Value": "curl/7.53.1"
      },
      {
        "Name": "Accept",
        "Value": "/"
      },
      {
        "Name": "Content-Type",
        "Value": "application/x-www-form-urlencoded"
      }
    ],
    "ClientIP": "198.51.100.08",
    "Method": "POST",
    "HTTPVersion": "HTTP/1.1"
  },
  "Weight": 1
}
```

```
    },
    {
      "Action": "BLOCK",
      "Timestamp": 1581541799.4,
      "RuleNameWithinRuleGroup": "AWS#AWSManagedRulesSQLiRuleSet#SQLi_BODY",
      "Request": {
        "Country": "US",
        "URI": "/",
        "Headers": [
          {
            "Name": "Host",
            "Value": "alb-test-1EXAMPLE1.us-east-1.elb.amazonaws.com"
          },
          {
            "Name": "Content-Length",
            "Value": "7456"
          },
          {
            "Name": "User-Agent",
            "Value": "curl/7.53.1"
          },
          {
            "Name": "Accept",
            "Value": "/"
          },
          {
            "Name": "Content-Type",
            "Value": "application/x-www-form-urlencoded"
          }
        ],
        "ClientIP": "198.51.100.08",
        "Method": "POST",
        "HTTPVersion": "HTTP/1.1"
      },
      "Weight": 1
    }
  ],
  "PopulationSize": 4
}
```

Para obtener más información, consulte [Visualización de un ejemplo de solicitudes web](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [GetSampledRequests](#) la Referencia de AWS CLI comandos.

## get-web-acl-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `get-web-acl-for-resource`.

### AWS CLI

Para recuperar la web ACL asociada a un AWS recurso

A continuación, se `get-web-acl-for-resource` recupera JSON la información de ACL la web asociada al recurso especificado.

```
aws wafv2 get-web-acl-for-resource \
  --resource-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/
  app/waf-cli-alb/1ea17125f8b25a2a
```

Salida:

```
{
  "WebACL":{
    "Capacity":3,
    "Description":"",
    "Rules":[
      {
        "Priority":1,
        "Action":{
          "Block":{

          }
        },
        "VisibilityConfig":{
          "SampledRequestsEnabled":true,
          "CloudWatchMetricsEnabled":true,
          "MetricName":"testrule01"
        },
        "Name":"testrule01",
        "Statement":{
          "AndStatement":{
            "Statements":[
              {
```

```

        "ByteMatchStatement":{
            "PositionalConstraint":"EXACTLY",
            "TextTransformations":[
                {
                    "Priority":0,
                    "Type":"NONE"
                }
            ],
            "SearchString":"dGVzdHN0cmluZw==",
            "FieldToMatch":{
                "UriPath":{

                }
            }
        },
        {
            "SizeConstraintStatement":{
                "ComparisonOperator":"EQ",
                "TextTransformations":[
                    {
                        "Priority":0,
                        "Type":"NONE"
                    }
                ],
                "FieldToMatch":{
                    "QueryString":{

                    }
                },
                "Size":0
            }
        }
    ]
},
"VisibilityConfig":{
    "SampledRequestsEnabled":true,
    "CloudWatchMetricsEnabled":true,
    "MetricName":"test01"
},
"DefaultAction":{

```

```

        "Allow":{
            }
        },
        "Id":"9a1b2c3d4-5678-90ab-cdef-EXAMPLE11111  ",
        "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test01/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111  ",
        "Name":"test01"
    }
}

```

Para obtener más información, consulte [Asociación o desasociación de una Web ACL a un AWS recurso](#) en la Guía avanzada para desarrolladores de AWS Firewall AWS WAF Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia de [GetWebAclForResource](#) comandos AWS CLI .

## get-web-acl

En el siguiente ejemplo de código, se muestra cómo usar `get-web-acl`.

### AWS CLI

Para recuperar una web ACL

A continuación, se `get-web-acl` recupera la web ACL con el nombre, el alcance y el ID especificados. Puede obtener el ID de una web con ACL los comandos `create-web-acl` y `list-web-acls`.

```

aws wafv2 get-web-acl \
  --name test01 \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

```

Salida:

```

{
  "WebACL":{
    "Capacity":3,
    "Description":"",
    "Rules":[

```



```
{
  "Priority":1,
  "Action":{
    "Block":{

    }
  },
  "VisibilityConfig":{
    "SampledRequestsEnabled":true,
    "CloudWatchMetricsEnabled":true,
    "MetricName":"testrule01"
  },
  "Name":"testrule01",
  "Statement":{
    "AndStatement":{
      "Statements":[
        {
          "ByteMatchStatement":{
            "PositionalConstraint":"EXACTLY",
            "TextTransformations":[
              {
                "Priority":0,
                "Type":"NONE"
              }
            ],
            "SearchString":"dGVzdHN0cmVuZw==",
            "FieldToMatch":{
              "UriPath":{

              }
            }
          }
        },
        {
          "SizeConstraintStatement":{
            "ComparisonOperator":"EQ",
            "TextTransformations":[
              {
                "Priority":0,
                "Type":"NONE"
              }
            ],
            "FieldToMatch":{
              "QueryString":{

              }
            }
          }
        }
      ]
    }
  }
}
```

```

    }
  },
  "Size":0
}
]
}
}
},
"VisibilityConfig":{
  "SampledRequestsEnabled":true,
  "CloudWatchMetricsEnabled":true,
  "MetricName":"test01"
},
"DefaultAction":{
  "Allow":{

  }
},
"Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test01/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"Name":"test01"
},
"LockToken":"e3db7e2c-d58b-4ee6-8346-6aec5511c6fb"
}

```

Para obtener más información, consulte [Administración y uso de una lista de control de acceso a la web \(WebACL\)](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager and AWS Shield.

- Para API obtener más información, consulte [GetWebAcl](#) la Referencia de AWS CLI comandos.

## list-available-managed-rule-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-available-managed-rule-groups`.

### AWS CLI

Para recuperar los grupos de reglas gestionados

A continuación, `list-available-managed-rule-groups` se muestra la lista de todos los grupos de reglas administrados que están disponibles actualmente para su uso en la webACLs.

```
aws wafv2 list-available-managed-rule-groups \
  --scope REGIONAL
```

Salida:

```
{
  "ManagedRuleGroups": [
    {
      "VendorName": "AWS",
      "Name": "AWSManagedRulesCommonRuleSet",
      "Description": "Contains rules that are generally applicable to web applications. This provides protection against exploitation of a wide range of vulnerabilities, including those described in OWASP publications and common Common Vulnerabilities and Exposures (CVE).",
    },
    {
      "VendorName": "AWS",
      "Name": "AWSManagedRulesAdminProtectionRuleSet",
      "Description": "Contains rules that allow you to block external access to exposed admin pages. This may be useful if you are running third-party software or would like to reduce the risk of a malicious actor gaining administrative access to your application.",
    },
    {
      "VendorName": "AWS",
      "Name": "AWSManagedRulesKnownBadInputsRuleSet",
      "Description": "Contains rules that allow you to block request patterns that are known to be invalid and are associated with exploitation or discovery of vulnerabilities. This can help reduce the risk of a malicious actor discovering a vulnerable application.",
    },
    {
      "VendorName": "AWS",
      "Name": "AWSManagedRulesSQLiRuleSet",
      "Description": "Contains rules that allow you to block request patterns associated with exploitation of SQL databases, like SQL injection attacks. This can help prevent remote injection of unauthorized queries.",
    },
    {
      "VendorName": "AWS",
```

```
    "Name": "AWSManagedRulesLinuxRuleSet",
    "Description": "Contains rules that block request patterns associated
with exploitation of vulnerabilities specific to Linux, including LFI attacks. This
can help prevent attacks that expose file contents or execute code for which the
attacker should not have had access."
  },
  {
    "VendorName": "AWS",
    "Name": "AWSManagedRulesUnixRuleSet",
    "Description": "Contains rules that block request patterns associated
with exploiting vulnerabilities specific to POSIX/POSIX-like OS, including LFI
attacks. This can help prevent attacks that expose file contents or execute code
for which access should not been allowed."
  },
  {
    "VendorName": "AWS",
    "Name": "AWSManagedRulesWindowsRuleSet",
    "Description": "Contains rules that block request patterns associated
with exploiting vulnerabilities specific to Windows, (e.g., PowerShell commands).
This can help prevent exploits that allow attacker to run unauthorized commands or
execute malicious code."
  },
  {
    "VendorName": "AWS",
    "Name": "AWSManagedRulesPHPRuleSet",
    "Description": "Contains rules that block request patterns associated
with exploiting vulnerabilities specific to the use of the PHP, including injection
of unsafe PHP functions. This can help prevent exploits that allow an attacker to
remotely execute code or commands."
  },
  {
    "VendorName": "AWS",
    "Name": "AWSManagedRulesWordPressRuleSet",
    "Description": "The WordPress Applications group contains rules that
block request patterns associated with the exploitation of vulnerabilities specific
to WordPress sites."
  },
  {
    "VendorName": "AWS",
    "Name": "AWSManagedRulesAmazonIpReputationList",
    "Description": "This group contains rules that are based on Amazon
threat intelligence. This is useful if you would like to block sources associated
with bots or other threats."
  }
}
```

```
]
}
```

Para obtener más información, consulte [los grupos de reglas gestionados](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [ListAvailableManagedRuleGroups](#) la Referencia de AWS CLI comandos.

## list-ip-sets

En el siguiente ejemplo de código, se muestra cómo usar `list-ip-sets`.

### AWS CLI

Para recuperar una lista de conjuntos de IP

A continuación, se `list-ip-sets` recuperan todos los conjuntos de IP de la cuenta que tienen un alcance regional.

```
aws wafv2 list-ip-sets \
  --scope REGIONAL
```

Salida:

```
{
  "IPSets": [
    {
      "ARN": "arn:aws:wafv2:us-west-2:123456789012:regional/ipset/testip/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Description": "",
      "Name": "testip",
      "LockToken": "0674c84b-0304-47fe-8728-c6bff46af8fc",
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 "
    }
  ],
  "NextMarker": "testip"
}
```

Para obtener más información, consulte [Conjuntos de IP y conjuntos de patrones de expresiones regulares](#) en la Guía AWS WAF avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia [ListIpSets](#) de AWS CLI comandos.

## list-logging-configurations

En el siguiente ejemplo de código, se muestra cómo usar `list-logging-configurations`.

### AWS CLI

Para recuperar una lista de todas las configuraciones de registro de una región

A continuación, se `list-logging-configurations` recuperan todas las configuraciones de registro para la web ACLs que están destinadas al uso regional en la `us-west-2` región.

```
aws wafv2 list-logging-configurations \  
  --scope REGIONAL \  
  --region us-west-2
```

Salida:

```
{  
  "LoggingConfigurations": [  
    {  
      "ResourceArn": "arn:aws:wafv2:us-west-2:123456789012:regional/webacl/  
test-2/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "RedactedFields": [  
        {  
          "QueryString": {  
            }  
        }  
      ],  
      "LogDestinationConfigs": [  
        "arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-  
logs-test"  
      ]  
    },  
    {  
      "ResourceArn": "arn:aws:wafv2:us-west-2:123456789012:regional/webacl/  
test/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "RedactedFields": [  
        {  
          "Method": {  
            }  
        }  
      ]  
    }  
  ]  
}
```

```

    }
  ],
  "LogDestinationConfigs": [
    "arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-logs-custom-transformation"
  ]
}
]
}

```

Para obtener más información, consulte [Registro de información de ACL tráfico web](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [ListLoggingConfigurations](#) la Referencia de AWS CLI comandos.

## list-regex-pattern-sets

En el siguiente ejemplo de código, se muestra cómo usar `list-regex-pattern-sets`.

### AWS CLI

Para recuperar una lista de conjuntos de patrones de expresiones regulares

A continuación, se `list-regex-pattern-sets` recuperan todos los conjuntos de patrones de expresiones regulares de la cuenta que están definidos en la región. `us-west-2`

```

aws wafv2 list-regex-pattern-sets \
--scope REGIONAL \
--region us-west-2

```

Salida:

```

{
  "NextMarker": "regexPatterSet01",
  "RegexPatternSets": [
    {
      "ARN": "arn:aws:wafv2:us-west-2:123456789012:regional/regexpatternset/regexPatterSet01/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
      "Description": "Test web-acl",
      "Name": "regexPatterSet01",
    }
  ]
}

```

```

        "LockToken": "f17743f7-0000-0000-0000-19a8b93bfb01",
        "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
]
}

```

Para obtener más información, consulte Conjuntos de [IP y conjuntos de patrones de expresiones regulares](#) en la Guía AWS WAFavanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia [ListRegexPatternSets](#) de AWS CLI comandos.

## list-resources-for-web-acl

En el siguiente ejemplo de código, se muestra cómo usar `list-resources-for-web-acl`.

### AWS CLI

Para recuperar los recursos asociados a una web ACL

A continuación, se `list-resources-for-web-acl` recuperan REST API los recursos de API Gateway que están actualmente asociados a la web especificada ACL en la región `us-west-2`.

```

aws wafv2 list-resources-for-web-acl \
  --web-acl-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/TestWebAcl/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --resource-type API_GATEWAY \
  --region us-west-2

```

Salida:

```

{
  "ResourceArns": [
    "arn:aws:apigateway:us-west-2::/restapis/EXAMPLE111/stages/testing"
  ]
}

```

Para obtener más información, consulte [Asociación o desasociación de una Web ACL a un AWS recurso](#) en la Guía avanzada para desarrolladores de AWS Firewall AWS WAF Manager y AWS Shield.



- Para API obtener más información, consulte la Referencia de [ListResourcesForWebACL](#) comandos AWS CLI .

## list-rule-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-rule-groups`.

### AWS CLI

Para recuperar una lista de grupos de reglas personalizados

A continuación, se `list-rule-groups` recuperan todos los grupos de reglas personalizados que se han definido para la cuenta en el ámbito y la ubicación de la región especificados.

```
aws wafv2 list-rule-groups \  
  --scope REGIONAL \  
  --region us-west-2
```

Salida:

```
{  
  "RuleGroups": [  
    {  
      "ARN": "arn:aws:wafv2:us-west-2:123456789012:regional/rulegroup/  
TestRuleGroup/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "Description": "",  
      "Name": "TestRuleGroup",  
      "LockToken": "1eb5ec48-0000-0000-0000-ee9b906c541e",  
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
    },  
    {  
      "ARN": "arn:aws:wafv2:us-west-2:123456789012:regional/rulegroup/test/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",  
      "Description": "",  
      "Name": "test",  
      "LockToken": "b0f4583e-998b-4880-9069-3fbe45738b43",  
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"  
    }  
  ],  
  "NextMarker": "test"  
}
```

Para obtener más información, consulte [Administrar sus propios grupos de reglas](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [ListRuleGroups](#)la Referencia de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

### AWS CLI

Para recuperar todas las etiquetas de un AWS WAF recurso

A continuación, se `list-tags-for-resource` recupera la lista de todos los pares de etiquetas, claves y valores de la web ACL especificada.

```
aws wafv2 list-tags-for-resource \
  --resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/testwebacl2/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Salida:

```
{
  "NextMarker": "",
  "TagInfoForResource": {
    "ResourceARN": "arn:aws:wafv2:us-west-2:123456789012:regional/webacl/
testwebacl2/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "TagList": [
      ]
    ]
  }
}
```

Para obtener más información, consulte [Introducción a AWS WAF](#) la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [ListTagsForResource](#)la Referencia de AWS CLI comandos.

## list-web-acls

En el siguiente ejemplo de código, se muestra cómo usar `list-web-acls`.

### AWS CLI

Para recuperar la web ACLs para un osciloscopio

A continuación, `list-web-acls` se recuperan todos ACLs los sitios web definidos para la cuenta del ámbito especificado.

```
aws wafv2 list-web-acls \  
  --scope REGIONAL
```

Salida:

```
{  
  "NextMarker": "Testt",  
  "WebACLs": [  
    {  
      "ARN": "arn:aws:wafv2:us-west-2:123456789012:regional/webacl/Testt/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",  
      "Description": "sssss",  
      "Name": "Testt",  
      "LockToken": "7f36cb30-74ef-4cff-8cd4-a77e1aba1746",  
      "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
    }  
  ]  
}
```

Para obtener más información, consulte [Administración y uso de una lista de control de acceso a la web \(WebACL\)](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager and AWS Shield.

- Para API obtener más información, consulte [ListWebAcls](#) la Referencia de AWS CLI comandos.

## put-logging-configuration

En el siguiente ejemplo de código, se muestra cómo usar `put-logging-configuration`.

### AWS CLI

Para añadir una configuración de registro a una web ACL

A continuación, `put-logging-configuration` se añade la configuración de `aws-waf-logs-custom-transformation` registro de Amazon Kinesis Data Firehose a la web ACL especificada, sin que se eliminen campos de los registros.

```
aws wafv2 put-logging-configuration \
  --logging-configuration ResourceArn=arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test-cli/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111,LogDestinationConfigs=arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-logs-custom-transformation \
  --region us-west-2
```

Salida:

```
{
  "LoggingConfiguration":{
    "ResourceArn":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test-cli/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "LogDestinationConfigs":[
      "arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-logs-custom-transformation"
    ]
  }
}
```

Para obtener más información, consulte [Registro de información de ACL tráfico web](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [PutLoggingConfiguration](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para añadir etiquetas a un AWS WAF recurso

En el siguiente `tag-resource` ejemplo, se agrega una etiqueta con una clave `Name` y un valor establecidos en AWS WAF a la web especificada ACL.

```
aws wafv2 tag-resource \
```

```
--resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/
apiGatewayWebAcl/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
--tags Key=Name, Value=AWSWAF
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Introducción a AWS WAF](#) la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar untag-resource.

### AWS CLI

Para eliminar etiquetas de un AWS WAF recurso

En el siguiente untag-resource ejemplo, se elimina la etiqueta con la clave KeyName de la web especificada ACL.

```
aws wafv2 untag-resource \
--resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/
apiGatewayWebAcl/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
--tag-keys "KeyName"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Introducción a AWS WAF](#) la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-ip-set

En el siguiente ejemplo de código, se muestra cómo usar update-ip-set.

### AWS CLI

Para modificar la configuración de un conjunto de IP existente

A continuación, se `update-ip-set` actualiza la configuración del conjunto de IP especificado. Esta llamada requiere un identificador, que puede obtener de la llamada `list-ip-sets`, y un token de bloqueo que puede obtener de las llamadas, `list-ip-sets` y `get-ip-set`. Esta llamada también devuelve un token de bloqueo que puedes usar para una actualización posterior.

```
aws wafv2 update-ip-set \  
  --name testip \  
  --scope REGIONAL \  
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --addresses 198.51.100.0/16 \  
  --lock-token 447e55ac-2396-4c6d-b9f9-86b67c17f8b5
```

Salida:

```
{  
  "NextLockToken": "0674c84b-0304-47fe-8728-c6bff46af8fc"  
}
```

Para obtener más información, consulte Conjuntos de [IP y conjuntos de patrones de expresiones regulares](#) en la Guía AWS WAFavanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia [UpdateIpSet](#) de AWS CLI comandos.

## update-regex-pattern-set

En el siguiente ejemplo de código, se muestra cómo usar `update-regex-pattern-set`.

AWS CLI

Para modificar la configuración de un conjunto de patrones de expresiones regulares existente

A continuación, se `update-regex-pattern-set` actualiza la configuración del conjunto de patrones de expresiones regulares especificado. Esta llamada requiere un identificador, que puede obtener de la llamada `list-regex-pattern-sets`, y un token de bloqueo que puede obtener de las llamadas, `list-regex-pattern-sets` y `get-regex-pattern-set`. Esta llamada también devuelve un token de bloqueo que puedes usar para una actualización posterior.

```
aws wafv2 update-regex-pattern-set \  
  --name ExampleRegex \  
  --lock-token 447e55ac-2396-4c6d-b9f9-86b67c17f8b5
```

```
--scope REGIONAL \
--id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
--regular-expression-list RegexpString="^.+ $" \
--lock-token ed207e9c-82e9-4a77-aadd-81e6173ab7eb
```

Salida:

```
{
  "NextLockToken": "12ebc73e-fa68-417d-a9b8-2bdd761a4fa5"
}
```

Para obtener más información, consulte Conjuntos de [IP y conjuntos de patrones de expresiones regulares](#) en la Guía AWS WAFavanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte la Referencia [UpdateRegexpPatternSet](#) de AWS CLI comandos.

## update-rule-group

En el siguiente ejemplo de código, se muestra cómo usar `update-rule-group`.

### AWS CLI

Para actualizar un grupo de reglas personalizado

A continuación, se `update-rule-group` cambia la configuración de visibilidad de un grupo de reglas personalizado existente. Esta llamada requiere un identificador, que puede obtener de la llamada `list-rule-groups`, y un token de bloqueo que puede obtener de las llamadas, `list-rule-groups` y `get-rule-group`. Esta llamada también devuelve un token de bloqueo que puedes usar para una actualización posterior.

```
aws wafv2 update-rule-group \
  --name TestRuleGroup \
  --scope REGIONAL \
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
  --lock-token 7b3bcec2-0000-0000-0000-563bf47249f0 \
  --visibility-
config SampledRequestsEnabled=false,CloudWatchMetricsEnabled=false,MetricName=TestMetricsFor
\
```

```
--region us-west-2
```

Salida:

```
{  
  "NextLockToken": "1eb5ec48-0000-0000-0000-ee9b906c541e"  
}
```

Para obtener más información, consulte [Administrar sus propios grupos de reglas](#) en la AWS WAF Guía avanzada para desarrolladores de AWS Firewall Manager y AWS Shield.

- Para API obtener más información, consulte [UpdateRuleGroup](#) la Referencia de AWS CLI comandos.

## update-web-acl

En el siguiente ejemplo de código, se muestra cómo usar `update-web-acl`.

### AWS CLI

Para actualizar una web ACL

A continuación, se `update-web-acl` cambia la configuración de un sitio web existente ACL. Esta llamada requiere un identificador, que puede obtener de la llamada `list-web-acls`, y un token de bloqueo y otros ajustes, que puede obtener de la llamada `get-web-acl`. Esta llamada también devuelve un token de bloqueo que puedes usar para una actualización posterior.

```
aws wafv2 update-web-acl \  
  --name TestWebAcl \  
  --scope REGIONAL \  
  --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \  
  --lock-token 2294b3a1-0000-0000-0000-a3ae04329de9 \  
  --default-action Block={} \  
  --visibility-  
config SampledRequestsEnabled=false,CloudWatchMetricsEnabled=false,MetricName=NewMetricTestW  
 \  
  --rules file://waf-rule.json \  
  --region us-west-2
```

Salida:



```
{
  "NextLockToken": "714a0cfb-0000-0000-0000-2959c8b9a684"
}
```

Para obtener más información, consulte [Administración y uso de una lista de control de acceso a la web \(WebACL\)](#) en la AWS WAFGuía avanzada para desarrolladores de AWS Firewall Manager and AWS Shield.

- Para API obtener más información, consulte [UpdateWebAcl](#) la Referencia de AWS CLI comandos.

## WorkDocs Ejemplos de Amazon que utilizan AWS CLI

En los siguientes ejemplos de código, se muestra cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con Amazon WorkDocs.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **abort-document-version-upload**

En el siguiente ejemplo de código, se muestra cómo usar `abort-document-version-upload`.

AWS CLI

Para detener la carga de una versión de un documento

En este ejemplo se detiene la carga de una versión de documento iniciada anteriormente.

Comando:

```
aws workdocs abort-document-version-upload --document-  
id feaba64d4efdf271c2521b60a2a44a8f057e84beaabb22f01267313209835f2 --version-  
id 1536773972914-ddb67663e782e7ce8455ebc962217cf9f9e47b5a9a702e5c84dccc417da9313
```

Salida:

```
None
```

- Para API obtener más información, consulte [AbortDocumentVersionUpload](#) la Referencia de AWS CLI comandos.

## activate-user

En el siguiente ejemplo de código, se muestra cómo usar activate-user.

### AWS CLI

Para activar un usuario

En este ejemplo se activa un usuario inactivo.

Comando:

```
aws workdocs activate-user --user-  
id "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
```

Salida:

```
{  
  "User": {  
    "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",  
    "Username": "exampleUser",  
    "EmailAddress": "exampleUser@site.awsapps.com",  
    "GivenName": "Example",  
    "Surname": "User",  
    "OrganizationId": "d-926726012c",  
    "RootFolderId":  
    "75f67c183aa1217409ac87576a45c03a5df5e6d8c51c35c01669970538e86cd0",  
    "RecycleBinFolderId":  
    "642b7dd3e60b14204534f3df7b1959e01b5d170f8c2707f410e40a8149120a57",
```

```

    "Status": "ACTIVE",
    "Type": "MINIMALUSER",
    "CreatedTimestamp": 1521226107.747,
    "ModifiedTimestamp": 1525297406.462,
    "Storage": {
      "StorageUtilizedInBytes": 0,
      "StorageRule": {
        "StorageAllocatedInBytes": 0,
        "StorageType": "QUOTA"
      }
    }
  }
}

```

- Para API obtener más información, consulte [ActivateUser](#) la Referencia de AWS CLI comandos.

## add-resource-permissions

En el siguiente ejemplo de código, se muestra cómo usar `add-resource-permissions`.

### AWS CLI

Para añadir permisos a un recurso

En este ejemplo, se agregan permisos al recurso para las entidades principales especificadas.

Comando:

```

aws workdocs add-resource-permissions --resource-
id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --
principals Id=anonymous, Type=ANONYMOUS, Role=VIEWER

```

Salida:

```

{
  "ShareResults": [
    {
      "PrincipalId": "anonymous",
      "Role": "VIEWER",
      "Status": "SUCCESS",
      "ShareId":
        "d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65",
    }
  ]
}

```

```

        "StatusMessage": ""
    }
]
}

```

- Para API obtener más información, consulte [AddResourcePermissions](#) la Referencia de AWS CLI comandos.

## create-comment

En el siguiente ejemplo de código, se muestra cómo usar `create-comment`.

### AWS CLI

Para añadir un comentario nuevo

En este ejemplo, se agrega un comentario nuevo a la versión del documento especificada.

Comando:

```

aws workdocs create-comment --document-
id 15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-
id 1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920 --
text "This is a comment."

```

Salida:

```

{
  "Comment": {
    "CommentId": "1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5",
    "ThreadId": "1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5",
    "Text": "This is a comment.",
    "Contributor": {
      "Id": "arn:aws:iam::123456789123:user/exampleUser",
      "Username": "exampleUser",
      "GivenName": "Example",
      "Surname": "User",
      "Status": "ACTIVE"
    },
    "CreatedTimestamp": 1534799058.197,
  },
}

```

```
    "Status": "PUBLISHED",  
    "Visibility": "PUBLIC"  
  }  
}
```

- Para API obtener más información, consulte [CreateComment](#) la Referencia de AWS CLI comandos.

## create-custom-metadata

En el siguiente ejemplo de código, se muestra cómo usar `create-custom-metadata`.

### AWS CLI

Para crear metadatos personalizados

En este ejemplo, se crean metadatos personalizados para el documento especificado.

Comando:

```
aws workdocs create-custom-metadata --resource-  
id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --custom-  
metadata KeyName1=example,KeyName2=example2
```

Salida:

```
None
```

- Para API obtener más información, consulte [CreateCustomMetadata](#) la Referencia de AWS CLI comandos.

## create-folder

En el siguiente ejemplo de código, se muestra cómo usar `create-folder`.

### AWS CLI

Para crear una carpeta

En este ejemplo se crea una carpeta.

Comando:

```
aws workdocs create-folder --name documents --parent-folder-id 1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678
```

Salida:

```
{
  "Metadata": {
    "Id": "50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08",
    "Name": "documents",
    "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "ParentFolderId":
"1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
    "CreatedTimestamp": 1534450467.622,
    "ModifiedTimestamp": 1534450467.622,
    "ResourceState": "ACTIVE",
    "Signature": "",
    "Size": 0,
    "LatestVersionSize": 0
  }
}
```

- Para API obtener más información, consulte [CreateFolder](#) la Referencia de AWS CLI comandos.

## create-labels

En el siguiente ejemplo de código, se muestra cómo usar `create-labels`.

### AWS CLI

Para crear etiquetas

En este ejemplo, se crea una serie de etiquetas para un documento.

Comando:

```
aws workdocs create-labels --resource-id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --labels "documents" "examples" "my_documents"
```

Salida:

```
None
```

- Para API obtener más información, consulte [CreateLabels](#) la Referencia de AWS CLI comandos.

## create-notification-subscription

En el siguiente ejemplo de código, se muestra cómo usar `create-notification-subscription`.

AWS CLI

Para crear una suscripción a notificaciones

El siguiente `create-notification-subscription` ejemplo configura una suscripción de notificaciones para la WorkDocs organización de Amazon especificada.

```
aws workdocs create-notification-subscription \  
  --organization-id d-123456789c \  
  --protocol HTTPS \  
  --subscription-type ALL \  
  --notification-endpoint "https://example.com/example"
```

Salida:

```
{  
  "Subscription": {  
    "SubscriptionId": "123ab4c5-678d-901e-f23g-45h6789j0123",  
    "EndPoint": "https://example.com/example",  
    "Protocol": "HTTPS"  
  }  
}
```

Para obtener más información, consulta [Suscríbete a las notificaciones](#) en la Guía para WorkDocs desarrolladores de Amazon.

- Para API obtener más información, consulte [CreateNotificationSubscription](#) la Referencia de AWS CLI comandos.

## create-user

En el siguiente ejemplo de código, se muestra cómo usar `create-user`.

### AWS CLI

Para crear un usuario nuevo

En este ejemplo, se crea un usuario nuevo en un directorio de Simple AD o Microsoft AD.

Comando:

```
aws workdocs create-user --organization-id d-926726012c --username exampleUser2
--email-address exampleUser2@site.awsapps.com --given-name example2Name --
surname example2Surname --password examplePa$$w0rd
```

Salida:

```
{
  "User": {
    "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "Username": "exampleUser2",
    "EmailAddress": "exampleUser2@site.awsapps.com",
    "GivenName": "example2Name",
    "Surname": "example2Surname",
    "OrganizationId": "d-926726012c",
    "RootFolderId":
"35b886cb17198cbd547655e58b025dff0cf34aaed638be52009567e23dc67390",
    "RecycleBinFolderId":
"9858c3e9ed4c2460dde9aadb4c69fde998070dd46e5e985bd08ec6169ea249ff",
    "Status": "ACTIVE",
    "Type": "MINIMALUSER",
    "CreatedTimestamp": 1535478836.584,
    "ModifiedTimestamp": 1535478836.584,
    "Storage": {
      "StorageUtilizedInBytes": 0,
      "StorageRule": {
        "StorageAllocatedInBytes": 0,
        "StorageType": "QUOTA"
      }
    }
  }
}
```



- Para API obtener más información, consulte [CreateUser](#) la Referencia de AWS CLI comandos.

## deactivate-user

En el siguiente ejemplo de código, se muestra cómo usar `deactivate-user`.

### AWS CLI

Para desactivar un usuario

En este ejemplo se desactiva un usuario activo.

Comando:

```
aws workdocs deactivate-user --user-id "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
```

Salida:

```
None
```

- Para API obtener más información, consulte [DeactivateUser](#) la Referencia de AWS CLI comandos.

## delete-comment

En el siguiente ejemplo de código, se muestra cómo usar `delete-comment`.

### AWS CLI

Para eliminar un comentario específico de una versión de documento

En este ejemplo, se elimina el comentario especificado de la versión del documento especificada.

Comando:

```
aws workdocs delete-comment --document-id 15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-id 1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920 --comment-id 1534799058197-c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5
```

Salida:

```
None
```

- Para API obtener más información, consulte [DeleteComment](#) la Referencia de AWS CLI comandos.

## **delete-custom-metadata**

En el siguiente ejemplo de código, se muestra cómo usar `delete-custom-metadata`.

AWS CLI

Para eliminar metadatos personalizados de un recurso

En este ejemplo, se eliminan todos los metadatos personalizados del recurso especificado.

Comando:

```
aws workdocs delete-custom-metadata --resource-  
id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --delete-all
```

Salida:

```
None
```

- Para API obtener más información, consulte [DeleteCustomMetadata](#) la Referencia de AWS CLI comandos.

## **delete-document**

En el siguiente ejemplo de código, se muestra cómo usar `delete-document`.

AWS CLI

Eliminación de un documento

En este ejemplo se elimina el documento especificado.

Comando:

```
aws workdocs delete-document --document-id b83ed5e5b167b65ef69de9d597627ff1a0d4f07a45e67f1fab7d26b54427de0a
```

Salida:

```
None
```

- Para API obtener más información, consulte [DeleteDocument](#) la Referencia de AWS CLI comandos.

## **delete-folder-contents**

En el siguiente ejemplo de código, se muestra cómo usar `delete-folder-contents`.

AWS CLI

Para eliminar el contenido de una carpeta

En este ejemplo, se elimina el contenido de la carpeta especificada.

Comando:

```
aws workdocs delete-folder-contents --folder-id 26fa8aa4ba2071447c194f7b150b07149dbdb9e1c8a301872dcd93a4735ce65d
```

Salida:

```
None
```

- Para API obtener más información, consulte [DeleteFolderContents](#) la Referencia de AWS CLI comandos.

## **delete-folder**

En el siguiente ejemplo de código, se muestra cómo usar `delete-folder`.

AWS CLI

Para eliminar una carpeta

En este ejemplo, se elimina la carpeta especificada.

Comando:

```
aws workdocs delete-folder --folder-id 26fa8aa4ba2071447c194f7b150b07149dbdb9e1c8a301872dcd93a4735ce65d
```

Salida:

```
None
```

- Para API obtener más información, consulte [DeleteFolder](#) la Referencia de AWS CLI comandos.

## **delete-labels**

En el siguiente ejemplo de código, se muestra cómo usar `delete-labels`.

AWS CLI

Para eliminar etiquetas

En este ejemplo, se eliminan las etiquetas especificadas de un documento.

Comando:

```
aws workdocs delete-labels --resource-id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --labels "documents" "examples"
```

Salida:

```
None
```

- Para API obtener más información, consulte [DeleteLabels](#) la Referencia de AWS CLI comandos.

## **delete-notification-subscription**

En el siguiente ejemplo de código, se muestra cómo usar `delete-notification-subscription`.

AWS CLI

Para eliminar una suscripción a notificaciones

En el siguiente delete-notification-subscription ejemplo, se elimina la suscripción de notificaciones especificada.

```
aws workdocs delete-notification-subscription \  
  --subscription-id 123ab4c5-678d-901e-f23g-45h6789j0123 \  
  --organization-id d-123456789c
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Suscríbete a las notificaciones](#) en la Guía para WorkDocs desarrolladores de Amazon.

- Para API obtener más información, consulte [DeleteNotificationSubscription](#) la Referencia de AWS CLI comandos.

## delete-user

En el siguiente ejemplo de código, se muestra cómo usar delete-user.

### AWS CLI

Para eliminar un usuario

En este ejemplo se elimina un usuario.

Comando:

```
aws workdocs delete-user --user-  
id "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
```

Salida:

```
None
```

- Para API obtener más información, consulte [DeleteUser](#) la Referencia de AWS CLI comandos.

## describe-activities

En el siguiente ejemplo de código, se muestra cómo usar describe-activities.

## AWS CLI

Para obtener una lista de las actividades de los usuarios

En este ejemplo, se devuelve una lista de las últimas actividades de los usuarios de la organización especificada, con un límite establecido para las dos últimas actividades.

Comando:

```
aws workdocs describe-activities --organization-id d-926726012c --limit 2
```

Salida:

```
{
  "UserActivities": [
    {
      "Type": "DOCUMENT_VERSION_DOWNLOADED",
      "TimeStamp": 1534800122.17,
      "Initiator": {
        "Id": "arn:aws:iam::123456789123:user/exampleUser"
      },
      "ResourceMetadata": {
        "Type": "document",
        "Name": "updatedDoc",
        "Id":
"15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3",
        "Owner": {
          "Id":
"S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
          "GivenName": "exampleName",
          "Surname": "exampleSurname"
        }
      }
    },
    {
      "Type": "DOCUMENT_VERSION_VIEWED",
      "TimeStamp": 1534799079.207,
      "Initiator": {
        "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
        "GivenName": "exampleName",
        "Surname": "exampleSurname"
      },
      "ResourceMetadata": {
```

```

        "Type": "document",
        "Name": "updatedDoc",
        "Id":
"15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3",
        "Owner": {
            "Id":
"S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
            "GivenName": "exampleName",
            "Surname": "exampleSurname"
        }
    }
},
"Marker":
"DnF1ZXJ5VGhlbkZldGNoAgAAAAAAS7Fm1TaU10d1FTU1h1UU00VVFibD1RWHcAAAAAAAJTRY3bWh5eUgzaVF1ZX
}

```

- Para API obtener más información, consulte [DescribeActivities](#) la Referencia de AWS CLI comandos.

## describe-comments

En el siguiente ejemplo de código, se muestra cómo usar `describe-comments`.

### AWS CLI

Para enumerar todos los comentarios de una versión de documento específica

En este ejemplo se enumeran todos los comentarios de la versión del documento especificada.

Comando:

```

aws workdocs describe-comments --document-
id 15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-
id 1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920

```

Salida:

```

{
  "Comments": [
    {
      "CommentId": "1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5",

```

```

    "ThreadId": "1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5",
    "Text": "This is a comment.",
    "Contributor": {
      "Username": "arn:aws:iam::123456789123:user/exampleUser",
      "Type": "USER"
    },
    "CreatedTimestamp": 1534799058.197,
    "Status": "PUBLISHED",
    "Visibility": "PUBLIC"
  }
]
}

```

- Para API obtener más información, consulte [DescribeComments](#) la Referencia de AWS CLI comandos.

## describe-document-versions

En el siguiente ejemplo de código, se muestra cómo usar `describe-document-versions`.

### AWS CLI

Para recuperar las versiones de un documento

En este ejemplo, se recuperan las versiones del documento especificado, incluidas las versiones inicializadas y una del documento URL de origen.

Comando:

```

aws workdocs describe-document-versions --document-
id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --fields SOURCE

```

Salida:

```

{
  "DocumentVersions": [
    {
      "Id":
"1534452029587-15e129dfc187505c407588df255be83de2920d733859f1d2762411d22a83e3ef",
      "Name": "exampleDoc.docx",

```



```

    "ContentType": "application/vnd.openxmlformats-officedocument.wordprocessingml.document",
    "Size": 13922,
    "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
    "Status": "ACTIVE",
    "CreatedTimestamp": 1534452029.587,
    "ModifiedTimestamp": 1534452029.849,
    "CreatorId":
    "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "Source": {
        "ORIGINAL": "https://gb-us-west-2-prod-doc-source.s3.us-west-2.amazonaws.com/d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65/1534452029587-15e129dfc1875response-content-disposition=attachment%3B%20filename%2A%3DUTF-8%27%27exampleDoc29.docx&X-Amz-Algorithm=AWS1-ABCD-EFG234&X-Amz-Date=20180816T204149Z&X-Amz-SignedHeaders=host&X-Amz-Expires=900&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20180816%2Fus-west-2%2Fs3%2Faws1_request&X-Amz-Signature=01Ab2c34d567e8f90123g456hi78j901k2345678l901234mno56pqr78EXAMPLE"
    }
  },
  {
    "Id": "1529005196082-bb75fa19abc287699cb07147f75816dce43a53a10f28dc001bf61ef2fab01c59",
    "Name": "exampleDoc.pdf",
    "ContentType": "application/pdf",
    "Size": 425916,
    "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
    "Status": "ACTIVE",
    "CreatedTimestamp": 1529005196.082,
    "ModifiedTimestamp": 1529005196.796,
    "CreatorId":
    "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "Source": {
        "ORIGINAL": "https://gb-us-west-2-prod-doc-source.s3.us-west-2.amazonaws.com/d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65/1529005196082-bb75fa19abc287699cb07147f75816dce43a53a10f28dc001bf61ef2fab01c59?response-content-disposition=attachment%3B%20filename%2A%3DUTF-8%27%27exampleDoc29.pdf&X-Amz-Algorithm=AWS1-ABCD-EFG234&X-Amz-Date=20180816T204149Z&X-Amz-SignedHeaders=host&X-Amz-Expires=900&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20180816%2Fus-west-2%2Fs3%2Faws1_request&X-Amz-Signature=01Ab2c34d567e8f90123g456hi78j901k2345678l901234mno56pqr78EXAMPLE"
    }
  }
}

```

```
]
}
```

- Para API obtener más información, consulte la Referencia [DescribeDocumentVersions](#) de AWS CLI comandos.

## describe-folder-contents

En el siguiente ejemplo de código, se muestra cómo usar `describe-folder-contents`.

### AWS CLI

Para describir el contenido de una carpeta

En este ejemplo, se describe todo el contenido activo de la carpeta especificada, incluidos sus documentos y subcarpetas, ordenados por fecha en orden ascendente.

Comando:

```
aws workdocs describe-folder-contents --folder-  
id 1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678 --sort DATE --  
order ASCENDING --type ALL
```

Salida:

```
{  
  "Folders": [  
    {  
      "Id": "50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08",  
      "Name": "testing",  
      "CreatorId":  
"S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",  
      "ParentFolderId":  
"1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",  
      "CreatedTimestamp": 1534450467.622,  
      "ModifiedTimestamp": 1534451113.504,  
      "ResourceState": "ACTIVE",  
      "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",  
      "Size": 23019,  
      "LatestVersionSize": 11537  
    }  
  ],  
}
```

```

"Documents": [
  {
    "Id": "d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65",
    "CreatorId":
    "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "ParentFolderId":
    "1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
    "CreatedTimestamp": 1529005196.082,
    "ModifiedTimestamp": 1534452483.01,
    "LatestVersionMetadata": {
      "Id":
      "1534452029587-15e129dfc187505c407588df255be83de2920d733859f1d2762411d22a83e3ef",
      "Name": "exampleDoc.docx",
      "ContentType": "application/vnd.openxmlformats-
officedocument.wordprocessingml.document",
      "Size": 13922,
      "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
      "Status": "ACTIVE",
      "CreatedTimestamp": 1534452029.587,
      "ModifiedTimestamp": 1534452029.587,
      "CreatorId":
      "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
    },
    "ResourceState": "ACTIVE"
  }
]
}

```

- Para API obtener más información, consulte la Referencia [DescribeFolderContents](#) de AWS CLI comandos.

## describe-groups

En el siguiente ejemplo de código, se muestra cómo usar `describe-groups`.

### AWS CLI

Para recuperar una lista de grupos

En el siguiente `describe-groups` ejemplo, se enumeran los grupos asociados a la WorkDocs organización de Amazon especificada.

```
aws workdocs describe-groups \
```

```
--search-query "e" \  
--organization-id d-123456789c
```

Salida:

```
{  
  "Groups": [  
    {  
      "Id": "S-1-1-11-1122222222-2222233333-3333334444-4444&d-123456789c",  
      "Name": "Example Group 1"  
    },  
    {  
      "Id": "S-1-1-11-1122222222-2222233333-3333334444-5555&d-123456789c",  
      "Name": "Example Group 2"  
    }  
  ]  
}
```

Para obtener más información, consulta [Cómo empezar con Amazon WorkDocs](#) en la Guía de WorkDocs administración de Amazon.

- Para API obtener más información, consulte [DescribeGroups](#) la Referencia de AWS CLI comandos.

## describe-notification-subscriptions

En el siguiente ejemplo de código, se muestra cómo usar `describe-notification-subscriptions`.

AWS CLI

Para recuperar una lista de suscripciones a notificaciones

En el siguiente `describe-notification-subscriptions` ejemplo, se recuperan las suscripciones de notificaciones de la WorkDocs organización de Amazon especificada.

```
aws workdocs describe-notification-subscriptions \  
--organization-id d-123456789c
```

Salida:

```
{
```

```
"Subscriptions": [  
  {  
    "SubscriptionId": "123ab4c5-678d-901e-f23g-45h6789j0123",  
    "EndPoint": "https://example.com/example",  
    "Protocol": "HTTPS"  
  }  
]  
}
```

Para obtener más información, consulta [Suscríbete a las notificaciones](#) en la Guía para WorkDocs desarrolladores de Amazon.

- Para API obtener más información, consulte [DescribeNotificationSubscriptions](#) la Referencia de AWS CLI comandos.

## describe-resource-permissions

En el siguiente ejemplo de código, se muestra cómo usar `describe-resource-permissions`.

### AWS CLI

Para obtener una lista de permisos para un recurso

En este ejemplo, se devuelve una lista de los permisos del recurso (documento o carpeta) especificado.

Comando:

```
aws workdocs describe-resource-permissions --resource-  
id 15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3
```

Salida:

```
{  
  "Principals": [  
    {  
      "Id": "anonymous",  
      "Type": "ANONYMOUS",  
      "Roles": [  
        {  
          "Role": "VIEWER",  
          "Type": "DIRECT"  
        }  
      ]  
    }  
  ]  
}
```

```

    }
  ]
},
{
  "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
  "Type": "USER",
  "Roles": [
    {
      "Role": "OWNER",
      "Type": "DIRECT"
    }
  ]
},
{
  "Id": "d-926726012c",
  "Type": "ORGANIZATION",
  "Roles": [
    {
      "Role": "VIEWER",
      "Type": "INHERITED"
    }
  ]
}
]
}
}

```

- Para API obtener más información, consulte [DescribeResourcePermissions](#) la Referencia de AWS CLI comandos.

## describe-users

En el siguiente ejemplo de código, se muestra cómo usar `describe-users`.

### AWS CLI

Para recuperar detalles de usuarios específicos

En este ejemplo, se recuperan los detalles de todos los usuarios de la organización especificada.

Comando:

```
aws workdocs describe-users --organization-id d-926726012c
```

**Salida:**

```
{
  "Users": [
    {
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
      "Username": "example1User",
      "OrganizationId": "d-926726012c",
      "RootFolderId":
"3c0e3f849dd20a9771d937b9bbcc97e18796150ae56c26d64a4fa0320a2dedc9",
      "RecycleBinFolderId":
"c277f4c4d647be1f5147b3184ffa96e1e2bf708278b696cacba68ba13b91f4fe",
      "Status": "INACTIVE",
      "Type": "USER",
      "CreatedTimestamp": 1535478999.452,
      "ModifiedTimestamp": 1535478999.452
    },
    {
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-4444&d-926726012c",
      "Username": "example2User",
      "EmailAddress": "example2User@site.awsapps.com",
      "GivenName": "example2Name",
      "Surname": "example2Surname",
      "OrganizationId": "d-926726012c",
      "RootFolderId":
"35b886cb17198cbd547655e58b025dff0cf34aaed638be52009567e23dc67390",
      "RecycleBinFolderId":
"9858c3e9ed4c2460dde9aadb4c69fde998070dd46e5e985bd08ec6169ea249ff",
      "Status": "ACTIVE",
      "Type": "MINIMALUSER",
      "CreatedTimestamp": 1535478836.584,
      "ModifiedTimestamp": 1535478836.584
    }
  ]
}
```

- Para API obtener más información, consulte [DescribeUsers](#) la Referencia de AWS CLI comandos.

**get-document-path**

En el siguiente ejemplo de código, se muestra cómo usar `get-document-path`.

## AWS CLI

Para recuperar la información de la ruta de un documento

Este ejemplo recupera la información de la ruta (jerarquía de la carpeta raíz) del documento especificado e incluye los nombres de las carpetas principales.

Comando:

```
aws workdocs get-document-path --document-id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --fields NAME
```

Salida:

```
{
  "Path": {
    "Components": [
      {
        "Id":
        "a43d29cbb8e7c4d25cfee8b803a504b0dc63e760b55ad0c611c6b87691eb6ff3",
        "Name": "/"
      },
      {
        "Id":
        "1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
        "Name": "Top Level Folder"
      },
      {
        "Id":
        "d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65",
        "Name": "exampleDoc.docx"
      }
    ]
  }
}
```

- Para API obtener más información, consulte [GetDocumentPath](#) la Referencia de AWS CLI comandos.

### get-document-version

En el siguiente ejemplo de código, se muestra cómo usar `get-document-version`.



## AWS CLI

Para recuperar los metadatos de la versión de un documento específico

En este ejemplo, se recuperan los metadatos de la versión del documento especificado, incluidos el origen URL y los metadatos personalizados.

Comando:

```
aws workdocs get-document-version --document-
id 15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-
id 1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920 --
fields SOURCE --include-custom-metadata
```

Salida:

```
{
  "Metadata": {
    "Id":
    "1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920",
    "Name": "exampleDoc",
    "ContentType": "application/vnd.openxmlformats-
officedocument.wordprocessingml.document",
    "Size": 11537,
    "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
    "Status": "ACTIVE",
    "CreatedTimestamp": 1521672507.741,
    "ModifiedTimestamp": 1534451113.504,
    "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "Source": {
      "ORIGINAL": "https://gb-us-west-2-prod-doc-source.s3.us-
west-2.amazonaws.com/15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3/152167
response-content-disposition=attachment%3B%20filename%2A
%3DUTF-8%27%27exampleDoc&X-Amz-Algorithm=AWS1-ABCD-EFG234&X-Amz-
Date=20180820T212202Z&X-Amz-SignedHeaders=host&X-Amz-Expires=900&X-Amz-
Credential=AKIAIOSFODNN7EXAMPLE%2F20180820%2Fus-west-2%2Fs3%2Faws1_request&X-Amz-
Signature=01Ab2c34d567e8f90123g456hi78j901k23456781901234mno56pqr78EXAMPLE"
    }
  }
}
```

- Para API obtener más información, consulte [GetDocumentVersion](#) la Referencia de AWS CLI comandos.

## get-document

En el siguiente ejemplo de código, se muestra cómo usar `get-document`.

### AWS CLI

Para recuperar los detalles del documento

En este ejemplo, se recuperan los detalles del documento especificado.

Comando:

```
aws workdocs get-document --document-id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65
```

Salida:

```
{
  "Metadata": {
    "Id": "d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65",
    "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "ParentFolderId":
"1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
    "CreatedTimestamp": 1529005196.082,
    "ModifiedTimestamp": 1534452483.01,
    "LatestVersionMetadata": {
      "Id":
"1534452029587-15e129dfc187505c407588df255be83de2920d733859f1d2762411d22a83e3ef",
      "Name": "exampleDoc.docx",
      "ContentType": "application/vnd.openxmlformats-officedocument.wordprocessingml.document",
      "Size": 13922,
      "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
      "Status": "ACTIVE",
      "CreatedTimestamp": 1534452029.587,
      "ModifiedTimestamp": 1534452029.587,
      "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
    },
    "ResourceState": "ACTIVE"
  }
}
```

- Para API obtener más información, consulte [GetDocument](#) la Referencia de AWS CLI comandos.

## get-folder-path

En el siguiente ejemplo de código, se muestra cómo usar `get-folder-path`.

### AWS CLI

Para recuperar la información de la ruta de una carpeta

Este ejemplo recupera la información de la ruta (jerarquía de la carpeta raíz) de la carpeta especificada e incluye los nombres de las carpetas principales.

Comando:

```
aws workdocs get-folder-path --folder-id 50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08 --fields NAME
```

Salida:

```
{
  "Path": {
    "Components": [
      {
        "Id":
"a43d29cbb8e7c4d25cfee8b803a504b0dc63e760b55ad0c611c6b87691eb6ff3",
        "Name": "/"
      },
      {
        "Id":
"1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
        "Name": "Top Level Folder"
      },
      {
        "Id":
"50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08",
        "Name": "Sublevel Folder"
      }
    ]
  }
}
```

- Para API obtener más información, consulte [GetFolderPath](#) la Referencia de AWS CLI comandos.

## get-folder

En el siguiente ejemplo de código, se muestra cómo usar `get-folder`.

### AWS CLI

Para recuperar los metadatos de una carpeta

En este ejemplo, se recuperan los metadatos de la carpeta especificada.

Comando:

```
aws workdocs get-folder --folder-id 50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08
```

Salida:

```
{
  "Metadata": {
    "Id": "50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08",
    "Name": "exampleFolder",
    "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "ParentFolderId":
"1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
    "CreatedTimestamp": 1534450467.622,
    "ModifiedTimestamp": 1534451113.504,
    "ResourceState": "ACTIVE",
    "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
    "Size": 23019,
    "LatestVersionSize": 11537
  }
}
```

- Para API obtener más información, consulte [GetFolder](#) la Referencia de AWS CLI comandos.

## get-resources

En el siguiente ejemplo de código, se muestra cómo usar `get-resources`.

## AWS CLI

Para recuperar recursos compartidos

El siguiente `get-resources` ejemplo recupera los recursos compartidos con el WorkDocs usuario de Amazon especificado.

```
aws workdocs get-resources \  
  --user-id "S-1-1-11-1111111111-222222222-3333333333-3333" \  
  --collection-type SHARED_WITH_ME
```

Salida:

```
{  
  "Folders": [],  
  "Documents": []  
}
```

Para obtener más información, consulta [Compartir archivos y carpetas](#) en la Guía del WorkDocs usuario de Amazon.

- Para API obtener más información, consulte [GetResources](#) la Referencia de AWS CLI comandos.

## **initiate-document-version-upload**

En el siguiente ejemplo de código, se muestra cómo usar `initiate-document-version-upload`.

## AWS CLI

Para iniciar la carga de una versión de un documento

En el siguiente `initiate-document-upload` ejemplo, se crea un objeto de documento y un objeto de versión nuevos.

```
aws workdocs initiate-document-version-upload \  
  --name exampledocname \  
  --parent-folder-  
id eacd546d952531c633452ed67cac23161aa0d5df2e8061223a59e8f67e7b6189
```

**Salida:**

```
{
  "Metadata": {
    "Id": "feaba64d4efdf271c2521b60a2a44a8f057e84beaabbe22f01267313209835f2",
    "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "ParentFolderId":
"eacd546d952531c633452ed67cac23161aa0d5df2e8061223a59e8f67e7b6189",
    "CreatedTimestamp": 1536773972.914,
    "ModifiedTimestamp": 1536773972.914,
    "LatestVersionMetadata": {
      "Id": "1536773972914-
ddb67663e782e7ce8455ebc962217cf9f9e47b5a9a702e5c84dccccd417da9313",
      "Name": "exampledocname",
      "ContentType": "application/octet-stream",
      "Size": 0,
      "Status": "INITIALIZED",
      "CreatedTimestamp": 1536773972.914,
      "ModifiedTimestamp": 1536773972.914,
      "CreatorId": "arn:aws:iam::123456789123:user/EXAMPLE"
    },
    "ResourceState": "ACTIVE"
  },
  "UploadMetadata": {
    "UploadUrl": "https://gb-us-west-2-prod-doc-source.s3.us-
west-2.amazonaws.com/
feaba64d4efdf271c2521b60a2a44a8f057e84beaabbe22f01267313209835f2/1536773972914-
ddb67663e782e7ce8455ebc962217cf9f9e47b5a9a702e5c84dccccd417da9313?X-Amz-
Algorithm=AWS1-ABCD-EFG234&X-Amz-Date=20180912T173932Z&X-Amz-SignedHeaders=content-
type%3Bhost%3Bx-amz-server-side-encryption&X-Amz-Expires=899&X-Amz-
Credential=AKIAIOSFODNN7EXAMPLE%2F20180912%2Fus-west-2%2Fs3%2Faws1_request&X-Amz-
Signature=01Ab2c34d567e8f90123g456hi78j901k23456781901234mno56pqr78EXAMPLE",
    "SignedHeaders": {
      "Content-Type": "application/octet-stream",
      "x-amz-server-side-encryption": "ABC123"
    }
  }
}
```

- Para API obtener más información, consulte [InitiateDocumentVersionUpload](#) la Referencia de AWS CLI comandos.

## **remove-all-resource-permissions**

En el siguiente ejemplo de código, se muestra cómo usar `remove-all-resource-permissions`.

### AWS CLI

Para eliminar todos los permisos de un recurso específico

En este ejemplo, se quitan todos los permisos del recurso especificado.

Comando:

```
aws workdocs remove-all-resource-permissions --resource-id 1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678
```

Salida:

```
None
```

- Para API obtener más información, consulte [RemoveAllResourcePermissions](#) la Referencia de AWS CLI comandos.

## **remove-resource-permission**

En el siguiente ejemplo de código, se muestra cómo usar `remove-resource-permission`.

### AWS CLI

Para eliminar los permisos de un recurso

En este ejemplo, se quitan los permisos del recurso para el principal especificado.

Comando:

```
aws workdocs remove-resource-permission --resource-id 1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678 --principal-id anonymous
```

Salida:

```
None
```

- Para API obtener más información, consulte [RemoveResourcePermission](#) la Referencia de AWS CLI comandos.

## update-document-version

En el siguiente ejemplo de código, se muestra cómo usar `update-document-version`.

### AWS CLI

Para cambiar el estado de la versión de un documento a Activo

En este ejemplo, se cambia el estado de la versión del documento a Activa.

Comando:

```
aws workdocs update-document-version --document-  
id 15df51e0335fcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-  
id 1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920 --  
version-status ACTIVE
```

Salida:

```
None
```

- Para API obtener más información, consulte [UpdateDocumentVersion](#) la Referencia de AWS CLI comandos.

## update-document

En el siguiente ejemplo de código, se muestra cómo usar `update-document`.

### AWS CLI

Para actualizar un documento

En este ejemplo, se actualizan el nombre y la carpeta principal de un documento.

Comando:

```
aws workdocs update-document --document-  
id 15df51e0335fcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --
```



```
name updatedDoc --parent-folder-  
id 50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08
```

Salida:

```
None
```

- Para API obtener más información, consulte [UpdateDocument](#) la Referencia de AWS CLI comandos.

## update-folder

En el siguiente ejemplo de código, se muestra cómo usar `update-folder`.

AWS CLI

Para actualizar una carpeta

En este ejemplo, se actualizan el nombre y la carpeta principal de una carpeta.

Comando:

```
aws workdocs update-folder --folder-  
id 50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08 --  
name exampleFolder1 --parent-folder-  
id 1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678
```

Salida:

```
None
```

- Para API obtener más información, consulte [UpdateFolder](#) la Referencia de AWS CLI comandos.

## update-user

En el siguiente ejemplo de código, se muestra cómo usar `update-user`.

AWS CLI

Para actualizar un usuario

En este ejemplo, se actualiza la zona horaria del usuario especificado.

Comando:

```
aws workdocs update-user --user-id "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c" --time-zone-id "America/Los_Angeles"
```

Salida:

```
{
  "User": {
    "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
    "Username": "exampleUser",
    "EmailAddress": "exampleUser@site.awsapps.com",
    "GivenName": "Example",
    "Surname": "User",
    "OrganizationId": "d-926726012c",
    "RootFolderId":
    "c5eceb5e1a2d1d460c9d1af8330ae117fc8d39bb1d3ed6acd0992d5ff192d986",
    "RecycleBinFolderId":
    "6ca20102926ad15f04b1d248d6d6e44f2449944eda5c758f9a1e9df6a6b7fa66",
    "Status": "ACTIVE",
    "Type": "USER",
    "TimeZoneId": "America/Los_Angeles",
    "Storage": {
      "StorageUtilizedInBytes": 0,
      "StorageRule": {
        "StorageAllocatedInBytes": 53687091200,
        "StorageType": "QUOTA"
      }
    }
  }
}
```

- Para API obtener más información, consulte [UpdateUser](#) la Referencia de AWS CLI comandos.

## WorkMail Ejemplos de Amazon que utilizan AWS CLI

En los siguientes ejemplos de código, se muestra cómo realizar acciones e implementar situaciones comunes AWS Command Line Interface con Amazon WorkMail.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **associate-delegate-to-resource**

En el siguiente ejemplo de código, se muestra cómo usar `associate-delegate-to-resource`.

AWS CLI

Para añadir un delegado a un recurso

El siguiente `associate-delegate-to-resource` comando agrega un delegado a un recurso.

```
aws workmail associate-delegate-to-resource \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --resource-id r-68bf2d3b1c0244aab7264c24b9217443 \  
  --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [AssociateDelegateToResource](#) la Referencia de AWS CLI comandos.

### **associate-member-to-group**

En el siguiente ejemplo de código, se muestra cómo usar `associate-member-to-group`.

AWS CLI

Para añadir un miembro a un grupo

El siguiente `associate-member-to-group` comando agrega el miembro especificado a un grupo.

```
aws workmail associate-member-to-group \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --group-id S-1-1-11-1122222222-2222233333-3333334444-4444 \  
  --member-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [AssociateMemberToGroup](#) la Referencia de AWS CLI comandos.

## create-alias

En el siguiente ejemplo de código, se muestra cómo usar `create-alias`.

### AWS CLI

Para crear un alias

El siguiente `create-alias` comando crea un alias para la entidad especificada (usuario o grupo).

```
aws workmail create-alias \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1122222222-2222233333-3333334444-4444 \  
  --alias exampleAlias@site.awsapps.com
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [CreateAlias](#) la Referencia de AWS CLI comandos.

## create-group

En el siguiente ejemplo de código, se muestra cómo usar `create-group`.

### AWS CLI

Para crear un grupo nuevo

El siguiente `create-group` comando crea un grupo nuevo para la organización especificada.

```
aws workmail create-group \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --group-name ExampleGroup \  
  --group-type Standard
```

```
--organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
--name exampleGroup1
```

Salida:

```
{  
  "GroupId": "S-1-1-11-1122222222-2222233333-3333334444-4444"  
}
```

- Para API obtener más información, consulte [CreateGroup](#) la Referencia de AWS CLI comandos.

## create-resource

En el siguiente ejemplo de código, se muestra cómo usar `create-resource`.

### AWS CLI

Para crear un recurso nuevo

El siguiente `create-resource` comando crea un nuevo recurso (sala de reuniones) para la organización especificada.

```
aws workmail create-resource \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --name exampleRoom1 \  
  --type ROOM
```

Salida:

```
{  
  "ResourceId": "r-7afe0efbade843a58cdc10251fce992c"  
}
```

- Para API obtener más información, consulte [CreateResource](#) la Referencia de AWS CLI comandos.

## create-user

En el siguiente ejemplo de código, se muestra cómo usar `create-user`.

## AWS CLI

Para crear un usuario nuevo

El siguiente `create-user` comando crea un usuario nuevo.

```
aws workmail create-user \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --name exampleName \  
  --display-name exampleDisplayName \  
  --password examplePa$$w0rd
```

Salida:

```
{  
  "UserId": "S-1-1-11-1111111111-2222222222-3333333333-3333"  
}
```

- Para API obtener más información, consulte [CreateUser](#) la Referencia de AWS CLI comandos.

## `delete-access-control-rule`

En el siguiente ejemplo de código, se muestra cómo usar `delete-access-control-rule`.

## AWS CLI

Para eliminar una regla de control de acceso

En el siguiente `delete-access-control-rule` ejemplo, se elimina la regla de control de acceso especificada de la WorkMail organización de Amazon especificada.

```
aws workmail delete-access-control-rule \  
  --organization-id m-n1pq2345678r901st2u3vx45x6789yza \  
  --name "myRule"
```

Este comando no genera ninguna salida.

Para obtener más información, consulta Cómo [trabajar con reglas de control de acceso](#) en la Guía del WorkMail administrador de Amazon.

- Para API obtener más información, consulte [DeleteAccessControlRule](#) la Referencia de AWS CLI comandos.

## delete-alias

En el siguiente ejemplo de código, se muestra cómo usar `delete-alias`.

### AWS CLI

Para eliminar un alias

El siguiente `delete-alias` comando elimina el alias de la entidad especificada (usuario o grupo).

```
aws workmail delete-alias \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1122222222-2222233333-3333334444-4444 \  
  --alias exampleAlias@site.awsapps.com
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteAlias](#) la Referencia de AWS CLI comandos.

## delete-group

En el siguiente ejemplo de código, se muestra cómo usar `delete-group`.

### AWS CLI

Para eliminar un grupo existente

El siguiente `delete-group` comando elimina un grupo existente de Amazon WorkMail.

```
aws workmail delete-group \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --group-id S-1-1-11-1122222222-2222233333-3333334444-4444
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteGroup](#) la Referencia de AWS CLI comandos.

## delete-mailbox-permissions

En el siguiente ejemplo de código, se muestra cómo usar `delete-mailbox-permissions`.

## AWS CLI

Para eliminar los permisos de los buzones

El siguiente `delete-mailbox-permissions` comando elimina los permisos de buzón que se concedieron anteriormente a un usuario o grupo. La entidad representa al usuario propietario del buzón y el cesionario representa al usuario o grupo para el que se van a eliminar los permisos.

```
aws workmail delete-mailbox-permissions \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1122222222-2222233333-3333334444-4444 \  
  --grantee-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte la Referencia [DeleteMailboxPermissions](#) de AWS CLI comandos.

## `delete-resource`

En el siguiente ejemplo de código, se muestra cómo usar `delete-resource`.

### AWS CLI

Para eliminar un recurso existente

El siguiente `delete-resource` comando elimina un recurso existente de Amazon WorkMail.

```
aws workmail delete-resource \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --resource-id r-7afe0efbade843a58cdc10251fce992c
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteResource](#) la Referencia de AWS CLI comandos.

## `delete-user`

En el siguiente ejemplo de código, se muestra cómo usar `delete-user`.



## AWS CLI

Para eliminar un usuario

El siguiente `delete-user` comando elimina el usuario especificado de Amazon WorkMail y de todos los sistemas posteriores.

```
aws workmail delete-user \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --user-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeleteUser](#) la Referencia de AWS CLI comandos.

## `deregister-from-work-mail`

En el siguiente ejemplo de código, se muestra cómo usar `deregister-from-work-mail`.

## AWS CLI

Para deshabilitar una entidad existente

El siguiente `deregister-from-work-mail` comando inhabilita el uso de Amazon WorkMail por parte de una entidad existente (usuario, grupo o recurso).

```
aws workmail deregister-from-work-mail \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DeregisterFromWorkMail](#) la Referencia de AWS CLI comandos.

## `describe-group`

En el siguiente ejemplo de código, se muestra cómo usar `describe-group`.

## AWS CLI

Para recuperar información de un grupo

El siguiente `describe-group` comando recupera información sobre el grupo especificado.

```
aws workmail describe-group \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --group-id S-1-1-11-1122222222-2222233333-3333334444-4444
```

Salida:

```
{  
  "GroupId": "S-1-1-11-1122222222-2222233333-3333334444-4444",  
  "Name": "exampleGroup1",  
  "State": "ENABLED"  
}
```

- Para API obtener más información, consulte [DescribeGroup](#) la Referencia de AWS CLI comandos.

## **describe-organization**

En el siguiente ejemplo de código, se muestra cómo usar `describe-organization`.

AWS CLI

Para recuperar información de una organización

El siguiente `describe-organization` comando recupera la información de la WorkMail organización de Amazon especificada.

```
aws workmail describe-organization \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

Salida:

```
{  
  "OrganizationId": "m-d281d0a2fd824be5b6cd3d3ce909fd27",  
  "Alias": "alias",  
  "State": "Active",  
  "DirectoryId": "d-926726012c",  
  "DirectoryType": "VpcDirectory",  
  "DefaultMailDomain": "site.awsapps.com",  
}
```

```
"CompletedDate": 1522693605.468,  
"ARN": "arn:aws:workmail:us-west-2:11112223333:organization/m-  
n1pq2345678r901st2u3vx45x6789yza"  
}
```

Para obtener más información, consulte [Working with Organizations](#) en la Guía del WorkMail administrador de Amazon.

- Para API obtener más información, consulte [DescribeOrganization](#) la Referencia de AWS CLI comandos.

## describe-resource

En el siguiente ejemplo de código, se muestra cómo usar `describe-resource`.

### AWS CLI

Para recuperar información de un recurso

El siguiente `describe-resource` comando recupera información sobre el recurso especificado.

```
aws workmail describe-resource \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --resource-id r-7afe0efbade843a58cdc10251fce992c
```

Salida:

```
{  
  "ResourceId": "r-7afe0efbade843a58cdc10251fce992c",  
  "Name": "exampleRoom1",  
  "Type": "ROOM",  
  "BookingOptions": {  
    "AutoAcceptRequests": true,  
    "AutoDeclineRecurringRequests": false,  
    "AutoDeclineConflictingRequests": true  
  },  
  "State": "ENABLED"  
}
```

- Para API obtener más información, consulte [DescribeResource](#) la Referencia de AWS CLI comandos.

## describe-user

En el siguiente ejemplo de código, se muestra cómo usar `describe-user`.

### AWS CLI

Para recuperar la información del usuario

El siguiente `describe-user` comando recupera información sobre el usuario especificado.

```
aws workmail describe-user \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --user-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Salida:

```
{  
  "UserId": "S-1-1-11-1111111111-2222222222-3333333333-3333",  
  "Name": "exampleUser1",  
  "Email": "exampleUser1@site.awsapps.com",  
  "DisplayName": "",  
  "State": "ENABLED",  
  "UserRole": "USER",  
  "EnabledDate": 1532459261.827  
}
```

- Para API obtener más información, consulte [DescribeUser](#) la Referencia de AWS CLI comandos.

## disassociate-delegate-from-resource

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-delegate-from-resource`.

### AWS CLI

Para eliminar un miembro de un recurso

El siguiente `disassociate-delegate-from-resource` comando elimina el miembro especificado de un recurso.

```
aws workmail disassociate-delegate-from-resource \  
  --resource-id exampleResourceId \  
  --member-id exampleMemberId
```

```
--organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
--resource-id r-68bf2d3b1c0244aab7264c24b9217443 \  
--entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DisassociateDelegateFromResource](#) la Referencia de AWS CLI comandos.

## **disassociate-member-from-group**

En el siguiente ejemplo de código, se muestra cómo usar `disassociate-member-from-group`.

### AWS CLI

Para eliminar un miembro de un grupo

El siguiente `disassociate-member-from-group` comando elimina el miembro especificado de un grupo.

```
aws workmail disassociate-member-from-group \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --group-id S-1-1-11-1122222222-2222233333-3333334444-4444 \  
  --member-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [DisassociateMemberFromGroup](#) la Referencia de AWS CLI comandos.

## **get-access-control-effect**

En el siguiente ejemplo de código, se muestra cómo usar `get-access-control-effect`.

### AWS CLI

Para obtener el efecto de las reglas de control de acceso

El siguiente `get-access-control-effect` ejemplo recupera el efecto de las reglas de control de acceso de la WorkMail organización de Amazon especificada para la dirección IP, la acción del protocolo de acceso y el ID de usuario especificados.

```
aws workmail get-access-control-effect \  
  --organization-id m-n1pq2345678r901st2u3vx45x6789yza \  
  --ip-address "192.0.2.0" \  
  --action "WindowsOutlook" \  
  --user-id "S-1-1-11-1111111111-2222222222-3333333333-3333"
```

Salida:

```
{  
  "Effect": "DENY",  
  "MatchedRules": [  
    "myRule"  
  ]  
}
```

Para obtener más información, consulta Cómo [trabajar con reglas de control de acceso](#) en la Guía del WorkMail administrador de Amazon.

- Para API obtener más información, consulte [GetAccessControlEffect](#) la Referencia de AWS CLI comandos.

## get-mailbox-details

En el siguiente ejemplo de código, se muestra cómo usar `get-mailbox-details`.

AWS CLI

Para obtener los detalles del buzón de correo de un usuario

El siguiente `get-mailbox-details` comando recupera detalles sobre el buzón del usuario especificado.

```
aws workmail get-mailbox-details \  
  --organization-id m-n1pq2345678r901st2u3vx45x6789yza \  
  --user-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Salida:

```
{  
  "MailboxQuota": 51200,  
}
```

```
"MailboxSize": 0.03890800476074219
}
```

Para obtener más información, consulte [Gestión de cuentas de usuario](#) en la Guía del WorkMail administrador de Amazon.

- Para API obtener más información, consulte [GetMailboxDetails](#) la Referencia de AWS CLI comandos.

## list-access-control-rules

En el siguiente ejemplo de código, se muestra cómo usar `list-access-control-rules`.

### AWS CLI

Para enumerar las reglas de control de acceso

En el siguiente `list-access-control-rules` ejemplo, se enumeran las reglas de control de acceso de la WorkMail organización de Amazon especificada.

```
aws workmail list-access-control-rules \
  --organization-id m-n1pq2345678r901st2u3vx45x6789yza
```

Salida:

```
{
  "Rules": [
    {
      "Name": "default",
      "Effect": "ALLOW",
      "Description": "Default WorkMail Rule",
      "DateCreated": 0.0,
      "DateModified": 0.0
    },
    {
      "Name": "myRule",
      "Effect": "DENY",
      "Description": "my rule",
      "UserIds": [
        "S-1-1-11-1111111111-2222222222-3333333333-3333"
      ],
      "DateCreated": 1581635628.0,
    }
  ]
}
```

```

    "DateModified": 1581635628.0
  }
]
}

```

Para obtener más información, consulta Cómo [trabajar con reglas de control de acceso](#) en la Guía del WorkMail administrador de Amazon.

- Para API obtener más información, consulte [ListAccessControlRules](#) la Referencia de AWS CLI comandos.

## list-aliases

En el siguiente ejemplo de código, se muestra cómo usar `list-aliases`.

### AWS CLI

Para enumerar los alias de un miembro

El siguiente `list-aliases` comando muestra los alias del miembro especificado (usuario o grupo).

```

aws workmail list-aliases \
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
  --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333

```

Salida:

```

{
  "Aliases": [
    "exampleAlias@site.awsapps.com",
    "exampleAlias1@site.awsapps.com"
  ]
}

```

- Para API obtener más información, consulte [ListAliases](#) la Referencia de AWS CLI comandos.

## list-group-members

En el siguiente ejemplo de código, se muestra cómo usar `list-group-members`.



## AWS CLI

Para ver una lista de los miembros del grupo

El siguiente `list-group-members` comando muestra una lista de los miembros del grupo especificado.

```
aws workmail list-group-members \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --group-id S-1-1-11-1122222222-2222233333-3333334444-4444
```

Salida:

```
{  
  "Members": [  
    {  
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333",  
      "Name": "exampleUser1",  
      "Type": "USER",  
      "State": "ENABLED",  
      "EnabledDate": 1532459261.827  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListGroupMembers](#) la Referencia de AWS CLI comandos.

## list-groups

En el siguiente ejemplo de código, se muestra cómo usar `list-groups`.

## AWS CLI

Para recuperar una lista de grupos

El siguiente `list-groups` comando recupera los resúmenes de los grupos de la organización especificada.

```
aws workmail list-groups \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

```
--organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

Salida:

```
{
  "Groups": [
    {
      "Id": "S-1-1-11-1122222222-2222233333-3333334444-4444",
      "Name": "exampleGroup1",
      "State": "DISABLED"
    },
    {
      "Id": "S-4-4-44-1122222222-2222233333-3333334444-4444",
      "Name": "exampleGroup2",
      "State": "ENABLED"
    }
  ]
}
```

- Para API obtener más información, consulte la Referencia [ListGroups](#) de AWS CLI comandos.

## list-mailbox-permissions

En el siguiente ejemplo de código, se muestra cómo usar `list-mailbox-permissions`.

AWS CLI

Para recuperar los permisos del buzón

El siguiente `list-mailbox-permissions` comando recupera los permisos de buzón asociados al buzón de la entidad especificada.

```
aws workmail list-mailbox-permissions \
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
  --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Salida:

```
{
  "Permissions": [
```

```
{
  "GranteeId": "S-1-1-11-1122222222-2222233333-3333334444-4444",
  "GranteeType": "USER",
  "PermissionValues": [
    "FULL_ACCESS"
  ]
}
```

- Para API obtener más información, consulte [ListMailboxPermissions](#) la Referencia de AWS CLI comandos.

## list-organizations

En el siguiente ejemplo de código, se muestra cómo usar `list-organizations`.

### AWS CLI

Para recuperar una lista de organizaciones

El siguiente `list-organizations` comando recupera los resúmenes de las organizaciones no eliminadas.

```
aws workmail list-organizations
```

Salida:

```
{
  "OrganizationSummaries": [
    {
      "OrganizationId": "m-d281d0a2fd824be5b6cd3d3ce909fd27",
      "Alias": "exampleAlias",
      "State": "Active"
    }
  ]
}
```

- Para API obtener más información, consulte la Referencia de [ListOrganizations](#) comandos AWS CLI .

## list-resource-delegates

En el siguiente ejemplo de código, se muestra cómo usar `list-resource-delegates`.

### AWS CLI

Para enumerar los delegados de un recurso

El siguiente `list-resource-delegates` comando recupera los delegados asociados al recurso especificado.

```
aws workmail list-resource-delegates \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --resource-id r-68bf2d3b1c0244aab7264c24b9217443
```

Salida:

```
{  
  "Delegates": [  
    {  
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333",  
      "Type": "USER"  
    }  
  ]  
}
```

- Para API obtener más información, consulte [ListResourceDelegates](#) la Referencia de AWS CLI comandos.

## list-resources

En el siguiente ejemplo de código, se muestra cómo usar `list-resources`.

### AWS CLI

Para recuperar una lista de recursos

El siguiente `list-resources` comando recupera los resúmenes de los recursos de la organización especificada.

```
aws workmail list-resources \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

```
--organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

Salida:

```
{
  "Resources": [
    {
      "Id": "r-7afe0efbade843a58cdc10251fce992c",
      "Name": "exampleRoom1",
      "Type": "ROOM",
      "State": "ENABLED"
    }
  ]
}
```

- Para API obtener más información, consulte la Referencia [ListResources](#) de AWS CLI comandos.

## list-tags-for-resource

En el siguiente ejemplo de código, se muestra cómo usar `list-tags-for-resource`.

AWS CLI

Para enumerar las etiquetas de un recurso

En el siguiente `list-tags-for-resource` ejemplo, se muestran las etiquetas de la WorkMail organización de Amazon especificada.

```
aws workmail list-tags-for-resource \
  --resource-arn arn:aws:workmail:us-west-2:111122223333:organization/m-
  n1pq2345678r901st2u3vx45x6789yza
```

Salida:

```
{
  "Tags": [
    {
      "Key": "priority",
      "Value": "1"
    }
  ]
}
```

```
}
```

Para obtener más información, consulta Cómo [etiquetar una organización](#) en la Guía del WorkMail administrador de Amazon.

- Para API obtener más información, consulte [ListTagsForResource](#) la Referencia de AWS CLI comandos.

## list-users

En el siguiente ejemplo de código, se muestra cómo usar `list-users`.

### AWS CLI

Para recuperar una lista de usuarios

El siguiente `list-users` comando recupera los resúmenes de los usuarios de la organización especificada.

```
aws workmail list-users \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

Salida:

```
{  
  "Users": [  
    {  
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333",  
      "Email": "exampleUser1@site.awsapps.com",  
      "Name": "exampleUser1",  
      "State": "ENABLED",  
      "UserRole": "USER",  
      "EnabledDate": 1532459261.827  
    },  
    {  
      "Id": "S-1-1-11-1122222222-2222233333-3333334444-4444",  
      "Name": "exampleGuestUser",  
      "State": "DISABLED",  
      "UserRole": "SYSTEM_USER"  
    }  
  ]  
}
```



```
--entity-id S-1-1-11-1111111111-2222222222-3333333333-3333 \  
--grantee-id S-1-1-11-1122222222-2222233333-3333334444-4444 \  
--permission-values FULL_ACCESS
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [PutMailboxPermissions](#) la Referencia de AWS CLI comandos.

## register-to-work-mail

En el siguiente ejemplo de código, se muestra cómo usar `register-to-work-mail`.

### AWS CLI

Para registrar una entidad existente o deshabilitada

El siguiente `register-to-work-mail` comando permite que la entidad existente especificada (usuario, grupo o recurso) utilice Amazon WorkMail.

```
aws workmail register-to-work-mail \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1122222222-2222233333-3333334444-4444 \  
  --email exampleGroup1@site.awsapps.com
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [RegisterToWorkMail](#) la Referencia de AWS CLI comandos.

## reset-password

En el siguiente ejemplo de código, se muestra cómo usar `reset-password`.

### AWS CLI

Para restablecer la contraseña de un usuario

El siguiente `reset-password` comando restablece la contraseña del usuario especificado.

```
aws workmail reset-password \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --user-name exampleGroup1@site.awsapps.com
```



```
--user-id S-1-1-11-1111111111-2222222222-3333333333-3333 \  
--password examplePa$$w0rd
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [ResetPassword](#) la Referencia de AWS CLI comandos.

## tag-resource

En el siguiente ejemplo de código, se muestra cómo usar `tag-resource`.

### AWS CLI

Para aplicar una etiqueta a un recurso

En el siguiente `tag-resource` ejemplo, se aplica una etiqueta con la clave «priority» y el valor «1» a la WorkMail organización de Amazon especificada.

```
aws workmail tag-resource \  
  --resource-arn arn:aws:workmail:us-west-2:111122223333:organization/m-  
n1pq2345678r901st2u3vx45x6789yza \  
  --tags "Key=priority, Value=1"
```

Este comando no genera ninguna salida.

Para obtener más información, consulta Cómo [etiquetar una organización](#) en la Guía del WorkMail administrador de Amazon.

- Para API obtener más información, consulte [TagResource](#) la Referencia de AWS CLI comandos.

## untag-resource

En el siguiente ejemplo de código, se muestra cómo usar `untag-resource`.

### AWS CLI

Para quitar la etiqueta de un recurso

En el siguiente `untag-resource` ejemplo, se elimina la etiqueta especificada de la WorkMail organización de Amazon especificada.

```
aws workmail untag-resource \  
  --resource-arn arn:aws:workmail:us-west-2:111122223333:organization/m-  
n1pq2345678r901st2u3vx45x6789yza \  
  --tag-keys "priority"
```

Este comando no genera ninguna salida.

Para obtener más información, consulta Cómo [etiquetar una organización](#) en la Guía del WorkMail administrador de Amazon.

- Para API obtener más información, consulte [UntagResource](#) la Referencia de AWS CLI comandos.

## update-mailbox-quota

En el siguiente ejemplo de código, se muestra cómo usar update-mailbox-quota.

### AWS CLI

Para actualizar la cuota de buzones de correo de un usuario

El siguiente update-mailbox-quota comando cambia la cuota de buzón del usuario especificado.

```
aws workmail update-mailbox-quota \  
  --organization-id m-n1pq2345678r901st2u3vx45x6789yza \  
  --user-id S-1-1-11-1111111111-2222222222-3333333333-3333 \  
  --mailbox-quota 40000
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Gestión de cuentas de usuario](#) en la Guía del WorkMail administrador de Amazon.

- Para API obtener más información, consulte [UpdateMailboxQuota](#) la Referencia de AWS CLI comandos.

## update-primary-email-address

En el siguiente ejemplo de código, se muestra cómo usar update-primary-email-address.

## AWS CLI

Para actualizar una dirección de correo electrónico principal

El siguiente `update-primary-email-address` comando actualiza la dirección de correo electrónico principal de la entidad especificada (usuario, grupo o recurso).

```
aws workmail update-primary-email-address \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333 \  
  --email exampleUser2@site.awsapps.com
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UpdatePrimaryEmailAddress](#) la Referencia de AWS CLI comandos.

## update-resource

En el siguiente ejemplo de código, se muestra cómo usar `update-resource`.

## AWS CLI

Para actualizar un recurso

El siguiente `update-resource` comando actualiza el nombre del recurso especificado.

```
aws workmail update-resource \  
  --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \  
  --resource-id r-7afe0efbade843a58cdc10251fce992c \  
  --name exampleRoom2
```

Este comando no genera ninguna salida.

- Para API obtener más información, consulte [UpdateResource](#) la Referencia de AWS CLI comandos.

## Ejemplos de Amazon WorkMail Message Flow utilizando AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar situaciones comunes mediante Amazon WorkMail Message Flow. AWS Command Line Interface

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

## Acciones

### **get-raw-message-content**

En el siguiente ejemplo de código, se muestra cómo usar `get-raw-message-content`.

AWS CLI

Para obtener el contenido sin procesar de un mensaje de correo electrónico

El siguiente `get-raw-message-content` ejemplo obtiene el contenido sin procesar de un mensaje de correo electrónico en tránsito y lo envía a un archivo de texto denominado `test`.

```
aws workmailmessageflow get-raw-message-content \  
  --message-id a1b2cd34-ef5g-6h7j-k18m-npq9012345rs \  
  test
```

Contenido del archivo `test` después de ejecutar el comando:

```
Subject: Hello World  
From: =?UTF-8?Q?marymajor_marymajor?= <marymajor@example.com>  
To: =?UTF-8?Q?mateojackson=40example=2Enet?= <mateojackson@example.net>  
Date: Thu, 7 Nov 2019 19:22:46 +0000  
Mime-Version: 1.0  
Content-Type: multipart/alternative;  
  boundary="=_EXAMPLE+"  
References: <mail.1ab23c45.5de6.7f890g123hj45678@storage.wm.amazon.com>  
X-Priority: 3 (Normal)  
X-Mailer: Amazon WorkMail  
Thread-Index: EXAMPLE  
Thread-Topic: Hello World  
Message-Id: <mail.1ab23c45.5de6.7f890g123hj45678@storage.wm.amazon.com>
```

```
This is a multi-part message in MIME format. Your mail reader does not
understand MIME message format.
--=_EXAMPLE+
Content-Type: text/plain; charset=UTF-8
Content-Transfer-Encoding: 7bit

hello world

--=_EXAMPLE+
Content-Type: text/html; charset=utf-8
Content-Transfer-Encoding: quoted-printable

<!DOCTYPE HTML><html>
<head>
<meta name=3D"Generator" content=3D"Amazon WorkMail v3.0-4510">
<meta http-equiv=3D"Content-Type" content=3D"text/html; charset=3Dutf-8">=

<title>testing</title>
</head>
<body>
<p style=3D"margin: 0px; font-family: Arial, Tahoma, Helvetica, sans-seri=
f; font-size: small;">hello world</p>
</body>
</html>
--=_EXAMPLE+--
```

Para obtener más información, consulte [Recuperación del contenido de los mensajes con AWS Lambda](#) en la Guía del administrador de WorkMail Amazon.

- Para API obtener más información, consulte la Referencia [GetRawMessageContent](#) de AWS CLI comandos.

## WorkSpaces ejemplos que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del AWS Command Line Interface with WorkSpaces.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### **create-tags**

En el siguiente ejemplo de código, se muestra cómo usar `create-tags`.

#### AWS CLI

Para añadir etiquetas a un `WorkSpace`

En el siguiente `create-tags` ejemplo, se agregan las etiquetas especificadas a las especificadas `WorkSpace`.

```
aws workspaces create-tags \  
  --resource-id ws-dk1x zr417 \  
  --tags Key=Department, Value=Finance
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [WorkSpaces los recursos sobre etiquetas](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [CreateTags](#) la Referencia de AWS CLI comandos.

### **create-workspaces**

En el siguiente ejemplo de código, se muestra cómo usar `create-workspaces`.

#### AWS CLI

Ejemplo 1: Para crear un `AlwaysOn WorkSpace`

En el siguiente `create-workspaces` ejemplo, se crea una `AlwaysOn WorkSpace` para el usuario especificado, utilizando el directorio y el paquete especificados.

```
aws workspaces create-workspaces \  
  --workspaces DirectoryId=d-926722edaf,UserName=Mateo,BundleId=wsb-0zsvgp8fc
```

Salida:

```
{  
  "FailedRequests": [],  
  "PendingRequests": [  
    {  
      "WorkspaceId": "ws-kcqms853t",  
      "DirectoryId": "d-926722edaf",  
      "UserName": "Mateo",  
      "State": "PENDING",  
      "BundleId": "wsb-0zsvgp8fc"  
    }  
  ]  
}
```

Ejemplo 2: Para crear un AutoStop WorkSpace

En el siguiente create-workspaces ejemplo, se crea una AutoStop WorkSpace para el usuario especificado, utilizando el directorio y el paquete especificados.

```
aws workspaces create-workspaces \  
  --  
  workspaces DirectoryId=d-926722edaf,UserName=Mary,BundleId=wsb-0zsvgp8fc,WorkspaceProperties
```

Salida:

```
{  
  "FailedRequests": [],  
  "PendingRequests": [  
    {  
      "WorkspaceId": "ws-dk1x zr417",  
      "DirectoryId": "d-926722edaf",  
      "UserName": "Mary",  
      "State": "PENDING",  
      "BundleId": "wsb-0zsvgp8fc"  
    }  
  ]  
}
```

### Ejemplo 3: Para crear un desacoplado por usuarios WorkSpace

En el siguiente `create-workspaces` ejemplo, se crea una disociación de usuarios WorkSpace al establecer el nombre de usuario en y especificar un WorkSpace nombre `[UNDEFINED]`, un identificador de directorio y un identificador de paquete.

```
aws workspaces create-workspaces \
  --workspaces
  DirectoryId=d-926722edaf,UserName='"[UNDEFINED]"',WorkspaceName=MaryWorkspace1,BundleId=wsb
```

Salida:

```
{
  "FailedRequests": [],
  "PendingRequests": [
    {
      "WorkspaceId": "ws-abcd1234",
      "DirectoryId": "d-926722edaf",
      "UserName": "[UNDEFINED]",
      "State": "PENDING",
      "BundleId": "wsb-0zsvgp8fc",
      "WorkspaceName": "MaryWorkspace1"
    }
  ]
}
```

Para obtener más información, consulte [Lanzar un escritorio virtual](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [CreateWorkspaces](#) la Referencia de AWS CLI comandos.

### **delete-tags**

En el siguiente ejemplo de código, se muestra cómo usar `delete-tags`.

#### AWS CLI

Para eliminar una etiqueta de un WorkSpace

En el siguiente `delete-tags` ejemplo, se elimina la etiqueta especificada de la especificada WorkSpace.



```
aws workspaces delete-tags \  
  --resource-id ws-dk1xzc417 \  
  --tag-keys Department
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [WorkSpaces los recursos sobre etiquetas](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [DeleteTags](#) la Referencia de AWS CLI comandos.

## deregister-workspace-directory

En el siguiente ejemplo de código, se muestra cómo usar `deregister-workspace-directory`.

### AWS CLI

Para anular el registro de un directorio

El siguiente `deregister-workspace-directory` ejemplo anula el registro del directorio especificado.

```
aws workspaces deregister-workspace-directory \  
  --directory-id d-926722edaf
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Registrar un directorio WorkSpaces](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [DeregisterWorkspaceDirectory](#) la Referencia de AWS CLI comandos.

## describe-tags

En el siguiente ejemplo de código, se muestra cómo usar `describe-tags`.

### AWS CLI

Para describir las etiquetas de un Workspace

En el siguiente `describe-tags` ejemplo, se describen las etiquetas del objeto especificado `Workspace`.

```
aws workspaces describe-tags \  
  --resource-id ws-dk1xzzr417
```

Salida:

```
{  
  "TagList": [  
    {  
      "Key": "Department",  
      "Value": "Finance"  
    }  
  ]  
}
```

Para obtener más información, consulta [WorkSpaces los recursos sobre etiquetas](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [DescribeTags](#) la Referencia de AWS CLI comandos.

## `describe-workspace-bundles`

En el siguiente ejemplo de código, se muestra cómo usar `describe-workspace-bundles`.

AWS CLI

Para publicar los paquetes proporcionados por Amazon

El siguiente `describe-workspace-bundles` ejemplo muestra los nombres y IDs los paquetes proporcionados por Amazon, en formato de tabla y ordenados por nombre.

```
aws workspaces describe-workspace-bundles \  
  --owner AMAZON \  
  --query "Bundles[*].[Name, BundleId]"
```

Salida:

```
[
```

```
[
  "Standard with Amazon Linux 2",
  "wsb-clj85qzj1"
],
[
  "Performance with Windows 10 (Server 2016 based)",
  "wsb-gm4d5tx2v"
],
[
  "PowerPro with Windows 7",
  "wsb-1pzkp0bx4"
],
[
  "Power with Amazon Linux 2",
  "wsb-2bs6k5lgn"
],
[
  "Graphics with Windows 10 (Server 2019 based)",
  "wsb-03gyjnfyy"
],
  ...
]
```

Para obtener más información, consulta [WorkSpaces los paquetes y las imágenes](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [DescribeWorkspaceBundles](#) la Referencia de AWS CLI comandos.

## describe-workspace-directories

En el siguiente ejemplo de código, se muestra cómo usar `describe-workspace-directories`.

### AWS CLI

Para describir un directorio registrado

El siguiente `describe-workspace-directories` ejemplo describe el directorio registrado especificado.

```
aws workspaces describe-workspace-directories \
  --directory-ids d-926722edaf
```

**Salida:**

```
{
  "Directories": [
    {
      "DirectoryId": "d-926722edaf",
      "Alias": "d-926722edaf",
      "DirectoryName": "example.com",
      "RegistrationCode": "WSpdx+9RJ8JT",
      "SubnetIds": [
        "subnet-9d19c4c6",
        "subnet-500d5819"
      ],
      "DnsIpAddresses": [
        "172.16.1.140",
        "172.16.0.30"
      ],
      "CustomerUserName": "Administrator",
      "IamRoleId": "arn:aws:iam::123456789012:role/workspaces_DefaultRole",
      "DirectoryType": "SIMPLE_AD",
      "WorkspaceSecurityGroupId": "sg-0d89e927e5645d7c5",
      "State": "REGISTERED",
      "WorkspaceCreationProperties": {
        "EnableWorkDocs": false,
        "EnableInternetAccess": false,
        "UserEnabledAsLocalAdministrator": true,
        "EnableMaintenanceMode": true
      },
      "WorkspaceAccessProperties": {
        "DeviceTypeWindows": "ALLOW",
        "DeviceTypeOsx": "ALLOW",
        "DeviceTypeWeb": "DENY",
        "DeviceTypeIos": "ALLOW",
        "DeviceTypeAndroid": "ALLOW",
        "DeviceTypeChromeOs": "ALLOW",
        "DeviceTypeZeroClient": "ALLOW",
        "DeviceTypeLinux": "DENY"
      },
      "Tenancy": "SHARED",
      "SelfservicePermissions": {
        "RestartWorkspace": "ENABLED",
        "IncreaseVolumeSize": "DISABLED",
        "ChangeComputeType": "DISABLED",
        "SwitchRunningMode": "DISABLED",

```

```

    "RebuildWorkspace": "DISABLED"
  }
}
]
}

```

Para obtener más información, consulte [Administrar directorios WorkSpaces](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [DescribeWorkspaceDirectories](#) la Referencia de AWS CLI comandos.

## describe-workspaces-connection-status

En el siguiente ejemplo de código, se muestra cómo usar `describe-workspaces-connection-status`.

### AWS CLI

Para describir el estado de conexión de un WorkSpace

El siguiente `describe-workspaces-connection-status` ejemplo describe el estado de conexión del especificado WorkSpace.

```

aws workspaces describe-workspaces-connection-status \
  --workspace-ids ws-dk1xzt417

```

Salida:

```

{
  "WorkspacesConnectionStatus": [
    {
      "WorkspaceId": "ws-dk1xzt417",
      "ConnectionState": "CONNECTED",
      "ConnectionStateCheckTimestamp": 1662526214.744
    }
  ]
}

```

Para obtener más información, consulta [Administra tu WorkSpaces](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [DescribeWorkspacesConnectionStatus](#) la Referencia de AWS CLI comandos.

## describe-workspaces

En el siguiente ejemplo de código, se muestra cómo usar describe-workspaces.

### AWS CLI

Para describir un Workspace

El siguiente describe-workspaces ejemplo describe lo especificado Workspace.

```
aws workspaces describe-workspaces \  
  --workspace-ids ws-dk1xzr417
```

Salida:

```
{  
  "Workspaces": [  
    {  
      "WorkspaceId": "ws-dk1xzr417",  
      "DirectoryId": "d-926722edaf",  
      "UserName": "Mary",  
      "IpAddress": "172.16.0.175",  
      "State": "STOPPED",  
      "BundleId": "wsb-0zsvgp8fc",  
      "SubnetId": "subnet-500d5819",  
      "ComputerName": "WSAMZN-RBSLTTD9",  
      "WorkspaceProperties": {  
        "RunningMode": "AUTO_STOP",  
        "RunningModeAutoStopTimeoutInMinutes": 60,  
        "RootVolumeSizeGib": 80,  
        "UserVolumeSizeGib": 10,  
        "ComputeTypeName": "VALUE"  
      },  
      "ModificationStates": []  
    }  
  ]  
}
```

Para obtener más información, consulta [Administra tu WorkSpaces](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [DescribeWorkspaces](#) la Referencia de AWS CLI comandos.

## **migrate-workspace**

En el siguiente ejemplo de código, se muestra cómo usar `migrate-workspace`.

### AWS CLI

Para migrar un WorkSpace

En el siguiente `migrate-workspace` ejemplo, se WorkSpace migra el paquete especificado al especificado.

```
aws workspaces migrate-workspace \  
  --source-workspace-id ws-dk1xzt417 \  
  --bundle-id wsb-j4dky1gs4
```

Salida:

```
{  
  "SourceWorkspaceId": "ws-dk1xzt417",  
  "TargetWorkspaceId": "ws-x5h11bkb5"  
}
```

Para obtener más información, consulte [Migrate a WorkSpace](#) en la Amazon WorkSpaces Administration Guide.

- Para API obtener más información, consulte [MigrateWorkspace](#) la Referencia de AWS CLI comandos.

## **modify-workspace-creation-properties**

En el siguiente ejemplo de código, se muestra cómo usar `modify-workspace-creation-properties`.

### AWS CLI

Para modificar una propiedad de WorkSpace creación de un directorio

El siguiente `modify-workspace-creation-properties` ejemplo habilita la `EnableInternetAccess` propiedad para el directorio especificado. Esto permite la asignación automática de direcciones IP públicas a las direcciones IP WorkSpaces creadas para el directorio.

```
aws workspaces modify-workspace-creation-properties \  
  --resource-id d-926722edaf \  
  --workspace-creation-properties EnableInternetAccess=true
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Actualizar los detalles de tu directorio WorkSpaces](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [ModifyWorkspaceCreationProperties](#) la Referencia de AWS CLI comandos.

## `modify-workspace-properties`

En el siguiente ejemplo de código, se muestra cómo usar `modify-workspace-properties`.

### AWS CLI

Para modificar el modo de ejecución de un WorkSpace

El siguiente `modify-workspace-properties` ejemplo establece el modo de ejecución del especificado WorkSpace en `AUTO_STOP`.

```
aws workspaces modify-workspace-properties \  
  --workspace-id ws-dk1xzt417 \  
  --workspace-properties RunningMode=AUTO_STOP
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Modificar a WorkSpace](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [ModifyWorkspaceProperties](#) la Referencia de AWS CLI comandos.

## `modify-workspace-state`

En el siguiente ejemplo de código, se muestra cómo usar `modify-workspace-state`.



## AWS CLI

Para modificar el estado de un Workspace

El siguiente `modify-workspace-state` ejemplo establece el estado del especificado Workspace en `ADMIN_MAINTENANCE`.

```
aws workspaces modify-workspace-state \  
  --workspace-id ws-dk1xzt417 \  
  --workspace-state ADMIN_MAINTENANCE
```

Este comando no genera ninguna salida.

Para obtener más información, consulta el [Workspace mantenimiento](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [ModifyWorkspaceState](#) la Referencia de AWS CLI comandos.

## **reboot-workspaces**

En el siguiente ejemplo de código, se muestra cómo usar `reboot-workspaces`.

## AWS CLI

Para reiniciar un Workspace

En el siguiente `reboot-workspaces` ejemplo, se reinicia el especificado Workspace.

```
aws workspaces reboot-workspaces \  
  --reboot-workspace-requests ws-dk1xzt417
```

Salida:

```
{  
  "FailedRequests": []  
}
```

Para obtener más información, consulta [Reboot a Workspace](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [RebootWorkspaces](#) la Referencia de AWS CLI comandos.

## rebuild-workspaces

En el siguiente ejemplo de código, se muestra cómo usar `rebuild-workspaces`.

### AWS CLI

Para reconstruir un WorkSpace

El siguiente `rebuild-workspaces` ejemplo reconstruye lo especificado WorkSpace.

```
aws workspaces rebuild-workspaces \  
  --rebuild-workspace-requests ws-dk1xzr417
```

Salida:

```
{  
  "FailedRequests": []  
}
```

Para obtener más información, consulte [Rebuild a WorkSpace](#) en la Amazon WorkSpaces Administration Guide.

- Para API obtener más información, consulte [RebuildWorkspaces](#) la Referencia de AWS CLI comandos.

## register-workspace-directory

En el siguiente ejemplo de código, se muestra cómo usar `register-workspace-directory`.

### AWS CLI

Para registrar un directorio

En el siguiente `register-workspace-directory` ejemplo, se registra el directorio especificado para su uso con Amazon WorkSpaces.

```
aws workspaces register-workspace-directory \  
  --directory-id d-926722edaf \  
  --
```

```
--no-enable-work-docs
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Registrar un directorio WorkSpaces](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [RegisterWorkspaceDirectory](#) la Referencia de AWS CLI comandos.

## restore-workspace

En el siguiente ejemplo de código, se muestra cómo usar `restore-workspace`.

AWS CLI

Para restaurar un WorkSpace

El siguiente `restore-workspace` ejemplo restaura lo especificado WorkSpace.

```
aws workspaces restore-workspace \  
  --workspace-id ws-dk1xzt417
```

Este comando no genera ninguna salida.

Para obtener más información, consulta [Restaurar a WorkSpace](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [RestoreWorkspace](#) la Referencia de AWS CLI comandos.

## start-workspaces

En el siguiente ejemplo de código, se muestra cómo usar `start-workspaces`.

AWS CLI

Para iniciar una AutoStop WorkSpace

En el siguiente `start-workspaces` ejemplo se inicia el especificado WorkSpace. WorkSpace Debe tener un modo de ejecución deAutoStop.

```
aws workspaces start-workspaces \  
  --start-workspace-requests WorkspaceId=ws-dk1x zr417
```

Salida:

```
{  
  "FailedRequests": []  
}
```

Para obtener más información, consulta [Stop and start an AutoStop WorkSpace](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [StartWorkspaces](#) la Referencia de AWS CLI comandos.

## stop-workspaces

En el siguiente ejemplo de código, se muestra cómo usar stop-workspaces.

### AWS CLI

Para detener un AutoStop WorkSpace

El siguiente stop-workspaces ejemplo detiene lo especificado WorkSpace. WorkSpace Debe tener un modo de ejecución deAutoStop.

```
aws workspaces stop-workspaces \  
  --stop-workspace-requests WorkspaceId=ws-dk1x zr417
```

Salida:

```
{  
  "FailedRequests": []  
}
```

Para obtener más información, consulta [Stop and start an AutoStop WorkSpace](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [StopWorkspaces](#) la Referencia de AWS CLI comandos.

## terminate-workspaces

En el siguiente ejemplo de código, se muestra cómo usar `terminate-workspaces`.

### AWS CLI

Para terminar un WorkSpace

En el siguiente `terminate-workspaces` ejemplo se termina el espacio de trabajo especificado.

```
aws workspaces terminate-workspaces \  
  --terminate-workspace-requests ws-dk1xzt417
```

Salida:

```
{  
  "FailedRequests": []  
}
```

Para obtener más información, consulta [Eliminar un Workspace](#) en la Guía de WorkSpaces administración de Amazon.

- Para API obtener más información, consulte [TerminateWorkspaces](#) la Referencia de AWS CLI comandos.

## Ejemplos de rayos X que utilizan AWS CLI

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso de AWS Command Line Interface X-Ray.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)

## Acciones

### batch-traces-get

En el siguiente ejemplo de código, se muestra cómo usar `batch-traces-get`.

#### AWS CLI

Para obtener una lista de trazas

En el siguiente `batch-get-traces` ejemplo, se recupera una lista de trazas especificada por un identificador. El registro de seguimiento completo incluye un documento para cada segmento, compilado a partir de todos los documentos de segmento recibidos que tienen el mismo ID de registro de seguimiento.

```
aws xray batch-get-traces \
  --trace-ids 1-5d82881a-0a9126e92a73e971eed891b9
```

Salida:

```
{
  "Traces": [
    {
      "Id": "1-5d82881a-0a9126e92a73e971eed891b9",
      "Duration": 0.232,
      "Segments": [
        {
          "Id": "54aff5735b12dd28",
          "Document": "{\"id\":\"54aff5735b12dd28\",\"name\":\
\"Scorekeep\",\"start_time\":1.568835610432E9,\"end_time\":1.568835610664E9,\
\"http\":{\"request\":{\"url\":\"http://scorekeep-env-1.m4fg2pfzpv.us-\
east-2.elasticbeanstalk.com/api/user\",\"method\":\"POST\",\"user_agent\":\
\"curl/7.59.0\",\"client_ip\":\"52.95.4.28\",\"x_forwarded_for\":true},\
\"response\":{\"status\":200}},\"aws\":{\"elastic_beanstalk\":{\"version_label\
\":\"Sample Application-1\",\"deployment_id\":3,\"environment_name\":\"Scorekeep-\
env-1\"},\"ec2\":{\"availability_zone\":\"us-east-2b\",\"instance_id\":\
\"i-0e3cf4d2de0f3f37a\"},\"xray\":{\"sdk_version\":\"1.1.0\",\"sdk\":\"X-Ray for\
Java\"}},\"service\":{\"runtime\":\"OpenJDK 64-Bit Server VM\",\"runtime_version\
\":\"1.8.0_222\"},\"trace_id\":\"1-5d82881a-0a9126e92a73e971eed891b9\",\
\"origin\":\"AWS:ElasticBeanstalk:Environment\",\"subsegments\":[\"id\":\
\"2d6900034ccfe558\",\"name\":\"DynamoDB\",\"start_time\":1.568835610658E9,\
\"end_time\":1.568835610664E9,\"http\":{\"response\":{\"status\":200,
```

```

{"content_length":61}},{"aws":{"table_name":"scorekeep-user","operation":
"UpdateItem","request_id":"TPEIDNDUR0MLP0V17U4A79555NVV4KQNS05AEMVJF66Q9ASUAAJG
","resource_names":["scorekeep-user"],"namespace":"aws"}}}
    },
    {
      "Id": "0f278b6334c34e6b",
      "Document": "{\"id\":\"0f278b6334c34e6b\",\"name\":
\"DynamoDB\",\"start_time\":1.568835610658E9,\"end_time\":1.568835610664E9,
\"parent_id\":\"2d6900034ccfe558\",\"inferred\":true,\"http\":{\"response
\":{\"status\":200,\"content_length\":61}},\"aws\":{\"table_name
\":\"scorekeep-user\",\"operation\":\"UpdateItem\",\"request_id\":
\"TPEIDNDUR0MLP0V17U4A79555NVV4KQNS05AEMVJF66Q9ASUAAJG\",\"resource_names\":
[\"scorekeep-user\"]},\"trace_id\":\"1-5d82881a-0a9126e92a73e971eed891b9\",\"origin
\":\"AWS::DynamoDB::Table\"}"
    }
  ]
}
],
"UnprocessedTraceIds": []
}

```

Para obtener más información, consulte [Uso de los AWS rayos X API con AWS CLI la Guía para desarrolladores de AWS X-Ray](#).

- Para API obtener más información, consulte [BatchTracesGet](#) la Referencia de AWS CLI comandos.

## create-group

En el siguiente ejemplo de código, se muestra cómo usar create-group.

### AWS CLI

Para crear un grupo

En el siguiente create-group ejemplo se crea un recurso de grupo denominado AdminGroup. El grupo obtiene una expresión de filtro que define los criterios del grupo como un segmento relacionado con un servicio específico que provoca un fallo o un error.

```

aws xray create-group \
  --group-name "AdminGroup" \
  --filter-expression "service(\"mydomain.com\") {fault OR error}"

```

Salida:

```
{
  "GroupName": "AdminGroup",
  "GroupARN": "arn:aws:xray:us-west-2:123456789012:group/AdminGroup/123456789",
  "FilterExpression": "service(\"mydomain.com\") {fault OR error}"
}
```

Para obtener más información, consulte [Configuración de los ajustes de muestreo, grupos y cifrado con AWS X-Ray API en la Guía para desarrolladores de AWS X-Ray](#).

- Para API obtener más información, consulte [CreateGroup](#) la Referencia de AWS CLI comandos.

## create-sampling-rule

En el siguiente ejemplo de código, se muestra cómo usar `create-sampling-rule`.

### AWS CLI

Para crear una regla de muestreo

El siguiente `create-sampling-rule` ejemplo crea una regla para controlar el comportamiento del muestreo en aplicaciones instrumentadas. Las reglas se proporcionan en un JSON archivo. La mayoría de los campos de la regla de muestreo son obligatorios para crear la regla.

```
aws xray create-sampling-rule \
  --cli-input-json file://9000-base-scorekeep.json
```

Contenidos de `9000-base-scorekeep.json`:

```
{
  "SamplingRule": {
    "RuleName": "base-scorekeep",
    "ResourceARN": "*",
    "Priority": 9000,
    "FixedRate": 0.1,
    "ReservoirSize": 5,
    "ServiceName": "Scorekeep",
    "ServiceType": "*",
    "Host": "*",
    "HTTPMethod": "*",
    "URLPath": "*"
  }
}
```



```
    "Version": 1
  }
}
```

Salida:

```
{
  "SamplingRuleRecord": {
    "SamplingRule": {
      "RuleName": "base-scorekeep",
      "RuleARN": "arn:aws:xray:us-west-2:123456789012:sampling-rule/base-
scorekeep",
      "ResourceARN": "*",
      "Priority": 9000,
      "FixedRate": 0.1,
      "ReservoirSize": 5,
      "ServiceName": "Scorekeep",
      "ServiceType": "*",
      "Host": "*",
      "HTTPMethod": "*",
      "URLPath": "*",
      "Version": 1,
      "Attributes": {}
    },
    "CreatedAt": 1530574410.0,
    "ModifiedAt": 1530574410.0
  }
}
```

Para obtener más información, consulte [Configuración de los ajustes de muestreo, grupos y cifrado con AWS X-Ray API en la Guía para desarrolladores de AWS X-Ray](#).

- Para API obtener más información, consulte [CreateSamplingRule](#) la Referencia de AWS CLI comandos.

## delete-group

En el siguiente ejemplo de código, se muestra cómo usar delete-group.

### AWS CLI

Para eliminar un grupo

En el siguiente `delete-group` ejemplo, se elimina el recurso de grupo especificado.

```
aws xray delete-group \  
  --group-name "AdminGroup" \  
  --group-arn "arn:aws:xray:us-east-2:123456789012:group/AdminGroup/123456789"
```

Este comando no genera ninguna salida.

Para obtener más información, consulte [Configuración de los ajustes de muestreo, grupos y cifrado con AWS X-Ray API en la Guía para desarrolladores de AWS X-Ray](#).

- Para API obtener más información, consulte [DeleteGroup](#) la Referencia de AWS CLI comandos.

## delete-sampling-rule

En el siguiente ejemplo de código, se muestra cómo usar `delete-sampling-rule`.

### AWS CLI

Para eliminar una regla de muestreo

En el siguiente `delete-sampling-rule` ejemplo, se elimina la regla de muestreo especificada. Puede especificar el grupo mediante el nombre del grupo o el grupoARN.

```
aws xray delete-sampling-rule \  
  --rule-name polling-scorekeep
```

Salida:

```
{  
  "SamplingRuleRecord": {  
    "SamplingRule": {  
      "RuleName": "polling-scorekeep",  
      "RuleARN": "arn:aws:xray:us-west-2:123456789012:sampling-rule/polling-  
scorekeep",  
      "ResourceARN": "*",  
      "Priority": 5000,  
      "FixedRate": 0.003,  
      "ReservoirSize": 0,  
      "ServiceName": "Scorekeep",  
      "ServiceType": "*",  
      "Host": "*",
```

```
        "HTTPMethod": "GET",
        "URLPath": "/api/state/*",
        "Version": 1,
        "Attributes": {}
    },
    "CreatedAt": 1530574399.0,
    "ModifiedAt": 1530574399.0
}
}
```

Para obtener más información, consulte [Configuración de los ajustes de muestreo, grupos y cifrado con AWS X-Ray API en la Guía para desarrolladores de AWS X-Ray](#).

- Para API obtener más información, consulte [DeleteSamplingRule](#) la Referencia de AWS CLI comandos.

## get-encryption-config

En el siguiente ejemplo de código, se muestra cómo usar `get-encryption-config`.

### AWS CLI

Para recuperar la configuración de cifrado

En el siguiente `get-encryption-config` ejemplo, se recupera la configuración de cifrado actual de los datos de AWS X-Ray.

```
aws xray get-encryption-config
```

Salida:

```
{
  "EncryptionConfig": {
    "KeyId": "ae4aa6d49-a4d8-9df9-a475-4ff6d7898456",
    "Status": "ACTIVE",
    "Type": "NONE"
  }
}
```

Para obtener más información, consulte [Configuración de los ajustes de muestreo, grupos y cifrado con AWS X-Ray API en la Guía para desarrolladores de AWS X-Ray](#).

- Para API obtener más información, consulte [GetEncryptionConfig](#) la Referencia de AWS CLI comandos.

## get-group

En el siguiente ejemplo de código, se muestra cómo usar `get-group`.

### AWS CLI

Para recuperar un grupo

En el siguiente `get-group` ejemplo, se muestran los detalles del recurso de grupo especificado. Los detalles incluyen el nombre del grupo, el grupo ARN y la expresión de filtro que define los criterios de ese grupo. Los grupos también se pueden recuperar mediante ARN.

```
aws xray get-group \  
  --group-name "AdminGroup"
```

Salida:

```
{  
  "Group": [  
    {  
      "GroupName": "AdminGroup",  
      "GroupARN": "arn:aws:xray:us-west-2:123456789012:group/  
AdminGroup/123456789",  
      "FilterExpression": "service(\"mydomain.com\") {fault OR error}"  
    }  
  ]  
}
```

Para obtener más información, consulte [Configuración de los ajustes de muestreo, grupos y cifrado con AWS X-Ray API en la Guía para desarrolladores de AWS X-Ray](#).

- Para API obtener más información, consulte [GetGroup](#) la Referencia de AWS CLI comandos.

## get-groups

En el siguiente ejemplo de código, se muestra cómo usar `get-groups`.

## AWS CLI

Para recuperar todos los grupos

En el siguiente ejemplo, se muestran los detalles de todos los grupos activos.

```
aws xray get-groups
```

Salida:

```
{
  "Groups": [
    {
      "GroupName": "AdminGroup",
      "GroupARN": "arn:aws:xray:us-west-2:123456789012:group/
AdminGroup/123456789",
      "FilterExpression": "service(\"example.com\") {fault OR error}"
    },
    {
      "GroupName": "SDETGroup",
      "GroupARN": "arn:aws:xray:us-west-2:123456789012:group/
SDETGroup/987654321",
      "FilterExpression": "responsetime > 2"
    }
  ]
}
```

Para obtener más información, consulte [Configuración de los ajustes de muestreo, grupos y cifrado con AWS X-Ray API en la Guía para desarrolladores de AWS X-Ray](#).

- Para API obtener más información, consulte [GetGroups](#) la Referencia de AWS CLI comandos.

## get-sampling-rules

En el siguiente ejemplo de código, se muestra cómo usar `get-sampling-rules`.

## AWS CLI

Para recuperar todas las reglas de muestreo

En el siguiente `get-sampling-rules` ejemplo, se muestran los detalles de todas las reglas de muestreo disponibles. :

**aws xray get-sampling-rules**

Salida:

```
{
  "SamplingRuleRecords": [
    {
      "SamplingRule": {
        "RuleName": "Default",
        "RuleARN": "arn:aws:xray:us-east-1::sampling-rule/Default",
        "ResourceARN": "*",
        "Priority": 10000,
        "FixedRate": 0.01,
        "ReservoirSize": 0,
        "ServiceName": "*",
        "ServiceType": "*",
        "Host": "*",
        "HTTPMethod": "*",
        "URLPath": "*",
        "Version": 1,
        "Attributes": {}
      },
      "CreatedAt": 0.0,
      "ModifiedAt": 1530558121.0
    },
    {
      "SamplingRule": {
        "RuleName": "base-scorekeep",
        "RuleARN": "arn:aws:xray:us-east-1::sampling-rule/base-scorekeep",
        "ResourceARN": "*",
        "Priority": 9000,
        "FixedRate": 0.1,
        "ReservoirSize": 2,
        "ServiceName": "Scorekeep",
        "ServiceType": "*",
        "Host": "*",
        "HTTPMethod": "*",
        "URLPath": "*",
        "Version": 1,
        "Attributes": {}
      },
      "CreatedAt": 1530573954.0,
      "ModifiedAt": 1530920505.0
    }
  ]
}
```

```

    },
    {
      "SamplingRule": {
        "RuleName": "polling-scorekeep",
        "RuleARN": "arn:aws:xray:us-east-1::sampling-rule/polling-
scorekeep",
        "ResourceARN": "*",
        "Priority": 5000,
        "FixedRate": 0.003,
        "ReservoirSize": 0,
        "ServiceName": "Scorekeep",
        "ServiceType": "*",
        "Host": "*",
        "HTTPMethod": "GET",
        "URLPath": "/api/state/*",
        "Version": 1,
        "Attributes": {}
      },
      "CreatedAt": 1530918163.0,
      "ModifiedAt": 1530918163.0
    }
  ]
}

```

Para obtener más información, consulte [Uso de las reglas de muestreo con los rayos X API](#) en la Guía para desarrolladores de AWS X-Ray.

- Para API obtener más información, consulte [GetSamplingRules](#) la Referencia de AWS CLI comandos.

## get-sampling-targets

En el siguiente ejemplo de código, se muestra cómo usar `get-sampling-targets`.

### AWS CLI

Para solicitar una cuota de muestreo

En el siguiente `get-sampling-targets` ejemplo, se solicita una cuota de muestreo para las reglas que el servicio utiliza para muestrear las solicitudes. La respuesta de AWS X-Ray incluye una cuota que se puede utilizar en lugar de pedir prestado al embalse.

```
aws xray get-sampling-targets \
```

```
--sampling-statistics-documents '[ { "RuleName": "base-scorekeep", "ClientID":
"ABCDEF1234567890ABCDEF10", "Timestamp": "2018-07-07T00:20:06", "RequestCount": 110,
"SampledCount": 20, "BorrowCount": 10 }, { "RuleName": "polling-scorekeep", 31,
"BorrowCount": 0 } ]'
```

Salida:

```
{
  "SamplingTargetDocuments": [
    {
      "RuleName": "base-scorekeep",
      "FixedRate": 0.1,
      "ReservoirQuota": 2,
      "ReservoirQuotaTTL": 1530923107.0,
      "Interval": 10
    },
    {
      "RuleName": "polling-scorekeep",
      "FixedRate": 0.003,
      "ReservoirQuota": 0,
      "ReservoirQuotaTTL": 1530923107.0,
      "Interval": 10
    }
  ],
  "LastRuleModification": 1530920505.0,
  "UnprocessedStatistics": []
}
```

Para obtener más información, consulte [Uso de las reglas de muestreo con los rayos X API](#) en la Guía para desarrolladores de AWS X-Ray.

- Para API obtener más información, consulte [GetSamplingTargets](#) la Referencia de AWS CLI comandos.

## get-service-graph

En el siguiente ejemplo de código, se muestra cómo usar `get-service-graph`.

AWS CLI

Para obtener un gráfico de servicio



En el siguiente ejemplo, se muestra un documento dentro de un período de tiempo específico que describe los servicios que procesan las solicitudes entrantes y los servicios posteriores a los que llaman en consecuencia. :

```
aws xray get-service-graph \  
  --start-time 1568835392.0 \  
  --end-time 1568835446.0
```

Salida:

```
{  
  "Services": [  
    {  
      "ReferenceId": 0,  
      "Name": "Scorekeep",  
      "Names": [  
        "Scorekeep"  
      ],  
      "Root": true,  
      "Type": "AWS::ElasticBeanstalk::Environment",  
      "State": "active",  
      "StartTime": 1568835392.0,  
      "EndTime": 1568835446.0,  
      "Edges": [  
        {  
          "ReferenceId": 1,  
          "StartTime": 1568835392.0,  
          "EndTime": 1568835446.0,  
          "SummaryStatistics": {  
            "OkCount": 14,  
            "ErrorStatistics": {  
              "ThrottleCount": 0,  
              "OtherCount": 0,  
              "TotalCount": 0  
            },  
            "FaultStatistics": {  
              "OtherCount": 0,  
              "TotalCount": 0  
            },  
            "TotalCount": 14,  
            "TotalResponseTime": 0.13  
          },  
          "ResponseTimeHistogram": [  

```

```
        {
            "Value": 0.008,
            "Count": 1
        },
        {
            "Value": 0.005,
            "Count": 7
        },
        {
            "Value": 0.009,
            "Count": 1
        },
        {
            "Value": 0.021,
            "Count": 1
        },
        {
            "Value": 0.038,
            "Count": 1
        },
        {
            "Value": 0.007,
            "Count": 1
        },
        {
            "Value": 0.006,
            "Count": 2
        }
    ],
    "Aliases": []
},

... TRUNCATED FOR BREVITY ...

]
}
],
"StartTime": 1568835392.0,
"EndTime": 1568835446.0,
"ContainsOldGroupVersions": false
}
```

Para obtener más información, consulte [Uso de los AWS rayos X API con AWS CLI](#) la Guía para desarrolladores de AWS X-Ray.

- Para API obtener más información, consulte [GetServiceGraph](#) la Referencia de AWS CLI comandos.

## get-trace-summaries

En el siguiente ejemplo de código, se muestra cómo usar `get-trace-summaries`.

### AWS CLI

Para obtener un resumen del rastreo

En el siguiente `get-trace-summaries` ejemplo, se recuperan IDs los metadatos de los seguimientos disponibles dentro de un período de tiempo específico.

```
aws xray get-trace-summaries \
  --start-time 1568835392.0 \
  --end-time 1568835446.0
```

Salida:

```
[
  "http://scorekeep-env-1.123456789.us-east-2.elasticbeanstalk.com/api/move/
  VSAE93HF/GSSD2NTB/DP0PCC09",
  "http://scorekeep-env-1.123456789.us-east-2.elasticbeanstalk.com/api/move/
  GCQ2B35P/FREELDFT/4LRE643M",
  "http://scorekeep-env-1.123456789.us-east-2.elasticbeanstalk.com/api/game/
  VSAE93HF/GSSD2NTB/starttime/1568835513",
  "http://scorekeep-env-1.123456789.us-east-2.elasticbeanstalk.com/api/
  move/4MQNA5NN/L99KK2RF/null"
]
```

Para obtener más información, consulte [Uso de los AWS rayos X API con AWS CLI](#) la Guía para desarrolladores de AWS X-Ray.

- Para API obtener más información, consulte [GetTraceSummaries](#) la Referencia de AWS CLI comandos.

## put-encryption-config

En el siguiente ejemplo de código, se muestra cómo usar `put-encryption-config`.

### AWS CLI

Para actualizar la configuración de cifrado

Lo siguiente `put-encryption-config` example updates the encryption configuration for AWS X-Ray data to use the default AWS managed KMS key `aws/xray`.

```
aws xray put-encryption-config \  
  --type KMS \  
  --key-id alias/aws/xray
```

Salida:

```
{  
  "EncryptionConfig": {  
    "KeyId": "arn:aws:kms:us-west-2:123456789012:key/c234g4e8-39e9-4gb0-84e2-  
b0ea215cbba5",  
    "Status": "UPDATING",  
    "Type": "KMS"  
  }  
}
```

Para obtener más información, consulte [Configuración de los ajustes de muestreo, grupos y cifrado con AWS X-Ray API en la Guía para desarrolladores de AWS X-Ray](#).

- Para API obtener más información, consulte [PutEncryptionConfig](#) la Referencia de AWS CLI comandos.

## put-trace-segments

En el siguiente ejemplo de código, se muestra cómo usar `put-trace-segments`.

### AWS CLI

Para cargar un segmento

En el siguiente `put-trace-segments` ejemplo, se cargan documentos segmentados a AWS X-Ray. El documento segmentado se consume como una lista de documentos JSON segmentados.

```
aws xray put-trace-segments \
  --trace-segment-documents '{"id":"20312a0e2b8809f4","name
  \":"DynamoDB","trace_id":"1-5832862d-a43aafded3334a971fe312db",
  \start_time":1.479706157195E9,"end_time":1.479706157202E9,"parent_id":
  \79736b962fe3239e","http":{"response":{"content_length":60,"status
  \":200}},"inferred":true,"aws":{"consistent_read":false,"table_name
  \":"scorekeep-session-xray","operation":"GetItem","request_id":
  \SCAU230M6M8F038UASGC7785ARVV4KQNS05AEMVJF66Q9ASUAAJG","resource_names":
  \["scorekeep-session-xray"]},"origin":"AWS::DynamoDB::Table"}'
```

Salida:

```
{
  "UnprocessedTraceSegments": []
}
```

Para obtener más información, consulte [Envío de datos de rastreo a AWS X-Ray](#) en la Guía para desarrolladores de AWS X-Ray.

- Para API obtener más información, consulte [PutTraceSegments](#) la Referencia de AWS CLI comandos.

## update-group

En el siguiente ejemplo de código, se muestra cómo usar `update-group`.

### AWS CLI

Para actualizar un grupo

En el siguiente `update-group` ejemplo, se actualizan los criterios para aceptar los rastreos en el grupo denominado `AdminGroup`. Puede especificar el grupo deseado mediante el nombre del grupo o el grupoARN.

```
aws xray update-group \
  --group-name "AdminGroup" \
  --group-arn "arn:aws:xray:us-west-2:123456789012:group/AdminGroup/123456789" \
  --filter-expression "service(\"mydomain.com\") {fault}"
```

Salida:

```
{
  "GroupName": "AdminGroup",
  "GroupARN": "arn:aws:xray:us-east-2:123456789012:group/AdminGroup/123456789",
  "FilterExpression": "service(\"mydomain.com\") {fault}"
}
```

Para obtener más información, consulte [Configuración de los ajustes de muestreo, grupos y cifrado con AWS X-Ray API en la Guía para desarrolladores de AWS X-Ray](#).

- Para API obtener más información, consulte [UpdateGroup](#) la Referencia de AWS CLI comandos.

## update-sampling-rule

En el siguiente ejemplo de código, se muestra cómo usar `update-sampling-rule`.

AWS CLI

Para actualizar una regla de muestreo

El siguiente `update-sampling-rule` ejemplo modifica la configuración de una regla de muestreo. Las reglas se consumen desde un JSON archivo. Solo son obligatorios los campos que se están actualizando.

```
aws xray update-sampling-rule \
  --cli-input-json file://1000-default.json
```

Contenidos de `1000-default.json`:

```
{
  "SamplingRuleUpdate": {
    "RuleName": "Default",
    "FixedRate": 0.01,
    "ReservoirSize": 0
  }
}
```

Salida:

```
{
```

```
"SamplingRuleRecords": [
  {
    "SamplingRule": {
      "RuleName": "Default",
      "RuleARN": "arn:aws:xray:us-west-2:123456789012:sampling-rule/
Default",
      "ResourceARN": "*",
      "Priority": 10000,
      "FixedRate": 0.01,
      "ReservoirSize": 0,
      "ServiceName": "*",
      "ServiceType": "*",
      "Host": "*",
      "HTTPMethod": "*",
      "URLPath": "*",
      "Version": 1,
      "Attributes": {}
    },
    "CreatedAt": 0.0,
    "ModifiedAt": 1529959993.0
  }
]
```

Para obtener más información, consulte [Configuración de los ajustes de muestreo, grupos y cifrado con AWS X-Ray API en la Guía para desarrolladores de AWS X-Ray](#).

- Para API obtener más información, consulte [UpdateSamplingRule](#) la Referencia de AWS CLI comandos.

## AWS CLI con ejemplos de código de script Bash

Los ejemplos de código de este tema muestran cómo utilizar el script AWS Command Line Interface with Bash with. AWS

Los conceptos básicos son ejemplos de código que muestran cómo realizar las operaciones esenciales dentro de un servicio.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Los escenarios son ejemplos de código que muestran cómo llevar a cabo una tarea específica a través de llamadas a varias funciones dentro del servicio o combinado con otros Servicios de AWS.

## Servicios

- [Ejemplos AWS CLI de DynamoDB que se utilizan con el script Bash](#)
- [EC2Ejemplos de Amazon que utilizan AWS CLI el script Bash](#)
- [HealthImaging ejemplos de uso AWS CLI con el script Bash](#)
- [IAMejemplos de uso AWS CLI con el script Bash](#)
- [Ejemplos de Amazon S3 que se utilizan AWS CLI con el script Bash](#)
- [AWS STS ejemplos de uso AWS CLI con el script Bash](#)

## Ejemplos AWS CLI de DynamoDB que se utilizan con el script Bash

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el script with Bash AWS Command Line Interface con DynamoDB.

Los conceptos básicos son ejemplos de código que muestran cómo realizar las operaciones esenciales dentro de un servicio.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Conceptos básicos](#)
- [Acciones](#)

## Conceptos básicos

Aprenda los conceptos básicos

En el siguiente ejemplo de código, se muestra cómo:

- Creación de una tabla que pueda contener datos de películas.



- Colocar, obtener y actualizar una sola película en la tabla.
- Escribe los datos de la película en la tabla a partir de un JSON archivo de ejemplo.
- Consultar películas que se hayan estrenado en un año determinado.
- Buscar películas que se hayan estrenado en un intervalo de años.
- Eliminación de una película de la tabla y, a continuación, eliminar la tabla.

## AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

## El escenario de introducción a DynamoDB.

```
#####
# function dynamodb_getting_started_movies
#
# Scenario to create an Amazon DynamoDB table and perform a series of operations on
# the table.
#
# Returns:
#     0 - If successful.
#     1 - If an error occurred.
#####
function dynamodb_getting_started_movies() {

    source ./dynamodb_operations.sh

    key_schema_json_file="dynamodb_key_schema.json"
    attribute_definitions_json_file="dynamodb_attr_def.json"
    item_json_file="movie_item.json"
    key_json_file="movie_key.json"
    batch_json_file="batch.json"
    attribute_names_json_file="attribute_names.json"
    attributes_values_json_file="attribute_values.json"

    echo_repeat "*" 88
    echo
```

```
echo "Welcome to the Amazon DynamoDB getting started demo."
echo
echo_repeat "*" 88
echo

local table_name
echo -n "Enter a name for a new DynamoDB table: "
get_input
table_name=$get_input_result

local provisioned_throughput="ReadCapacityUnits=5,WriteCapacityUnits=5"

echo '[
{"AttributeName": "year", "KeyType": "HASH"},
 {"AttributeName": "title", "KeyType": "RANGE"}
]' >"$key_schema_json_file"

echo '[
{"AttributeName": "year", "AttributeType": "N"},
 {"AttributeName": "title", "AttributeType": "S"}
]' >"$attribute_definitions_json_file"

if dynamodb_create_table -n "$table_name" -a "$attribute_definitions_json_file" \
  -k "$key_schema_json_file" -p "$provisioned_throughput" 1>/dev/null; then
  echo "Created a DynamoDB table named $table_name"
else
  errecho "The table failed to create. This demo will exit."
  clean_up
  return 1
fi

echo "Waiting for the table to become active...."

if dynamodb_wait_table_active -n "$table_name"; then
  echo "The table is now active."
else
  errecho "The table failed to become active. This demo will exit."
  cleanup "$table_name"
  return 1
fi

echo
echo_repeat "*" 88
echo
```

```
echo -n "Enter the title of a movie you want to add to the table: "
get_input
local added_title
added_title=$get_input_result

local added_year
get_int_input "What year was it released? "
added_year=$get_input_result

local rating
get_float_input "On a scale of 1 - 10, how do you rate it? " "1" "10"
rating=$get_input_result

local plot
echo -n "Summarize the plot for me: "
get_input
plot=$get_input_result

echo '{
  "year": {"N" : ""$added_year""},
  "title": {"S" : ""$added_title""},
  "info": {"M" : {"plot": {"S" : ""$plot""}, "rating": {"N" : ""$rating""} } }
}' >"$item_json_file"

if dynamodb_put_item -n "$table_name" -i "$item_json_file"; then
  echo "The movie '$added_title' was successfully added to the table
'$table_name'."
else
  errecho "Put item failed. This demo will exit."
  clean_up "$table_name"
  return 1
fi

echo
echo_repeat "*" 88
echo

echo "Let's update your movie '$added_title'."
get_float_input "You rated it $rating, what new rating would you give it? " "1"
"10"
rating=$get_input_result

echo -n "You summarized the plot as '$plot'."
```

```
echo "What would you say now? "  
get_input  
plot=$get_input_result  
  
echo '{  
  "year": {"N" : ""$added_year""},  
  "title": {"S" : ""$added_title""}  
}' >"$key_json_file"  
  
echo '{  
  "r": {"N" : ""$rating""},  
  "p": {"S" : ""$plot""}  
}' >"$item_json_file"  
  
local update_expression="SET info.rating = :r, info.plot = :p"  
  
if dynamodb_update_item -n "$table_name" -k "$key_json_file" -e  
"$update_expression" -v "$item_json_file"; then  
  echo "Updated '$added_title' with new attributes."  
else  
  errecho "Update item failed. This demo will exit."  
  clean_up "$table_name"  
  return 1  
fi  
  
echo  
echo_repeat "*" 88  
echo  
  
echo "We will now use batch write to upload 150 movie entries into the table."  
  
local batch_json  
for batch_json in movie_files/movies_*.json; do  
  echo "{ \"$table_name\" : $(<"$batch_json") }" >"$batch_json_file"  
  if dynamodb_batch_write_item -i "$batch_json_file" 1>/dev/null; then  
    echo "Entries in $batch_json added to table."  
  else  
    errecho "Batch write failed. This demo will exit."  
    clean_up "$table_name"  
    return 1  
  fi  
done  
  
local title="The Lord of the Rings: The Fellowship of the Ring"
```

```
local year="2001"

if get_yes_no_input "Let's move on...do you want to get info about '$title'? (y/n)
"; then
    echo '{
"year": {"N" :"""$year"""},
"title": {"S" : """$title"""}
}' >"$key_json_file"
    local info
    info=$(dynamodb_get_item -n "$table_name" -k "$key_json_file")

    # shellcheck disable=SC2181
    if [[ ${?} -ne 0 ]]; then
        errecho "Get item failed. This demo will exit."
        clean_up "$table_name"
        return 1
    fi

    echo "Here is what I found:"
    echo "$info"
fi

local ask_for_year=true
while [[ "$ask_for_year" == true ]]; do
    echo "Let's get a list of movies released in a given year."
    get_int_input "Enter a year between 1972 and 2018: " "1972" "2018"
    year=$get_input_result
    echo '{
"#n": "year"
}' >"$attribute_names_json_file"

    echo '{
":v": {"N" :"""$year"""}
}' >"$attributes_values_json_file"

    response=$(dynamodb_query -n "$table_name" -k "#n=:v" -a
"$attribute_names_json_file" -v "$attributes_values_json_file")

    # shellcheck disable=SC2181
    if [[ ${?} -ne 0 ]]; then
        errecho "Query table failed. This demo will exit."
        clean_up "$table_name"
        return 1
    fi
fi
```

```
    echo "Here is what I found:"
    echo "$response"

    if ! get_yes_no_input "Try another year? (y/n) "; then
        ask_for_year=false
    fi
done

echo "Now let's scan for movies released in a range of years. Enter a year: "
get_int_input "Enter a year between 1972 and 2018: " "1972" "2018"
local start=$get_input_result

get_int_input "Enter another year: " "1972" "2018"
local end=$get_input_result

echo '{
  "#n": "year"
}' >"$attribute_names_json_file"

echo '{
  ":v1": {"N" : ""$start""},
  ":v2": {"N" : ""$end""}
}' >"$attributes_values_json_file"

response=$(dynamodb_scan -n "$table_name" -f "#n BETWEEN :v1 AND :v2" -a
"$attribute_names_json_file" -v "$attributes_values_json_file")

# shellcheck disable=SC2181
if [[ ${?} -ne 0 ]]; then
    errecho "Scan table failed. This demo will exit."
    clean_up "$table_name"
    return 1
fi

echo "Here is what I found:"
echo "$response"

echo
echo_repeat "*" 88
echo

echo "Let's remove your movie '$added_title' from the table."
```

```

if get_yes_no_input "Do you want to remove '$added_title'? (y/n) "; then
    echo '{
"year": {"N" : ""'$added_year'""},
"title": {"S" : ""'$added_title'""}
}' >"$key_json_file"

    if ! dynamodb_delete_item -n "$table_name" -k "$key_json_file"; then
        errecho "Delete item failed. This demo will exit."
        clean_up "$table_name"
        return 1
    fi
fi

if get_yes_no_input "Do you want to delete the table '$table_name'? (y/n) "; then
    if ! clean_up "$table_name"; then
        return 1
    fi
else
    if ! clean_up; then
        return 1
    fi
fi

return 0
}

```

Las funciones de DynamoDB utilizadas en este escenario.

```

#####
# function dynamodb_create_table
#
# This function creates an Amazon DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table to create.
#     -a attribute_definitions -- JSON file path of a list of attributes and their
types.
#     -k key_schema -- JSON file path of a list of attributes and their key types.
#     -p provisioned_throughput -- Provisioned throughput settings for the table.
#
# Returns:
#     0 - If successful.

```

```

#      1 - If it fails.
#####
function dynamodb_create_table() {
    local table_name attribute_definitions key_schema provisioned_throughput response
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_create_table"
        echo "Creates an Amazon DynamoDB table."
        echo " -n table_name -- The name of the table to create."
        echo " -a attribute_definitions -- JSON file path of a list of attributes and
their types."
        echo " -k key_schema -- JSON file path of a list of attributes and their key
types."
        echo " -p provisioned_throughput -- Provisioned throughput settings for the
table."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:a:k:p:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            a) attribute_definitions="${OPTARG}" ;;
            k) key_schema="${OPTARG}" ;;
            p) provisioned_throughput="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$table_name" ]]; then
        errecho "ERROR: You must provide a table name with the -n parameter."
    fi
}

```



```
usage
return 1
fi

if [[ -z "$attribute_definitions" ]]; then
    errecho "ERROR: You must provide an attribute definitions json file path the -a
parameter."
    usage
    return 1
fi

if [[ -z "$key_schema" ]]; then
    errecho "ERROR: You must provide a key schema json file path the -k parameter."
    usage
    return 1
fi

if [[ -z "$provisioned_throughput" ]]; then
    errecho "ERROR: You must provide a provisioned throughput json file path the -p
parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    table_name:    $table_name"
iecho "    attribute_definitions:    $attribute_definitions"
iecho "    key_schema:    $key_schema"
iecho "    provisioned_throughput:    $provisioned_throughput"
iecho ""

response=$(aws dynamodb create-table \
    --table-name "$table_name" \
    --attribute-definitions file://"$attribute_definitions" \
    --key-schema file://"$key_schema" \
    --provisioned-throughput "$provisioned_throughput")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-table operation failed.$response"
    return 1
fi
```

```

    return 0
}

#####
# function dynamodb_describe_table
#
# This function returns the status of a DynamoDB table.
#
# Parameters:
#     -n table_name  -- The name of the table.
#
# Response:
#     - TableStatus:
#     And:
#     0 - Table is active.
#     1 - If it fails.
#####
function dynamodb_describe_table {
    local table_name
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_describe_table"
        echo "Describe the status of a DynamoDB table."
        echo "  -n table_name  -- The name of the table."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
        esac
    done
}

```

```
        ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

local table_status
table_status=$(
    aws dynamodb describe-table \
        --table-name "$table_name" \
        --output text \
        --query 'Table.TableStatus'
)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log "$error_code"
    errecho "ERROR: AWS reports describe-table operation failed.$table_status"
    return 1
fi

echo "$table_status"

return 0
}

#####
# function dynamodb_put_item
#
# This function puts an item into a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -i item      -- Path to json file containing the item values.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
```

```
#####
function dynamodb_put_item() {
    local table_name item response
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_put_item"
        echo "Put an item into a DynamoDB table."
        echo " -n table_name -- The name of the table."
        echo " -i item -- Path to json file containing the item values."
        echo ""
    }

    while getopt "n:i:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            i) item="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$table_name" ]]; then
        errecho "ERROR: You must provide a table name with the -n parameter."
        usage
        return 1
    fi

    if [[ -z "$item" ]]; then
        errecho "ERROR: You must provide an item with the -i parameter."
        usage
        return 1
    fi
}
#####
```

```

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  item:  $item"
iecho ""
iecho ""

response=$(aws dynamodb put-item \
  --table-name "$table_name" \
  --item file://" $item")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports put-item operation failed.$response"
  return 1
fi

return 0
}

#####
# function dynamodb_update_item
#
# This function updates an item in a DynamoDB table.
#
#
# Parameters:
#   -n table_name  -- The name of the table.
#   -k keys        -- Path to json file containing the keys that identify the item to
#   update.
#   -e update expression  -- An expression that defines one or more attributes
#   to be updated.
#   -v values      -- Path to json file containing the update values.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
function dynamodb_update_item() {
  local table_name keys update_expression values response
  local option OPTARG # Required to use getopt command in a function.

```

```
#####
# Function usage explanation
#####
function usage() {
    echo "function dynamodb_update_item"
    echo "Update an item in a DynamoDB table."
    echo " -n table_name -- The name of the table."
    echo " -k keys -- Path to json file containing the keys that identify the item
to update."
    echo " -e update expression -- An expression that defines one or more
attributes to be updated."
    echo " -v values -- Path to json file containing the update values."
    echo ""
}

while getopts "n:k:e:v:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        k) keys="${OPTARG}" ;;
        e) update_expression="${OPTARG}" ;;
        v) values="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$keys" ]]; then
    errecho "ERROR: You must provide a keys json file path the -k parameter."
    usage

```

```

    return 1
fi
if [[ -z "$update_expression" ]]; then
    errecho "ERROR: You must provide an update expression with the -e parameter."
    usage
    return 1
fi

if [[ -z "$values" ]]; then
    errecho "ERROR: You must provide a values json file path the -v parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  keys:       $keys"
iecho "  update_expression:  $update_expression"
iecho "  values:     $values"

response=$(aws dynamodb update-item \
  --table-name "$table_name" \
  --key file://" $keys" \
  --update-expression "$update_expression" \
  --expression-attribute-values file://" $values")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports update-item operation failed.$response"
    return 1
fi

return 0
}

#####
# function dynamodb_batch_write_item
#
# This function writes a batch of items into a DynamoDB table.
#
# Parameters:

```

```

#     -i item -- Path to json file containing the items to write.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_batch_write_item() {
    local item response
    local option OPTARG # Required to use getopt command in a function.

#####
# Function usage explanation
#####
function usage() {
    echo "function dynamodb_batch_write_item"
    echo "Write a batch of items into a DynamoDB table."
    echo " -i item -- Path to json file containing the items to write."
    echo ""
}
while getopt "i:h" option; do
    case "${option}" in
        i) item="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$item" ]]; then
    errecho "ERROR: You must provide an item with the -i parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    table_name:  $table_name"
iecho "    item:       $item"

```



```

iecho ""

response=$(aws dynamodb batch-write-item \
  --request-items file://"${item}")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports batch-write-item operation failed.$response"
  return 1
fi

return 0
}

#####
# function dynamodb_get_item
#
# This function gets an item from a DynamoDB table.
#
# Parameters:
#   -n table_name  -- The name of the table.
#   -k keys        -- Path to json file containing the keys that identify the item to
#   get.
#   [-q query]    -- Optional JMESPath query expression.
#
# Returns:
#   The item as text output.
#
# And:
#   0 - If successful.
#   1 - If it fails.
#####
function dynamodb_get_item() {
  local table_name keys query response
  local option OPTARG # Required to use getopt command in a function.

  # #####
  # Function usage explanation
  #####
  function usage() {
    echo "function dynamodb_get_item"
    echo "Get an item from a DynamoDB table."
    echo " -n table_name  -- The name of the table."
  }
}

```

```

    echo " -k keys -- Path to json file containing the keys that identify the item
to get."
    echo " [-q query] -- Optional JMESPath query expression."
    echo ""
}
query=""
while getopts "n:k:q:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        k) keys="${OPTARG}" ;;
        q) query="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$keys" ]]; then
    errecho "ERROR: You must provide a keys json file path the -k parameter."
    usage
    return 1
fi

if [[ -n "$query" ]]; then
    response=$(aws dynamodb get-item \
        --table-name "$table_name" \
        --key file://"${keys}" \
        --output text \
        --query "$query")
else
    response=$(

```

```

    aws dynamodb get-item \
        --table-name "$table_name" \
        --key file://"keys" \
        --output text
    )
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports get-item operation failed.$response"
    return 1
fi

if [[ -n "$query" ]]; then
    echo "$response" | sed "/^\t/s/\t//1" # Remove initial tab that the JMSEPath
query inserts on some strings.
else
    echo "$response"
fi

return 0
}

#####
# function dynamodb_query
#
# This function queries a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -k key_condition_expression -- The key condition expression.
#     -a attribute_names -- Path to JSON file containing the attribute names.
#     -v attribute_values -- Path to JSON file containing the attribute values.
#     [-p projection_expression] -- Optional projection expression.
#
# Returns:
#     The items as json output.
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_query() {

```

```

local table_name key_condition_expression attribute_names attribute_values
projection_expression response
local option OPTARG # Required to use getopt command in a function.

# #####
# Function usage explanation
# #####
function usage() {
    echo "function dynamodb_query"
    echo "Query a DynamoDB table."
    echo " -n table_name -- The name of the table."
    echo " -k key_condition_expression -- The key condition expression."
    echo " -a attribute_names -- Path to JSON file containing the attribute names."
    echo " -v attribute_values -- Path to JSON file containing the attribute
values."
    echo " [-p projection_expression] -- Optional projection expression."
    echo ""
}

while getopt "n:k:a:v:p:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        k) key_condition_expression="${OPTARG}" ;;
        a) attribute_names="${OPTARG}" ;;
        v) attribute_values="${OPTARG}" ;;
        p) projection_expression="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

```

```
if [[ -z "$key_condition_expression" ]]; then
    errecho "ERROR: You must provide a key condition expression with the -k
parameter."
    usage
    return 1
fi

if [[ -z "$attribute_names" ]]; then
    errecho "ERROR: You must provide a attribute names with the -a parameter."
    usage
    return 1
fi

if [[ -z "$attribute_values" ]]; then
    errecho "ERROR: You must provide a attribute values with the -v parameter."
    usage
    return 1
fi

if [[ -z "$projection_expression" ]]; then
    response=$(aws dynamodb query \
        --table-name "$table_name" \
        --key-condition-expression "$key_condition_expression" \
        --expression-attribute-names file://"${attribute_names}" \
        --expression-attribute-values file://"${attribute_values}")
else
    response=$(aws dynamodb query \
        --table-name "$table_name" \
        --key-condition-expression "$key_condition_expression" \
        --expression-attribute-names file://"${attribute_names}" \
        --expression-attribute-values file://"${attribute_values}" \
        --projection-expression "$projection_expression")
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports query operation failed.$response"
    return 1
fi

echo "$response"
```

```

    return 0
}

#####
# function dynamodb_scan
#
# This function scans a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -f filter_expression -- The filter expression.
#     -a expression_attribute_names -- Path to JSON file containing the expression
attribute names.
#     -v expression_attribute_values -- Path to JSON file containing the
expression attribute values.
#     [-p projection_expression] -- Optional projection expression.
#
# Returns:
#     The items as json output.
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_scan() {
    local table_name filter_expression expression_attribute_names
expression_attribute_values projection_expression response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_scan"
        echo "Scan a DynamoDB table."
        echo " -n table_name -- The name of the table."
        echo " -f filter_expression -- The filter expression."
        echo " -a expression_attribute_names -- Path to JSON file containing the
expression attribute names."
        echo " -v expression_attribute_values -- Path to JSON file containing the
expression attribute values."
        echo " [-p projection_expression] -- Optional projection expression."
        echo ""
    }
}

```

```
while getopts "n:f:a:v:p:h" option; do
  case "${option}" in
    n) table_name="${OPTARG}" ;;
    f) filter_expression="${OPTARG}" ;;
    a) expression_attribute_names="${OPTARG}" ;;
    v) expression_attribute_values="${OPTARG}" ;;
    p) projection_expression="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
  errecho "ERROR: You must provide a table name with the -n parameter."
  usage
  return 1
fi

if [[ -z "$filter_expression" ]]; then
  errecho "ERROR: You must provide a filter expression with the -f parameter."
  usage
  return 1
fi

if [[ -z "$expression_attribute_names" ]]; then
  errecho "ERROR: You must provide expression attribute names with the -a
parameter."
  usage
  return 1
fi

if [[ -z "$expression_attribute_values" ]]; then
  errecho "ERROR: You must provide expression attribute values with the -v
parameter."
  usage
```

```

    return 1
fi

if [[ -z "$projection_expression" ]]; then
    response=$(aws dynamodb scan \
        --table-name "$table_name" \
        --filter-expression "$filter_expression" \
        --expression-attribute-names file://"${expression_attribute_names}" \
        --expression-attribute-values file://"${expression_attribute_values}")
else
    response=$(aws dynamodb scan \
        --table-name "$table_name" \
        --filter-expression "$filter_expression" \
        --expression-attribute-names file://"${expression_attribute_names}" \
        --expression-attribute-values file://"${expression_attribute_values}" \
        --projection-expression "$projection_expression")
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports scan operation failed.$response"
    return 1
fi

echo "$response"

return 0
}

#####
# function dynamodb_delete_item
#
# This function deletes an item from a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -k keys -- Path to json file containing the keys that identify the item to
# delete.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.

```



```
#####
function dynamodb_delete_item() {
    local table_name keys response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_delete_item"
        echo "Delete an item from a DynamoDB table."
        echo " -n table_name -- The name of the table."
        echo " -k keys -- Path to json file containing the keys that identify the item
to delete."
        echo ""
    }
    while getopt "n:k:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            k) keys="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$table_name" ]]; then
        errecho "ERROR: You must provide a table name with the -n parameter."
        usage
        return 1
    fi

    if [[ -z "$keys" ]]; then
        errecho "ERROR: You must provide a keys json file path the -k parameter."
        usage
        return 1
    fi
}
#####
```

```

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  keys:  $keys"
iecho ""

response=$(aws dynamodb delete-item \
  --table-name "$table_name" \
  --key file://" $keys")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports delete-item operation failed.$response"
  return 1
fi

return 0
}

#####
# function dynamodb_delete_table
#
# This function deletes a DynamoDB table.
#
# Parameters:
#   -n table_name  -- The name of the table to delete.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
function dynamodb_delete_table() {
  local table_name response
  local option OPTARG # Required to use getopt command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function dynamodb_delete_table"
    echo "Deletes an Amazon DynamoDB table."
    echo " -n table_name  -- The name of the table to delete."
    echo ""
  }

```

```
}

# Retrieve the calling parameters.
while getopts "n:h" option; do
  case "${option}" in
    n) table_name="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
  errecho "ERROR: You must provide a table name with the -n parameter."
  usage
  return 1
fi

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho ""

response=$(aws dynamodb delete-table \
  --table-name "$table_name")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports delete-table operation failed.$response"
  return 1
fi

return 0
}
```

## Las funciones de utilidad usadas en este escenario.

```
#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
```

```
    errecho " Command line failed to parse."
elif [ "$err_code" == 130 ]; then
    errecho " Process received SIGINT."
elif [ "$err_code" == 252 ]; then
    errecho " Command syntax invalid."
elif [ "$err_code" == 253 ]; then
    errecho " The system environment or configuration was invalid."
elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}
```

- Para API obtener más información, consulte los siguientes temas en AWS CLI Command Reference.
  - [BatchWriteItem](#)
  - [CreateTable](#)
  - [DeleteItem](#)
  - [DeleteTable](#)
  - [DescribeTable](#)
  - [GetItem](#)
  - [PutItem](#)
  - [Query](#)
  - [Scan](#)
  - [UpdateItem](#)

## Acciones

### BatchGetItem

En el siguiente ejemplo de código, se muestra cómo usar BatchGetItem.

## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function dynamodb_batch_get_item
#
# This function gets a batch of items from a DynamoDB table.
#
# Parameters:
#     -i item -- Path to json file containing the keys of the items to get.
#
# Returns:
#     The items as json output.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_batch_get_item() {
    local item response
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_batch_get_item"
        echo "Get a batch of items from a DynamoDB table."
        echo " -i item -- Path to json file containing the keys of the items to get."
        echo ""
    }

    while getopt "i:h" option; do
        case "${option}" in
            i) item="${OPTARG}" ;;
            h)
                usage
                return 0
        esac
    done
}
```

```

        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

if [[ -z "$item" ]]; then
    errecho "ERROR: You must provide an item with the -i parameter."
    usage
    return 1
fi

response=$(aws dynamodb batch-get-item \
    --request-items file://"${item}")
local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports batch-get-item operation failed.$response"
    return 1
fi

echo "$response"

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

```

```
#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Para API obtener más información, consulte [BatchGetItem](#) la Referencia de AWS CLI comandos.



## BatchWriteItem

En el siguiente ejemplo de código, se muestra cómo usar BatchWriteItem.

### AWS CLI con el script Bash

#### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function dynamodb_batch_write_item
#
# This function writes a batch of items into a DynamoDB table.
#
# Parameters:
#     -i item -- Path to json file containing the items to write.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_batch_write_item() {
    local item response
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_batch_write_item"
        echo "Write a batch of items into a DynamoDB table."
        echo " -i item -- Path to json file containing the items to write."
        echo ""
    }
    while getopt "i:h" option; do
        case "${option}" in
            i) item="${OPTARG}" ;;
            h)
                usage
                return 0
        esac
    done
}
```

```

        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

if [[ -z "$item" ]]; then
    errecho "ERROR: You must provide an item with the -i parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    table_name:  $table_name"
iecho "    item:        $item"
iecho ""

response=$(aws dynamodb batch-write-item \
    --request-items file://"$item")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports batch-write-item operation failed.$response"
    return 1
fi

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.

```

```
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    fi
}

```

```

elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Para API obtener más información, consulte [BatchWriteItem](#) la Referencia de AWS CLI comandos.

## CreateTable

En el siguiente ejemplo de código, se muestra cómo usar CreateTable.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function dynamodb_create_table
#
# This function creates an Amazon DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table to create.
#     -a attribute_definitions -- JSON file path of a list of attributes and their
types.
#     -k key_schema -- JSON file path of a list of attributes and their key types.
#     -p provisioned_throughput -- Provisioned throughput settings for the table.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####

```

```

function dynamodb_create_table() {
  local table_name attribute_definitions key_schema provisioned_throughput response
  local option OPTARG # Required to use getopt command in a function.

  #####
  # Function usage explanation
  #####
  function usage() {
    echo "function dynamodb_create_table"
    echo "Creates an Amazon DynamoDB table."
    echo " -n table_name -- The name of the table to create."
    echo " -a attribute_definitions -- JSON file path of a list of attributes and
their types."
    echo " -k key_schema -- JSON file path of a list of attributes and their key
types."
    echo " -p provisioned_throughput -- Provisioned throughput settings for the
table."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopt "n:a:k:p:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      a) attribute_definitions="${OPTARG}" ;;
      k) key_schema="${OPTARG}" ;;
      p) provisioned_throughput="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
  fi
}

```

```
fi

if [[ -z "$attribute_definitions" ]]; then
    errecho "ERROR: You must provide an attribute definitions json file path the -a
parameter."
    usage
    return 1
fi

if [[ -z "$key_schema" ]]; then
    errecho "ERROR: You must provide a key schema json file path the -k parameter."
    usage
    return 1
fi

if [[ -z "$provisioned_throughput" ]]; then
    errecho "ERROR: You must provide a provisioned throughput json file path the -p
parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  attribute_definitions:  $attribute_definitions"
iecho "  key_schema:  $key_schema"
iecho "  provisioned_throughput:  $provisioned_throughput"
iecho ""

response=$(aws dynamodb create-table \
  --table-name "$table_name" \
  --attribute-definitions file://"${attribute_definitions}" \
  --key-schema file://"${key_schema}" \
  --provisioned-throughput "$provisioned_throughput")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-table operation failed.$response"
    return 1
fi

return 0
```

```
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```
#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1

```

```

errecho "Error code : $err_code"
if [ "$err_code" == 1 ]; then
    errecho " One or more S3 transfers failed."
elif [ "$err_code" == 2 ]; then
    errecho " Command line failed to parse."
elif [ "$err_code" == 130 ]; then
    errecho " Process received SIGINT."
elif [ "$err_code" == 252 ]; then
    errecho " Command syntax invalid."
elif [ "$err_code" == 253 ]; then
    errecho " The system environment or configuration was invalid."
elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Para API obtener más información, consulte [CreateTable](#) la Referencia de AWS CLI comandos.

## DeleteItem

En el siguiente ejemplo de código, se muestra cómo usar DeleteItem.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function dynamodb_delete_item
#
# This function deletes an item from a DynamoDB table.
#
# Parameters:
#     -n table_name  -- The name of the table.

```



```

#       -k keys  -- Path to json file containing the keys that identify the item to
delete.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
#####
function dynamodb_delete_item() {
    local table_name keys response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_delete_item"
        echo "Delete an item from a DynamoDB table."
        echo " -n table_name  -- The name of the table."
        echo " -k keys  -- Path to json file containing the keys that identify the item
to delete."
        echo ""
    }
    while getopt "n:k:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            k) keys="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$table_name" ]]; then
        errecho "ERROR: You must provide a table name with the -n parameter."
        usage
        return 1
    fi
}

```

```

if [[ -z "$keys" ]]; then
    errecho "ERROR: You must provide a keys json file path the -k parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  keys:        $keys"
iecho ""

response=$(aws dynamodb delete-item \
  --table-name "$table_name" \
  --key file://"${keys}")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-item operation failed.$response"
    return 1
fi

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

```

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}
```

```
}

```

- Para API obtener más información, consulte [DeleteItem](#) la Referencia de AWS CLI comandos.

## DeleteTable

En el siguiente ejemplo de código, se muestra cómo usar DeleteTable.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function dynamodb_delete_table
#
# This function deletes a DynamoDB table.
#
# Parameters:
#     -n table_name  -- The name of the table to delete.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_delete_table() {
    local table_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function dynamodb_delete_table"
        echo "Deletes an Amazon DynamoDB table."
        echo " -n table_name  -- The name of the table to delete."
        echo ""
    }
}

```

```
# Retrieve the calling parameters.
while getopts "n:h" option; do
  case "${option}" in
    n) table_name="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
  errecho "ERROR: You must provide a table name with the -n parameter."
  usage
  return 1
fi

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho ""

response=$(aws dynamodb delete-table \
  --table-name "$table_name")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports delete-table operation failed.$response"
  return 1
fi

return 0
}
```

Las funciones de utilidad utilizadas en este ejemplo.

```
#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    fi
}
```

```

elif [ "$err_code" == 130 ]; then
    errecho " Process received SIGINT."
elif [ "$err_code" == 252 ]; then
    errecho " Command syntax invalid."
elif [ "$err_code" == 253 ]; then
    errecho " The system environment or configuration was invalid."
elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Para API obtener más información, consulte [DeleteTable](#) la Referencia de AWS CLI comandos.

## DescribeTable

En el siguiente ejemplo de código, se muestra cómo usar DescribeTable.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function dynamodb_describe_table
#
# This function returns the status of a DynamoDB table.
#
# Parameters:
#     -n table_name  -- The name of the table.
#
# Response:
#     - TableStatus:
#     And:
#     0 - Table is active.

```

```

#      1 - If it fails.
#####
function dynamodb_describe_table {
    local table_name
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_describe_table"
        echo "Describe the status of a DynamoDB table."
        echo "  -n table_name  -- The name of the table."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$table_name" ]]; then
        errecho "ERROR: You must provide a table name with the -n parameter."
        usage
        return 1
    fi

    local table_status
    table_status=$(
        aws dynamodb describe-table \
            --table-name "$table_name" \
            --output text \

```



```

        --query 'Table.TableStatus'
    )

    local error_code=${?}

    if [[ $error_code -ne 0 ]]; then
        aws_cli_error_log "$error_code"
        errecho "ERROR: AWS reports describe-table operation failed.$table_status"
        return 1
    fi

    echo "$table_status"

    return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####

```

```
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho " One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho " Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho " Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho " Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho " The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
  fi

  return 0
}
```

- Para API obtener más información, consulte [DescribeTable](#) la Referencia de AWS CLI comandos.

## GetItem

En el siguiente ejemplo de código, se muestra cómo usar GetItem.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function dynamodb_get_item
#
```

```

# This function gets an item from a DynamoDB table.
#
# Parameters:
#   -n table_name  -- The name of the table.
#   -k keys        -- Path to json file containing the keys that identify the item to
get.
#   [-q query]    -- Optional JMESPath query expression.
#
# Returns:
#   The item as text output.
# And:
#   0 - If successful.
#   1 - If it fails.
#####
function dynamodb_get_item() {
    local table_name keys query response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    # #####
    function usage() {
        echo "function dynamodb_get_item"
        echo "Get an item from a DynamoDB table."
        echo " -n table_name  -- The name of the table."
        echo " -k keys        -- Path to json file containing the keys that identify the item
to get."
        echo " [-q query]  -- Optional JMESPath query expression."
        echo ""
    }
    query=""
    while getopt "n:k:q:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            k) keys="${OPTARG}" ;;
            q) query="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1

```

```
        ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$keys" ]]; then
    errecho "ERROR: You must provide a keys json file path the -k parameter."
    usage
    return 1
fi

if [[ -n "$query" ]]; then
    response=$(aws dynamodb get-item \
        --table-name "$table_name" \
        --key file://"${keys}" \
        --output text \
        --query "$query")
else
    response=$(
        aws dynamodb get-item \
            --table-name "$table_name" \
            --key file://"${keys}" \
            --output text
    )
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports get-item operation failed.$response"
    return 1
fi

if [[ -n "$query" ]]; then
    echo "$response" | sed "/^\t/s/\t//1" # Remove initial tab that the JMSEPath
    query inserts on some strings.
else
```

```

    echo "$response"
  fi

  return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho " One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho " Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho " Process received SIGINT."
  elif [ "$err_code" == 252 ]; then

```

```

    errecho " Command syntax invalid."
elif [ "$err_code" == 253 ]; then
    errecho " The system environment or configuration was invalid."
elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Para API obtener más información, consulte [GetItem](#) la Referencia de AWS CLI comandos.

## ListTables

En el siguiente ejemplo de código, se muestra cómo usar ListTables.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function dynamodb_list_tables
#
# This function lists all the tables in a DynamoDB.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_list_tables() {
    response=$(aws dynamodb list-tables \
        --output text \
        --query "TableNames")

    local error_code=${?}

```

```

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports batch-write-item operation failed.$response"
    return 1
fi

echo "$response" | tr -s "[:space:]" "\n"

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then

```

```

    errecho " One or more S3 transfers failed."
elif [ "$err_code" == 2 ]; then
    errecho " Command line failed to parse."
elif [ "$err_code" == 130 ]; then
    errecho " Process received SIGINT."
elif [ "$err_code" == 252 ]; then
    errecho " Command syntax invalid."
elif [ "$err_code" == 253 ]; then
    errecho " The system environment or configuration was invalid."
elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Para API obtener más información, consulte [ListTables](#) la Referencia de AWS CLI comandos.

## PutItem

En el siguiente ejemplo de código, se muestra cómo usar PutItem.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function dynamodb_put_item
#
# This function puts an item into a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -i item      -- Path to json file containing the item values.
#

```



```

# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_put_item() {
    local table_name item response
    local option OPTARG # Required to use getopt command in a function.

#####
# Function usage explanation
#####
function usage() {
    echo "function dynamodb_put_item"
    echo "Put an item into a DynamoDB table."
    echo " -n table_name -- The name of the table."
    echo " -i item -- Path to json file containing the item values."
    echo ""
}

while getopt "n:i:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        i) item="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$item" ]]; then
    errecho "ERROR: You must provide an item with the -i parameter."

```

```

    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    table_name:  $table_name"
iecho "    item:    $item"
iecho ""
iecho ""

response=$(aws dynamodb put-item \
    --table-name "$table_name" \
    --item file://" $item")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports put-item operation failed.$response"
    return 1
fi

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho

```

```
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}
```

- Para API obtener más información, consulte [PutItem](#) la Referencia de AWS CLI comandos.

## Query

En el siguiente ejemplo de código, se muestra cómo usar Query.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function dynamodb_query
#
# This function queries a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -k key_condition_expression -- The key condition expression.
#     -a attribute_names -- Path to JSON file containing the attribute names.
#     -v attribute_values -- Path to JSON file containing the attribute values.
#     [-p projection_expression] -- Optional projection expression.
#
# Returns:
#     The items as json output.
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_query() {
    local table_name key_condition_expression attribute_names attribute_values
    projection_expression response
    local option OPTARG # Required to use getopt command in a function.

    # #####
    # Function usage explanation
    #####
}
```

```
function usage() {
    echo "function dynamodb_query"
    echo "Query a DynamoDB table."
    echo " -n table_name -- The name of the table."
    echo " -k key_condition_expression -- The key condition expression."
    echo " -a attribute_names -- Path to JSON file containing the attribute names."
    echo " -v attribute_values -- Path to JSON file containing the attribute
values."
    echo " [-p projection_expression] -- Optional projection expression."
    echo ""
}

while getopts "n:k:a:v:p:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        k) key_condition_expression="${OPTARG}" ;;
        a) attribute_names="${OPTARG}" ;;
        v) attribute_values="${OPTARG}" ;;
        p) projection_expression="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$key_condition_expression" ]]; then
    errecho "ERROR: You must provide a key condition expression with the -k
parameter."
    usage
    return 1
fi
```

```
if [[ -z "$attribute_names" ]]; then
    errecho "ERROR: You must provide a attribute names with the -a parameter."
    usage
    return 1
fi

if [[ -z "$attribute_values" ]]; then
    errecho "ERROR: You must provide a attribute values with the -v parameter."
    usage
    return 1
fi

if [[ -z "$projection_expression" ]]; then
    response=$(aws dynamodb query \
        --table-name "$table_name" \
        --key-condition-expression "$key_condition_expression" \
        --expression-attribute-names file://"${attribute_names}" \
        --expression-attribute-values file://"${attribute_values}")
else
    response=$(aws dynamodb query \
        --table-name "$table_name" \
        --key-condition-expression "$key_condition_expression" \
        --expression-attribute-names file://"${attribute_names}" \
        --expression-attribute-values file://"${attribute_values}" \
        --projection-expression "$projection_expression")
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports query operation failed.$response"
    return 1
fi

echo "$response"

return 0
}
```

Las funciones de utilidad utilizadas en este ejemplo.

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi
}
```

```

    return 0
}

```

- Para API obtener más información, consulte [Query](#) in AWS CLI Command Reference.

## Scan

En el siguiente ejemplo de código, se muestra cómo usar Scan.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function dynamodb_scan
#
# This function scans a DynamoDB table.
#
# Parameters:
#     -n table_name -- The name of the table.
#     -f filter_expression -- The filter expression.
#     -a expression_attribute_names -- Path to JSON file containing the expression
#     attribute names.
#     -v expression_attribute_values -- Path to JSON file containing the
#     expression attribute values.
#     [-p projection_expression] -- Optional projection expression.
#
# Returns:
#     The items as json output.
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function dynamodb_scan() {
    local table_name filter_expression expression_attribute_names
    expression_attribute_values projection_expression response
    local option OPTARG # Required to use getopt command in a function.

```



```
# #####
# Function usage explanation
# #####
function usage() {
    echo "function dynamodb_scan"
    echo "Scan a DynamoDB table."
    echo " -n table_name -- The name of the table."
    echo " -f filter_expression -- The filter expression."
    echo " -a expression_attribute_names -- Path to JSON file containing the
expression attribute names."
    echo " -v expression_attribute_values -- Path to JSON file containing the
expression attribute values."
    echo " [-p projection_expression] -- Optional projection expression."
    echo ""
}

while getopts "n:f:a:v:p:h" option; do
    case "${option}" in
        n) table_name="${OPTARG}" ;;
        f) filter_expression="${OPTARG}" ;;
        a) expression_attribute_names="${OPTARG}" ;;
        v) expression_attribute_values="${OPTARG}" ;;
        p) projection_expression="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$filter_expression" ]]; then
```

```
errecho "ERROR: You must provide a filter expression with the -f parameter."
usage
return 1
fi

if [[ -z "$expression_attribute_names" ]]; then
    errecho "ERROR: You must provide expression attribute names with the -a
parameter."
    usage
    return 1
fi

if [[ -z "$expression_attribute_values" ]]; then
    errecho "ERROR: You must provide expression attribute values with the -v
parameter."
    usage
    return 1
fi

if [[ -z "$projection_expression" ]]; then
    response=$(aws dynamodb scan \
        --table-name "$table_name" \
        --filter-expression "$filter_expression" \
        --expression-attribute-names file://"${expression_attribute_names}" \
        --expression-attribute-values file://"${expression_attribute_values}")
else
    response=$(aws dynamodb scan \
        --table-name "$table_name" \
        --filter-expression "$filter_expression" \
        --expression-attribute-names file://"${expression_attribute_names}" \
        --expression-attribute-values file://"${expression_attribute_values}" \
        --projection-expression "$projection_expression")
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports scan operation failed.$response"
    return 1
fi

echo "$response"
```

```

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    }
}

```

```

elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Para API obtener más información, consulte la referencia de [Scan](#) in AWS CLI Command.

## UpdateItem

En el siguiente ejemplo de código, se muestra cómo usar UpdateItem.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function dynamodb_update_item
#
# This function updates an item in a DynamoDB table.
#
#
# Parameters:
#     -n table_name -- The name of the table.
#     -k keys      -- Path to json file containing the keys that identify the item to
update.
#     -e update expression -- An expression that defines one or more attributes
to be updated.
#     -v values    -- Path to json file containing the update values.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####

```

```

function dynamodb_update_item() {
    local table_name keys update_expression values response
    local option OPTARG # Required to use getopt command in a function.

    #####
    # Function usage explanation
    #####
    function usage() {
        echo "function dynamodb_update_item"
        echo "Update an item in a DynamoDB table."
        echo " -n table_name  -- The name of the table."
        echo " -k keys      -- Path to json file containing the keys that identify the item
to update."
        echo " -e update expression  -- An expression that defines one or more
attributes to be updated."
        echo " -v values    -- Path to json file containing the update values."
        echo ""
    }

    while getopt "n:k:e:v:h" option; do
        case "${option}" in
            n) table_name="${OPTARG}" ;;
            k) keys="${OPTARG}" ;;
            e) update_expression="${OPTARG}" ;;
            v) values="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$table_name" ]]; then
        errecho "ERROR: You must provide a table name with the -n parameter."
        usage
        return 1
    fi
}

```

```
if [[ -z "$keys" ]]; then
    errecho "ERROR: You must provide a keys json file path the -k parameter."
    usage
    return 1
fi
if [[ -z "$update_expression" ]]; then
    errecho "ERROR: You must provide an update expression with the -e parameter."
    usage
    return 1
fi

if [[ -z "$values" ]]; then
    errecho "ERROR: You must provide a values json file path the -v parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "  table_name:  $table_name"
iecho "  keys:        $keys"
iecho "  update_expression:  $update_expression"
iecho "  values:      $values"

response=$(aws dynamodb update-item \
  --table-name "$table_name" \
  --key file://" $keys" \
  --update-expression "$update_expression" \
  --expression-attribute-values file://" $values")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports update-item operation failed.$response"
    return 1
fi

return 0
}
```

Las funciones de utilidad utilizadas en este ejemplo.

```
#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-return-codes.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    fi
}
```

```
elif [ "$err_code" == 130 ]; then
    errecho " Process received SIGINT."
elif [ "$err_code" == 252 ]; then
    errecho " Command syntax invalid."
elif [ "$err_code" == 253 ]; then
    errecho " The system environment or configuration was invalid."
elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}
```

- Para API obtener más información, consulte [UpdateItem](#) la Referencia de AWS CLI comandos.

## EC2 Ejemplos de Amazon que utilizan AWS CLI el script Bash

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el script AWS Command Line Interface with Bash with AmazonEC2.

Los conceptos básicos son ejemplos de código que muestran cómo realizar las operaciones esenciales dentro de un servicio.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Conceptos básicos](#)
- [Acciones](#)



## Conceptos básicos

### Aprenda los conceptos básicos

En el siguiente ejemplo de código, se muestra cómo:

- Cree un par de claves y un grupo de seguridad.
- Seleccione una Amazon Machine Image (AMI) y un tipo de instancia compatible y, a continuación, cree una instancia.
- Detenga y vuelva a iniciar la instancia.
- Asocie una dirección IP elástica a su instancia.
- Conéctate a tu instancia con los recursos SSH y, a continuación, límpialos.

### AWS CLI con el script Bash

#### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

Ejecutar un escenario interactivo en un símbolo del sistema.

```
#####  
# function get_started_with_ec2_instances  
#  
# Runs an interactive scenario that shows how to get started using EC2 instances.  
#  
# "EC2 access" permissions are needed to run this code.  
#  
# Returns:  
# 0 - If successful.  
# 1 - If an error occurred.  
#####  
function get_started_with_ec2_instances() {  
    # Requires version 4 for mapfile.  
    local required_version=4.0  
  
    # Get the current Bash version  
    # Check if BASH_VERSION is set
```

```
local current_version
if [[ -n "$BASH_VERSION" ]]; then
    # Convert BASH_VERSION to a number for comparison
    current_version=$BASH_VERSION
else
    # Get the current Bash version using the bash command
    current_version=$(bash --version | head -n 1 | awk '{ print $4 }')
fi

# Convert version strings to numbers for comparison
local required_version_num current_version_num
required_version_num=$(echo "$required_version" | awk -F. '{ print ($1 * 10000) + ($2 * 100) + $3 }')
current_version_num=$(echo "$current_version" | awk -F. '{ print ($1 * 10000) + ($2 * 100) + $3 }')

# Compare versions
if ((current_version_num < required_version_num)); then
    echo "Error: This script requires Bash version $required_version or higher."
    echo "Your current Bash version is number is $current_version."
    exit 1
fi

{
    if [ "$EC2_OPERATIONS_SOURCED" != "True" ]; then

        source ./ec2_operations.sh
    fi
}

echo_repeat "*" 88
echo "Welcome to the Amazon Elastic Compute Cloud (Amazon EC2) get started with instances demo."
echo_repeat "*" 88
echo

echo "Let's create an RSA key pair that you can be use to securely connect to "
echo "your EC2 instance."

echo -n "Enter a unique name for your key: "
get_input
local key_name
key_name=$get_input_result
```

```
local temp_dir
temp_dir=$(mktemp -d)
local key_file_name="$temp_dir/${key_name}.pem"

if ec2_create_keypair -n "${key_name}" -f "${key_file_name}"; then
    echo "Created a key pair $key_name and saved the private key to $key_file_name"
    echo
else
    errecho "The key pair failed to create. This demo will exit."
    return 1
fi

chmod 400 "${key_file_name}"

if yes_no_input "Do you want to list some of your key pairs? (y/n) "; then
    local keys_and_fingerprints
    keys_and_fingerprints="$(ec2_describe_key_pairs)" && {
        local image_name_and_id
        while IFS=$'\n' read -r image_name_and_id; do
            local entries
            IFS=$'\t' read -ra entries <<<"$image_name_and_id"
            echo "Found rsa key ${entries[0]} with fingerprint:"
            echo "    ${entries[1]}"
        done <<<"$keys_and_fingerprints"
    }
fi

echo_repeat "*" 88
echo_repeat "*" 88

echo "Let's create a security group to manage access to your instance."
echo -n "Enter a unique name for your security group: "
get_input
local security_group_name
security_group_name=$get_input_result
local security_group_id
security_group_id=$(ec2_create_security_group -n "$security_group_name" \
-d "Security group for EC2 instance") || {
    errecho "The security failed to create. This demo will exit."
    clean_up "$key_name" "$key_file_name"
    return 1
}
```

```
echo "Security group created with ID $security_group_id"
echo

local public_ip
public_ip=$(curl -s http://checkip.amazonaws.com)

echo "Let's add a rule to allow SSH only from your current IP address."
echo "Your public IP address is $public_ip"
echo -n "press return to add this rule to your security group."
get_input

if ! ec2_authorize_security_group_ingress -g "$security_group_id" -i "$public_ip"
-p tcp -f 22 -t 22; then
    errecho "The security group rules failed to update. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1
fi

echo "Security group rules updated"

local security_group_description
security_group_description="$(ec2_describe_security_groups -g
"${security_group_id}")" || {
    errecho "Failed to describe security groups. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1
}

mapfile -t parameters <<<"$security_group_description"
IFS=$'\t' read -ra entries <<<"${parameters[0]}"
echo "Security group: ${entries[0]}"
echo "    ID: ${entries[1]}"
echo "    VPC: ${entries[2]}"
echo "Inbound permissions:"
IFS=$'\t' read -ra entries <<<"${parameters[1]}"
echo "    IpProtocol: ${entries[0]}"
echo "    FromPort: ${entries[1]}"
echo "    ToPort: ${entries[2]}"
echo "    CidrIp: ${parameters[2]}"

local parameters
parameters="$(ssm_get_parameters_by_path -p "/aws/service/ami-amazon-linux-
latest")" || {
    errecho "Failed to get parameters. This demo will exit."
```

```
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1
}

local image_ids=""
mapfile -t parameters <<<"$parameters"
for image_name_and_id in "${parameters[@]"; do
    IFS=$'\t' read -ra values <<<"$image_name_and_id"
    if [[ "${values[0]}" == *"amzn2"* ]]; then
        image_ids+="${values[1]} "
    fi
done

local images
images="$(ec2_describe_images -i "$image_ids")" || {
    errecho "Failed to describe images. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1
}

new_line_and_tab_to_list "$images"
local images=("${list_result[@]}")

# Get the size of the array
local images_count=${#images[@]}

if ((images_count == 0)); then
    errecho "No images found. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1
fi

echo_repeat "*" 88
echo_repeat "*" 88

echo "Let's create an instance from an Amazon Linux 2 AMI. Here are some options:"
for ((i = 0; i < images_count; i += 3)); do
    echo "$(((i / 3) + 1)) - ${images[$i]}"
done

integer_input "Please enter the number of the AMI you want to use: " 1
"$((images_count / 3))"
```

```

local choice=$get_input_result
choice=$((choice - 1) * 3))

echo "Great choice."
echo

local architecture=${images[$((choice + 1))]}
local image_id=${images[$((choice + 2))]}
echo "Here are some instance types that support the ${architecture} architecture
of the image:"
response="$(ec2_describe_instance_types -a "${architecture}" -t
"*micro,*small")" || {
    errecho "Failed to describe instance types. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1
}

local instance_types
mapfile -t instance_types <<<"$response"

# Get the size of the array
local instance_types_count=${#instance_types[@]}

echo "Here are some options:"
for ((i = 0; i < instance_types_count; i++)); do
    echo "$((i + 1)) - ${instance_types[$i]}"
done

integer_input "Which one do you want to use? " 1 "${#instance_types[@]}"
"
choice=$get_input_result
local instance_type=${instance_types[$((choice - 1))]}
echo "Another great choice."
echo

echo "Creating your instance and waiting for it to start..."
local instance_id
instance_id=$(ec2_run_instances -i "$image_id" -t "$instance_type" -k "$key_name"
-s "$security_group_id") || {
    errecho "Failed to run instance. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1
}

```

```
ec2_wait_for_instance_running -i "${instance_id}"
echo "Your instance is ready:"
echo

local instance_details
instance_details="$(ec2_describe_instances -i "${instance_id}")"

echo
print_instance_details "${instance_details}"

local public_ip
public_ip=$(echo "${instance_details}" | awk '{print $6}')
echo
echo "You can use SSH to connect to your instance"
echo "If the connection attempt times out, you might have to manually update the
SSH ingress rule"
echo "for your IP address in the AWS Management Console."
connect_to_instance "$key_file_name" "$public_ip"

echo -n "Press Enter when you're ready to continue the demo: "
get_input

echo_repeat "*" 88
echo_repeat "*" 88

echo "Let's stop and start your instance to see what changes."
echo "Stopping your instance and waiting until it's stopped..."
ec2_stop_instances -i "${instance_id}"
ec2_wait_for_instance_stopped -i "${instance_id}"

echo "Your instance is stopped. Restarting..."

ec2_start_instances -i "${instance_id}"
ec2_wait_for_instance_running -i "${instance_id}"

echo "Your instance is running again."
local instance_details
instance_details="$(ec2_describe_instances -i "${instance_id}")"

print_instance_details "${instance_details}"

public_ip=$(echo "${instance_details}" | awk '{print $6}')

echo "Every time your instance is restarted, its public IP address changes"
```

```
connect_to_instance "$key_file_name" "$public_ip"

echo -n "Press Enter when you're ready to continue the demo: "
get_input

echo_repeat "*" 88
echo_repeat "*" 88

echo "You can allocate an Elastic IP address and associate it with your instance"
echo "to keep a consistent IP address even when your instance restarts."

local result
result=$(ec2_allocate_address -d vpc) || {
    errecho "Failed to allocate an address. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id" "$instance_id"
    return 1
}

local elastic_ip allocation_id
elastic_ip=$(echo "$result" | awk '{print $1}')
allocation_id=$(echo "$result" | awk '{print $2}')

echo "Allocated static Elastic IP address: $elastic_ip"

local association_id
association_id=$(ec2_associate_address -i "$instance_id" -a "$allocation_id") || {
    errecho "Failed to associate an address. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id" "$instance_id"
"$allocation_id"
    return 1
}

echo "Associated your Elastic IP with your instance."
echo "You can now use SSH to connect to your instance by using the Elastic IP."
connect_to_instance "$key_file_name" "$elastic_ip"

echo -n "Press Enter when you're ready to continue the demo: "
get_input

echo_repeat "*" 88
echo_repeat "*" 88

echo "Let's stop and start your instance to see what changes."
echo "Stopping your instance and waiting until it's stopped..."
```



```
ec2_stop_instances -i "$instance_id"
ec2_wait_for_instance_stopped -i "$instance_id"

echo "Your instance is stopped. Restarting..."

ec2_start_instances -i "$instance_id"
ec2_wait_for_instance_running -i "$instance_id"

echo "Your instance is running again."
local instance_details
instance_details="$(ec2_describe_instances -i "${instance_id}")"

print_instance_details "${instance_details}"

echo "Because you have associated an Elastic IP with your instance, you can"
echo "connect by using a consistent IP address after the instance restarts."
connect_to_instance "$key_file_name" "$elastic_ip"

echo -n "Press Enter when you're ready to continue the demo: "
get_input

echo_repeat "*" 88
echo_repeat "*" 88

if yes_no_input "Do you want to delete the resources created in this demo: (y/n)
"; then
    clean_up "$key_name" "$key_file_name" "$security_group_id" "$instance_id" \
        "$allocation_id" "$association_id"
else
    echo "The following resources were not deleted."
    echo "Key pair: $key_name"
    echo "Key file: $key_file_name"
    echo "Security group: $security_group_id"
    echo "Instance: $instance_id"
    echo "Elastic IP address: $elastic_ip"
fi
}

#####
# function clean_up
#
# This function cleans up the created resources.
# $1 - The name of the ec2 key pair to delete.
# $2 - The name of the key file to delete.
```

```
# $3 - The ID of the security group to delete.
# $4 - The ID of the instance to terminate.
# $5 - The ID of the elastic IP address to release.
# $6 - The ID of the elastic IP address to disassociate.
#
# Returns:
# 0 - If successful.
# 1 - If an error occurred.
#####
function clean_up() {
    local result=0
    local key_pair_name=$1
    local key_file_name=$2
    local security_group_id=$3
    local instance_id=$4
    local allocation_id=$5
    local association_id=$6

    if [ -n "$association_id" ]; then
        # bashsupport disable=BP2002
        if (ec2_disassociate_address -a "$association_id"); then
            echo "Disassociated elastic IP address with ID $association_id"
        else
            errecho "The elastic IP address disassociation failed."
            result=1
        fi
    fi

    if [ -n "$allocation_id" ]; then
        # bashsupport disable=BP2002
        if (ec2_release_address -a "$allocation_id"); then
            echo "Released elastic IP address with ID $allocation_id"
        else
            errecho "The elastic IP address release failed."
            result=1
        fi
    fi

    if [ -n "$instance_id" ]; then
        # bashsupport disable=BP2002
        if (ec2_terminate_instances -i "$instance_id"); then
            echo "Started terminating instance with ID $instance_id"

            ec2_wait_for_instance_terminated -i "$instance_id"
        fi
    fi
}
```

```
    else
        errecho "The instance terminate failed."
        result=1
    fi
fi

if [ -n "$security_group_id" ]; then
    # bashsupport disable=BP2002
    if (ec2_delete_security_group -i "$security_group_id"); then
        echo "Deleted security group with ID $security_group_id"
    else
        errecho "The security group delete failed."
        result=1
    fi
fi

if [ -n "$key_pair_name" ]; then
    # bashsupport disable=BP2002
    if (ec2_delete_keypair -n "$key_pair_name"); then
        echo "Deleted key pair named $key_pair_name"
    else
        errecho "The key pair delete failed."
        result=1
    fi
fi

if [ -n "$key_file_name" ]; then
    rm -f "$key_file_name"
fi

return $result
}

#####
# function ssm_get_parameters_by_path
#
# This function retrieves one or more parameters from the AWS Systems Manager
# Parameter Store
# by specifying a parameter path.
#
# Parameters:
#     -p parameter_path - The path of the parameter(s) to retrieve.
#
# And:
```

```

#      0 - If successful.
#      1 - If it fails.
#####
function ssm_get_parameters_by_path() {
    local parameter_path response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ssm_get_parameters_by_path"
        echo "Retrieves one or more parameters from the AWS Systems Manager Parameter
Store by specifying a parameter path."
        echo "  -p parameter_path - The path of the parameter(s) to retrieve."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "p:h" option; do
        case "${option}" in
            p) parameter_path="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$parameter_path" ]]; then
        errecho "ERROR: You must provide a parameter path with the -p parameter."
        usage
        return 1
    fi

    response=$(aws ssm get-parameters-by-path \
        --path "$parameter_path" \
        --query "Parameters[*].[Name, Value]" \
        --output text) || {
        aws_cli_error_log $?
    }
}

```

```

    errecho "ERROR: AWS reports get-parameters-by-path operation failed.$response"
    return 1
}

echo "$response"

return 0
}

#####
# function print_instance_details
#
# This function prints the details of an Amazon Elastic Compute Cloud (Amazon EC2)
instance.
#
# Parameters:
#     instance_details - The instance details in the format "InstanceId ImageId
InstanceType KeyName VpcId PublicIpAddress State.Name".
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function print_instance_details() {
    local instance_details="$1"

    if [[ -z "${instance_details}" ]]; then
        echo "Error: Missing required instance details argument."
        return 1
    fi

    local instance_id image_id instance_type key_name vpc_id public_ip state
    instance_id=$(echo "${instance_details}" | awk '{print $1}')
    image_id=$(echo "${instance_details}" | awk '{print $2}')
    instance_type=$(echo "${instance_details}" | awk '{print $3}')
    key_name=$(echo "${instance_details}" | awk '{print $4}')
    vpc_id=$(echo "${instance_details}" | awk '{print $5}')
    public_ip=$(echo "${instance_details}" | awk '{print $6}')
    state=$(echo "${instance_details}" | awk '{print $7}')

    echo "    ID: ${instance_id}"
    echo "    Image ID: ${image_id}"
    echo "    Instance type: ${instance_type}"
    echo "    Key name: ${key_name}"
}

```

```
echo "    VPC ID: ${vpc_id}"
echo "    Public IP: ${public_ip}"
echo "    State: ${state}"

return 0
}

#####
# function connect_to_instance
#
# This function displays the public IP address of an Amazon Elastic Compute Cloud
# (Amazon EC2) instance and prompts the user to connect to the instance via SSH.
#
# Parameters:
#     $1 - The name of the key file used to connect to the instance.
#     $2 - The public IP address of the instance.
#
# Returns:
#     None
#####
function connect_to_instance() {
    local key_file_name="$1"
    local public_ip="$2"

    # Validate the input parameters
    if [[ -z "$key_file_name" ]]; then
        echo "ERROR: You must provide a key file name as the first argument." >&2
        return 1
    fi

    if [[ -z "$public_ip" ]]; then
        echo "ERROR: You must provide a public IP address as the second argument." >&2
        return 1
    fi

    # Display the public IP address and connection command
    echo "To connect, run the following command:"
    echo "    ssh -i ${key_file_name} ec2-user@${public_ip}"

    # Prompt the user to connect to the instance
    if yes_no_input "Do you want to connect now? (y/n) "; then
        echo "After you have connected, you can return to this example by typing 'exit'"
        ssh -i "${key_file_name}" ec2-user@"${public_ip}"
    fi
}
```

```
}

#####
# function get_input
#
# This function gets user input from the command line.
#
# Outputs:
#   User input to stdout.
#
# Returns:
#   0
#####
function get_input() {

    if [ -z "${mock_input+x}" ]; then
        read -r get_input_result
    else

        if [ "$mock_input_array_index" -lt ${#mock_input_array[@]} ]; then
            get_input_result="${mock_input_array[$mock_input_array_index]}"
            # bashsupport disable=BP2001
            # shellcheck disable=SC2206
            ((mock_input_array_index++))
            echo -n "$get_input_result"
        else
            echo "MOCK_INPUT_ARRAY has no more elements" 1>&2
            return 1
        fi
    fi

    return 0
}

#####
# function yes_no_input
#
# This function requests a yes/no answer from the user, following to a prompt.
#
# Parameters:
#   $1 - The prompt.
#
# Returns:
#   0 - If yes.
```

```

#      1 - If no.
#####
function yes_no_input() {
  if [ -z "$1" ]; then
    echo "Internal error yes_no_input"
    return 1
  fi

  local index=0
  local response="N"
  while [[ $index -lt 10 ]]; do
    index=$((index + 1))
    echo -n "$1"
    if ! get_input; then
      return 1
    fi
    response=$(echo "$get_input_result" | tr '[:upper:]' '[:lower:]')
    if [ "$response" = "y" ] || [ "$response" = "n" ]; then
      break
    else
      echo -e "\nPlease enter or 'y' or 'n'."
    fi
  done

  echo

  if [ "$response" = "y" ]; then
    return 0
  else
    return 1
  fi
}

#####
# function integer_input
#
# This function prompts the user to enter an integer within a specified range
# and validates the input.
#
# Parameters:
#   $1 - The prompt message to display to the user.
#   $2 - The minimum value of the accepted range.
#   $3 - The maximum value of the accepted range.
#

```



```

# Returns:
#     The valid integer input from the user.
#     If the input is invalid or out of range, the function will continue
#     prompting the user until a valid input is provided.
#####
function integer_input() {
    local prompt="$1"
    local min_value="$2"
    local max_value="$3"
    local input=""

    while true; do
        # Display the prompt message and wait for user input
        echo -n "$prompt"

        if ! get_input; then
            return 1
        fi

        input="$get_input_result"

        # Check if the input is a valid integer
        if [[ "$input" =~ ^-?[0-9]+$ ]]; then
            # Check if the input is within the specified range
            if ((input >= min_value && input <= max_value)); then
                return 0
            else
                echo "Error: Input, $input, must be between $min_value and $max_value."
            fi
        else
            echo "Error: Invalid input- $input. Please enter an integer."
        fi
    done
}
#####
# function new_line_and_tab_to_list
#
# This function takes a string input containing newlines and tabs, and
# converts it into a list (array) of elements.
#
# Parameters:
#     $1 - The input string containing newlines and tabs.
#
# Returns:

```

```

#       The resulting list (array) is stored in the global variable
#       'list_result'.
#####
function new_line_and_tab_to_list() {
    local input=$1
    export list_result

    list_result=()
    mapfile -t lines <<<"$input"
    local line
    for line in "${lines[@]"; do
        IFS=$'\t' read -ra parameters <<<"$line"
        list_result+=("${parameters[@]}")
    done
}

#####
# function echo_repeat
#
# This function prints a string 'n' times to stdout.
#
# Parameters:
#     $1 - The string.
#     $2 - Number of times to print the string.
#
# Outputs:
#     String 'n' times to stdout.
#
# Returns:
#     0
#####
function echo_repeat() {
    local end=$2
    for ((i = 0; i < end; i++)); do
        echo -n "$1"
    done
    echo
}

```

Las funciones de DynamoDB utilizadas en este escenario.

```
#####
```

```

# function ec2_create_keypair
#
# This function creates an Amazon Elastic Compute Cloud (Amazon EC2) ED25519 or
# 2048-bit RSA key pair
# and writes it to a file.
#
# Parameters:
#     -n key_pair_name - A key pair name.
#     -f file_path - File to store the key pair.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_create_keypair() {
    local key_pair_name file_path response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_create_keypair"
        echo "Creates an Amazon Elastic Compute Cloud (Amazon EC2) ED25519 or 2048-bit
RSA key pair"
        echo " and writes it to a file."
        echo "  -n key_pair_name - A key pair name."
        echo "  -f file_path - File to store the key pair."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:f:h" option; do
        case "${option}" in
            n) key_pair_name="${OPTARG}" ;;
            f) file_path="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
}

```

```

done
export OPTIND=1

if [[ -z "$key_pair_name" ]]; then
    errecho "ERROR: You must provide a key name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$file_path" ]]; then
    errecho "ERROR: You must provide a file path with the -f parameter."
    usage
    return 1
fi

response=$(aws ec2 create-key-pair \
    --key-name "$key_pair_name" \
    --query 'KeyMaterial' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports create-access-key operation failed.$response"
    return 1
}

if [[ -n "$file_path" ]]; then
    echo "$response" >"$file_path"
fi

return 0
}

#####
# function ec2_describe_key_pairs
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2) key
# pairs.
#
# Parameters:
#     -h - Display help.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####

```

```
function ec2_describe_key_pairs() {
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_describe_key_pairs"
        echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2) key
pairs."
        echo " -h - Display help."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "h" option; do
        case "${option}" in
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    local response

    response=$(aws ec2 describe-key-pairs \
        --query 'KeyPairs[*].[KeyName, KeyFingerprint]' \
        --output text) || {
        aws_cli_error_log ${?}
        errecho "ERROR: AWS reports describe-key-pairs operation failed.$response"
        return 1
    }

    echo "$response"

    return 0
}

#####
```

```

# function ec2_create_security_group
#
# This function creates an Amazon Elastic Compute Cloud (Amazon EC2) security group.
#
# Parameters:
#     -n security_group_name - The name of the security group.
#     -d security_group_description - The description of the security group.
#
# Returns:
#     The ID of the created security group, or an error message if the operation
#     fails.
# And:
#     0 - If successful.
#     1 - If it fails.
#
#####
function ec2_create_security_group() {
    local security_group_name security_group_description response

    # Function to display usage information
    function usage() {
        echo "function ec2_create_security_group"
        echo "Creates an Amazon Elastic Compute Cloud (Amazon EC2) security group."
        echo "  -n security_group_name - The name of the security group."
        echo "  -d security_group_description - The description of the security group."
        echo ""
    }

    # Parse the command-line arguments
    while getopts "n:d:h" option; do
        case "${option}" in
            n) security_group_name="${OPTARG}" ;;
            d) security_group_description="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
}

```

```

export OPTIND=1

# Validate the input parameters
if [[ -z "$security_group_name" ]]; then
    errecho "ERROR: You must provide a security group name with the -n parameter."
    return 1
fi

if [[ -z "$security_group_description" ]]; then
    errecho "ERROR: You must provide a security group description with the -d
parameter."
    return 1
fi

# Create the security group
response=$(aws ec2 create-security-group \
    --group-name "$security_group_name" \
    --description "$security_group_description" \
    --query "GroupId" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports create-security-group operation failed."
    errecho "$response"
    return 1
}

echo "$response"
return 0
}

#####
# function ec2_describe_security_groups
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
# security groups.
#
# Parameters:
#     -g security_group_id - The ID of the security group to describe (optional).
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_describe_security_groups() {

```

```
local security_group_id response
local option OPTARG # Required to use getopt command in a function.

# bashsupport disable=BP5008
function usage() {
    echo "function ec2_describe_security_groups"
    echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2) security
groups."
    echo "  -g security_group_id - The ID of the security group to describe
(optional)."
```



```

    errecho "ERROR: AWS reports describe-security-groups operation failed.$response"
    return 1
fi

echo "$response"

return 0
}

#####
# function ec2_authorize_security_group_ingress
#
# This function authorizes an ingress rule for an Amazon Elastic Compute Cloud
# (Amazon EC2) security group.
#
# Parameters:
#   -g security_group_id - The ID of the security group.
#   -i ip_address - The IP address or CIDR block to authorize.
#   -p protocol - The protocol to authorize (e.g., tcp, udp, icmp).
#   -f from_port - The start of the port range to authorize.
#   -t to_port - The end of the port range to authorize.
#
# And:
#   0 - If successful.
#   1 - If it fails.
#####
function ec2_authorize_security_group_ingress() {
    local security_group_id ip_address protocol from_port to_port response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_authorize_security_group_ingress"
        echo "Authorizes an ingress rule for an Amazon Elastic Compute Cloud (Amazon
        EC2) security group."
        echo "  -g security_group_id - The ID of the security group."
        echo "  -i ip_address - The IP address or CIDR block to authorize."
        echo "  -p protocol - The protocol to authorize (e.g., tcp, udp, icmp)."
        echo "  -f from_port - The start of the port range to authorize."
        echo "  -t to_port - The end of the port range to authorize."
        echo ""
    }

    # Retrieve the calling parameters.

```

```
while getopts "g:i:p:f:t:h" option; do
  case "${option}" in
    g) security_group_id="${OPTARG}" ;;
    i) ip_address="${OPTARG}" ;;
    p) protocol="${OPTARG}" ;;
    f) from_port="${OPTARG}" ;;
    t) to_port="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$security_group_id" ]]; then
  errecho "ERROR: You must provide a security group ID with the -g parameter."
  usage
  return 1
fi

if [[ -z "$ip_address" ]]; then
  errecho "ERROR: You must provide an IP address or CIDR block with the -i
parameter."
  usage
  return 1
fi

if [[ -z "$protocol" ]]; then
  errecho "ERROR: You must provide a protocol with the -p parameter."
  usage
  return 1
fi

if [[ -z "$from_port" ]]; then
  errecho "ERROR: You must provide a start port with the -f parameter."
  usage
  return 1
fi
```

```

if [[ -z "$to_port" ]]; then
    errecho "ERROR: You must provide an end port with the -t parameter."
    usage
    return 1
fi

response=$(aws ec2 authorize-security-group-ingress \
    --group-id "$security_group_id" \
    --cidr "${ip_address}/32" \
    --protocol "$protocol" \
    --port "$from_port-$to_port" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports authorize-security-group-ingress operation failed.
$response"
    return 1
}

return 0
}

#####
# function ec2_describe_images
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
# images.
#
# Parameters:
#     -i image_ids - A space-separated list of image IDs (optional).
#     -h - Display help.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_describe_images() {
    local image_ids response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_describe_images"
        echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2) images."
    }
}

```

```
    echo " -i image_ids - A space-separated list of image IDs (optional)."  
    echo " -h - Display help."  
    echo ""  
}  
  
# Retrieve the calling parameters.  
while getopts "i:h" option; do  
    case "${option}" in  
        i) image_ids="${OPTARG}" ;;  
        h)  
            usage  
            return 0  
            ;;  
        \?)  
            echo "Invalid parameter"  
            usage  
            return 1  
            ;;  
    esac  
done  
export OPTIND=1  
  
local aws_cli_args=(  
  
if [[ -n "$image_ids" ]]; then  
    # shellcheck disable=SC2206  
    aws_cli_args+=("--image-ids" $image_ids)  
fi  
  
response=$(aws ec2 describe-images \  
    "${aws_cli_args[@]}" \  
    --query 'Images[*].[Description,Architecture,ImageId]' \  
    --output text) || {  
    aws_cli_error_log ${?}  
    errecho "ERROR: AWS reports describe-images operation failed.$response"  
    return 1  
}  
  
echo "$response"  
  
return 0  
}
```

```
#####
```

```

# ec2_describe_instance_types
#
# This function describes EC2 instance types filtered by processor architecture
# and optionally by instance type. It takes the following arguments:
#
# -a, --architecture ARCHITECTURE Specify the processor architecture (e.g., x86_64)
# -t, --type INSTANCE_TYPE      Comma-separated list of instance types (e.g.,
#                               t2.micro)
# -h, --help                    Show the usage help
#
# The function prints the instance type and supported architecture for each
# matching instance type.
#####
function ec2_describe_instance_types() {
    local architecture=""
    local instance_types=""

    # bashsupport disable=BP5008
    function usage() {
        echo "Usage: ec2_describe_instance_types [-a|--architecture ARCHITECTURE] [-t|--
type INSTANCE_TYPE] [-h|--help]"
        echo "  -a, --architecture ARCHITECTURE Specify the processor architecture
(e.g., x86_64)"
        echo "  -t, --type INSTANCE_TYPE      Comma-separated list of instance types
(e.g., t2.micro)"
        echo "  -h, --help                    Show this help message"
    }

    while [[ $# -gt 0 ]]; do
        case "$1" in
            -a | --architecture)
                architecture="$2"
                shift 2
                ;;
            -t | --type)
                instance_types="$2"
                shift 2
                ;;
            -h | --help)
                usage
                return 0
                ;;
            *)
                echo "Unknown argument: $1"

```

```
        return 1
        ;;
    esac
done

if [[ -z "$architecture" ]]; then
    errecho "Error: Architecture not specified."
    usage
    return 1
fi

if [[ -z "$instance_types" ]]; then
    errecho "Error: Instance type not specified."
    usage
    return 1
fi

local tmp_json_file="temp_ec2.json"
echo -n '[
    {
        "Name": "processor-info.supported-architecture",
        "Values": [' >"$tmp_json_file"

local items
IFS=', ' read -ra items <<<"$architecture"
local array_size
array_size=${#items[@]}
for i in $(seq 0 $((array_size - 1))); do
    echo -n '""${items[$i]}""' >>"$tmp_json_file"
    if [[ $i -lt $((array_size - 1)) ]]; then
        echo -n ', ' >>"$tmp_json_file"
    fi
done
echo -n ']],
    {
        "Name": "instance-type",
        "Values": [' >>"$tmp_json_file"
IFS=', ' read -ra items <<<"$instance_types"
local array_size
array_size=${#items[@]}
for i in $(seq 0 $((array_size - 1))); do
    echo -n '""${items[$i]}""' >>"$tmp_json_file"
    if [[ $i -lt $((array_size - 1)) ]]; then
        echo -n ', ' >>"$tmp_json_file"
```

```

    fi
done

echo -n ']]]' >>"$tmp_json_file"

local response
response=$(aws ec2 describe-instance-types --filters file://"${tmp_json_file}" \
  --query 'InstanceTypes[*].[InstanceType]' --output text)

local error_code=$?

rm "$tmp_json_file"

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    echo "ERROR: AWS reports describe-instance-types operation failed."
    return 1
fi

echo "$response"
return 0
}

#####
# function ec2_run_instances
#
# This function launches one or more Amazon Elastic Compute Cloud (Amazon EC2)
# instances.
#
# Parameters:
#   -i image_id - The ID of the Amazon Machine Image (AMI) to use.
#   -t instance_type - The instance type to use (e.g., t2.micro).
#   -k key_pair_name - The name of the key pair to use.
#   -s security_group_id - The ID of the security group to use.
#   -c count - The number of instances to launch (default: 1).
#   -h - Display help.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
function ec2_run_instances() {
    local image_id instance_type key_pair_name security_group_id count response
    local option OPTARG # Required to use getopt command in a function.

```

```
# bashsupport disable=BP5008
function usage() {
    echo "function ec2_run_instances"
    echo "Launches one or more Amazon Elastic Compute Cloud (Amazon EC2) instances."
    echo "  -i image_id - The ID of the Amazon Machine Image (AMI) to use."
    echo "  -t instance_type - The instance type to use (e.g., t2.micro)."
    echo "  -k key_pair_name - The name of the key pair to use."
    echo "  -s security_group_id - The ID of the security group to use."
    echo "  -c count - The number of instances to launch (default: 1)."
    echo "  -h - Display help."
    echo ""
}

# Retrieve the calling parameters.
while getopts "i:t:k:s:c:h" option; do
    case "${option}" in
        i) image_id="${OPTARG}" ;;
        t) instance_type="${OPTARG}" ;;
        k) key_pair_name="${OPTARG}" ;;
        s) security_group_id="${OPTARG}" ;;
        c) count="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$image_id" ]]; then
    errecho "ERROR: You must provide an Amazon Machine Image (AMI) ID with the -i
parameter."
    usage
    return 1
fi

if [[ -z "$instance_type" ]]; then
    errecho "ERROR: You must provide an instance type with the -t parameter."
```



```

    usage
    return 1
fi

if [[ -z "$key_pair_name" ]]; then
    errecho "ERROR: You must provide a key pair name with the -k parameter."
    usage
    return 1
fi

if [[ -z "$security_group_id" ]]; then
    errecho "ERROR: You must provide a security group ID with the -s parameter."
    usage
    return 1
fi

if [[ -z "$count" ]]; then
    count=1
fi

response=$(aws ec2 run-instances \
    --image-id "$image_id" \
    --instance-type "$instance_type" \
    --key-name "$key_pair_name" \
    --security-group-ids "$security_group_id" \
    --count "$count" \
    --query 'Instances[*].[InstanceId]' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports run-instances operation failed.$response"
    return 1
}

echo "$response"

return 0
}

#####
# function ec2_describe_instances
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
# instances.
#

```

```

# Parameters:
#     -i instance_id - The ID of the instance to describe (optional).
#     -q query - The query to filter the response (optional).
#     -h - Display help.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_describe_instances() {
    local instance_id query response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_describe_instances"
        echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
instances."
        echo "  -i instance_id - The ID of the instance to describe (optional).\"
        echo "  -q query - The query to filter the response (optional).\"
        echo "  -h - Display help.\"
        echo \"\"
    }

    # Retrieve the calling parameters.
    while getopt \"i:q:h\" option; do
        case \"${option}\" in
            i) instance_id=\"${OPTARG}\" ;;
            q) query=\"${OPTARG}\" ;;
            h)
                usage
                return 0
                ;;
            \\?)
                echo \"Invalid parameter\"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    local aws_cli_args=()

```

```

if [[ -n "$instance_id" ]]; then
    # shellcheck disable=SC2206
    aws_cli_args+=("--instance-ids" $instance_id)
fi

local query_arg=""
if [[ -n "$query" ]]; then
    query_arg="--query '$query'"
else
    query_arg="--query Reservations[*].Instances[*].
[InstanceId,ImageId,InstanceType,KeyName,VpcId,PublicIpAddress,State.Name]"
fi

# shellcheck disable=SC2086
response=$(aws ec2 describe-instances \
    "${aws_cli_args[@]}" \
    $query_arg \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports describe-instances operation failed.$response"
    return 1
}

echo "$response"

return 0
}

#####
# function ec2_stop_instances
#
# This function stops one or more Amazon Elastic Compute Cloud (Amazon EC2)
# instances.
#
# Parameters:
#     -i instance_id - The ID(s) of the instance(s) to stop (comma-separated).
#     -h - Display help.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_stop_instances() {
    local instance_ids

```

```
local option OPTARG # Required to use getopt command in a function.

# bashsupport disable=BP5008
function usage() {
    echo "function ec2_stop_instances"
    echo "Stops one or more Amazon Elastic Compute Cloud (Amazon EC2) instances."
    echo "  -i instance_id - The ID(s) of the instance(s) to stop (comma-
separated)."
    echo "  -h - Display help."
    echo ""
}

# Retrieve the calling parameters.
while getopt "i:h" option; do
    case "${option}" in
        i) instance_ids="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$instance_ids" ]]; then
    errecho "ERROR: You must provide one or more instance IDs with the -i
parameter."
    usage
    return 1
fi

response=$(aws ec2 stop-instances \
    --instance-ids "${instance_ids}") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports stop-instances operation failed with $response."
    return 1
}

return 0
```

```

}

#####
# function ec2_start_instances
#
# This function starts one or more Amazon Elastic Compute Cloud (Amazon EC2)
# instances.
#
# Parameters:
#   -i instance_id - The ID(s) of the instance(s) to start (comma-separated).
#   -h - Display help.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
function ec2_start_instances() {
    local instance_ids
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_start_instances"
        echo "Starts one or more Amazon Elastic Compute Cloud (Amazon EC2) instances."
        echo "  -i instance_id - The ID(s) of the instance(s) to start (comma-
separated)."
        echo "  -h - Display help."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "i:h" option; do
        case "${option}" in
            i) instance_ids="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
}

```

```

done
export OPTIND=1

if [[ -z "$instance_ids" ]]; then
    errecho "ERROR: You must provide one or more instance IDs with the -i
parameter."
    usage
    return 1
fi

response=$(aws ec2 start-instances \
    --instance-ids "${instance_ids}") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports start-instances operation failed with $response."
    return 1
}

return 0
}

#####
# function ec2_allocate_address
#
# This function allocates an Elastic IP address for use with Amazon Elastic Compute
# Cloud (Amazon EC2) instances in a specific AWS Region.
#
# Parameters:
#     -d domain - The domain for the Elastic IP address (either 'vpc' or
# 'standard').
#
# Returns:
#     The allocated Elastic IP address, or an error message if the operation
# fails.
# And:
#     0 - If successful.
#     1 - If it fails.
#
#####
function ec2_allocate_address() {
    local domain response

    # Function to display usage information
    function usage() {
        echo "function ec2_allocate_address"
    }
}

```

```
    echo "Allocates an Elastic IP address for use with Amazon Elastic Compute Cloud
(Amazon EC2) instances in a specific AWS Region."
    echo "  -d domain - The domain for the Elastic IP address (either 'vpc' or
'standard')."
    echo ""
}

# Parse the command-line arguments
while getopts "d:h" option; do
    case "${option}" in
        d) domain="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

# Validate the input parameters
if [[ -z "$domain" ]]; then
    errecho "ERROR: You must provide a domain with the -d parameter (either 'vpc' or
'standard')."
    return 1
fi

if [[ "$domain" != "vpc" && "$domain" != "standard" ]]; then
    errecho "ERROR: Invalid domain value. Must be either 'vpc' or 'standard'."
    return 1
fi

# Allocate the Elastic IP address
response=$(aws ec2 allocate-address \
    --domain "$domain" \
    --query "[PublicIp,AllocationId]" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports allocate-address operation failed."
    errecho "$response"
```

```
    return 1
}

echo "$response"
return 0
}

#####
# function ec2_associate_address
#
# This function associates an Elastic IP address with an Amazon Elastic Compute
# Cloud (Amazon EC2) instance.
#
# Parameters:
#     -a allocation_id - The allocation ID of the Elastic IP address to associate.
#     -i instance_id - The ID of the EC2 instance to associate the Elastic IP
# address with.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#
#####
function ec2_associate_address() {
    local allocation_id instance_id response

    # Function to display usage information
    function usage() {
        echo "function ec2_associate_address"
        echo "Associates an Elastic IP address with an Amazon Elastic Compute Cloud
(Amazon EC2) instance."
        echo "  -a allocation_id - The allocation ID of the Elastic IP address to
associate."
        echo "  -i instance_id - The ID of the EC2 instance to associate the Elastic IP
address with."
        echo ""
    }

    # Parse the command-line arguments
    while getopts "a:i:h" option; do
        case "${option}" in
            a) allocation_id="${OPTARG}" ;;
            i) instance_id="${OPTARG}" ;;
            h)

```



```

        usage
        return 0
        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

# Validate the input parameters
if [[ -z "$allocation_id" ]]; then
    errecho "ERROR: You must provide an allocation ID with the -a parameter."
    return 1
fi

if [[ -z "$instance_id" ]]; then
    errecho "ERROR: You must provide an instance ID with the -i parameter."
    return 1
fi

# Associate the Elastic IP address
response=$(aws ec2 associate-address \
    --allocation-id "$allocation_id" \
    --instance-id "$instance_id" \
    --query "AssociationId" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports associate-address operation failed."
    errecho "$response"
    return 1
}

echo "$response"
return 0
}

#####
# function ec2_disassociate_address
#
# This function disassociates an Elastic IP address from an Amazon Elastic Compute
Cloud (Amazon EC2) instance.

```

```
#
# Parameters:
#   -a association_id - The association ID that represents the association of
#   the Elastic IP address with an instance.
#
# And:
#   0 - If successful.
#   1 - If it fails.
#
#####
function ec2_disassociate_address() {
    local association_id response

    # Function to display usage information
    function usage() {
        echo "function ec2_disassociate_address"
        echo "Disassociates an Elastic IP address from an Amazon Elastic Compute Cloud
(Amazon EC2) instance."
        echo "  -a association_id - The association ID that represents the association
of the Elastic IP address with an instance."
        echo ""
    }

    # Parse the command-line arguments
    while getopts "a:h" option; do
        case "${option}" in
            a) association_id="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    # Validate the input parameters
    if [[ -z "$association_id" ]]; then
        errecho "ERROR: You must provide an association ID with the -a parameter."
        return 1
    fi
}
```

```

fi

response=$(aws ec2 disassociate-address \
  --association-id "$association_id") || {
  aws_cli_error_log ${?}
  errecho "ERROR: AWS reports disassociate-address operation failed."
  errecho "$response"
  return 1
}

return 0
}

#####
# function ec2_release_address
#
# This function releases an Elastic IP address from an Amazon Elastic Compute Cloud
# (Amazon EC2) instance.
#
# Parameters:
#   -a allocation_id - The allocation ID of the Elastic IP address to release.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#
#####
function ec2_release_address() {
  local allocation_id response

  # Function to display usage information
  function usage() {
    echo "function ec2_release_address"
    echo "Releases an Elastic IP address from an Amazon Elastic Compute Cloud
(Amazon EC2) instance."
    echo "  -a allocation_id - The allocation ID of the Elastic IP address to
release."
    echo ""
  }

  # Parse the command-line arguments
  while getopts "a:h" option; do
    case "${option}" in
      a) allocation_id="${OPTARG}" ;;

```

```
h)
  usage
  return 0
  ;;
\?)
  echo "Invalid parameter"
  usage
  return 1
  ;;
esac
done
export OPTIND=1

# Validate the input parameters
if [[ -z "$allocation_id" ]]; then
  errecho "ERROR: You must provide an allocation ID with the -a parameter."
  return 1
fi

response=$(aws ec2 release-address \
  --allocation-id "$allocation_id") || {
  aws_cli_error_log ${?}
  errecho "ERROR: AWS reports release-address operation failed."
  errecho "$response"
  return 1
}

return 0
}

#####
# function ec2_terminate_instances
#
# This function terminates one or more Amazon Elastic Compute Cloud (Amazon EC2)
# instances using the AWS CLI.
#
# Parameters:
#   -i instance_ids - A space-separated list of instance IDs.
#   -h - Display help.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
```

```
function ec2_terminate_instances() {
    local instance_ids response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_terminate_instances"
        echo "Terminates one or more Amazon Elastic Compute Cloud (Amazon EC2)
instances."
        echo "  -i instance_ids - A space-separated list of instance IDs."
        echo "  -h - Display help."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "i:h" option; do
        case "${option}" in
            i) instance_ids="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    # Check if instance ID is provided
    if [[ -z "${instance_ids}" ]]; then
        echo "Error: Missing required instance IDs parameter."
        usage
        return 1
    fi

    # shellcheck disable=SC2086
    response=$(aws ec2 terminate-instances \
        "--instance-ids" $instance_ids \
        "--query 'TerminatingInstances[*].[InstanceId,CurrentState.Name]' \
        "--output text) || {
        aws_cli_error_log ${?}
    }
```

```

    errecho "ERROR: AWS reports terminate-instances operation failed.$response"
    return 1
}

return 0
}

#####
# function ec2_delete_security_group
#
# This function deletes an Amazon Elastic Compute Cloud (Amazon EC2) security group.
#
# Parameters:
#     -i security_group_id - The ID of the security group to delete.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_delete_security_group() {
    local security_group_id response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_delete_security_group"
        echo "Deletes an Amazon Elastic Compute Cloud (Amazon EC2) security group."
        echo "  -i security_group_id - The ID of the security group to delete."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "i:h" option; do
        case "${option}" in
            i) security_group_id="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
}

```

```

    esac
done
export OPTIND=1

if [[ -z "$security_group_id" ]]; then
    errecho "ERROR: You must provide a security group ID with the -i parameter."
    usage
    return 1
fi

response=$(aws ec2 delete-security-group --group-id "$security_group_id" --output
text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports delete-security-group operation failed.$response"
    return 1
}

return 0
}

#####
# function ec2_delete_keypair
#
# This function deletes an Amazon EC2 ED25519 or 2048-bit RSA key pair.
#
# Parameters:
#     -n key_pair_name - A key pair name.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_delete_keypair() {
    local key_pair_name response

    local option OPTARG # Required to use getopt command in a function.
    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_delete_keypair"
        echo "Deletes an Amazon EC2 ED25519 or 2048-bit RSA key pair."
        echo "  -n key_pair_name - A key pair name."
        echo ""
    }
}

```

```

# Retrieve the calling parameters.
while getopts "n:h" option; do
  case "${option}" in
    n) key_pair_name="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$key_pair_name" ]]; then
  errecho "ERROR: You must provide a key pair name with the -n parameter."
  usage
  return 1
fi

response=$(aws ec2 delete-key-pair \
  --key-name "$key_pair_name") || {
  aws_cli_error_log ${?}
  errecho "ERROR: AWS reports delete-key-pair operation failed.$response"
  return 1
}

return 0
}

```

Las funciones de utilidad usadas en este escenario.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

```



```

}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Para API obtener más información, consulte los siguientes temas en AWS CLI Command Reference.
  - [AllocateAddress](#)
  - [AssociateAddress](#)
  - [AuthorizeSecurityGroupIngress](#)

- [CreateKeyPair](#)
- [CreateSecurityGroup](#)
- [DeleteKeyPair](#)
- [DeleteSecurityGroup](#)
- [DescribeImages](#)
- [DescribeInstanceTypes](#)
- [DescribeInstances](#)
- [DescribeKeyPairs](#)
- [DescribeSecurityGroups](#)
- [DisassociateAddress](#)
- [ReleaseAddress](#)
- [RunInstances](#)
- [StartInstances](#)
- [StopInstances](#)
- [TerminateInstances](#)
- [UnmonitorInstances](#)

## Acciones

### **AllocateAddress**

En el siguiente ejemplo de código, se muestra cómo usar `AllocateAddress`.

AWS CLI con el script Bash

#### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####  
# function ec2_allocate_address
```

```

# This function allocates an Elastic IP address for use with Amazon Elastic Compute
Cloud (Amazon EC2) instances in a specific AWS Region.
#
# Parameters:
#     -d domain - The domain for the Elastic IP address (either 'vpc' or
'standard').
#
# Returns:
#     The allocated Elastic IP address, or an error message if the operation
fails.
# And:
#     0 - If successful.
#     1 - If it fails.
#
#####
function ec2_allocate_address() {
    local domain response

    # Function to display usage information
    function usage() {
        echo "function ec2_allocate_address"
        echo "Allocates an Elastic IP address for use with Amazon Elastic Compute Cloud
(Amazon EC2) instances in a specific AWS Region."
        echo "  -d domain - The domain for the Elastic IP address (either 'vpc' or
'standard')."
        echo ""
    }

    # Parse the command-line arguments
    while getopts "d:h" option; do
        case "${option}" in
            d) domain="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1
}

```

```

# Validate the input parameters
if [[ -z "$domain" ]]; then
    errecho "ERROR: You must provide a domain with the -d parameter (either 'vpc' or
'standard')."
    return 1
fi

if [[ "$domain" != "vpc" && "$domain" != "standard" ]]; then
    errecho "ERROR: Invalid domain value. Must be either 'vpc' or 'standard'."
    return 1
fi

# Allocate the Elastic IP address
response=$(aws ec2 allocate-address \
    --domain "$domain" \
    --query "[PublicIp,AllocationId]" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports allocate-address operation failed."
    errecho "$response"
    return 1
}

echo "$response"
return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#

```

```

# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Para API obtener más información, consulte [AllocateAddress](#) la Referencia de AWS CLI comandos.

## AssociateAddress

En el siguiente ejemplo de código, se muestra cómo usar AssociateAddress.

## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_associate_address
#
# This function associates an Elastic IP address with an Amazon Elastic Compute
# Cloud (Amazon EC2) instance.
#
# Parameters:
#   -a allocation_id - The allocation ID of the Elastic IP address to associate.
#   -i instance_id - The ID of the EC2 instance to associate the Elastic IP
# address with.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#
#####
function ec2_associate_address() {
    local allocation_id instance_id response

    # Function to display usage information
    function usage() {
        echo "function ec2_associate_address"
        echo "Associates an Elastic IP address with an Amazon Elastic Compute Cloud
        (Amazon EC2) instance."
        echo "  -a allocation_id - The allocation ID of the Elastic IP address to
        associate."
        echo "  -i instance_id - The ID of the EC2 instance to associate the Elastic IP
        address with."
        echo ""
    }

    # Parse the command-line arguments
    while getopts "a:i:h" option; do
        case "${option}" in
```

```
a) allocation_id="${OPTARG}" ;;
i) instance_id="${OPTARG}" ;;
h)
    usage
    return 0
    ;;
\?)
    echo "Invalid parameter"
    usage
    return 1
    ;;
esac
done
export OPTIND=1

# Validate the input parameters
if [[ -z "$allocation_id" ]]; then
    errecho "ERROR: You must provide an allocation ID with the -a parameter."
    return 1
fi

if [[ -z "$instance_id" ]]; then
    errecho "ERROR: You must provide an instance ID with the -i parameter."
    return 1
fi

# Associate the Elastic IP address
response=$(aws ec2 associate-address \
    --allocation-id "$allocation_id" \
    --instance-id "$instance_id" \
    --query "AssociationId" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports associate-address operation failed."
    errecho "$response"
    return 1
}

echo "$response"
return 0
}
```

## Las funciones de utilidad utilizadas en este ejemplo.

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}
```



- Para API obtener más información, consulte [AssociateAddress](#) la Referencia de AWS CLI comandos.

## AuthorizeSecurityGroupIngress

En el siguiente ejemplo de código, se muestra cómo usar AuthorizeSecurityGroupIngress.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_authorize_security_group_ingress
#
# This function authorizes an ingress rule for an Amazon Elastic Compute Cloud
# (Amazon EC2) security group.
#
# Parameters:
#   -g security_group_id - The ID of the security group.
#   -i ip_address - The IP address or CIDR block to authorize.
#   -p protocol - The protocol to authorize (e.g., tcp, udp, icmp).
#   -f from_port - The start of the port range to authorize.
#   -t to_port - The end of the port range to authorize.
#
# And:
#   0 - If successful.
#   1 - If it fails.
#####
function ec2_authorize_security_group_ingress() {
    local security_group_id ip_address protocol from_port to_port response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_authorize_security_group_ingress"
```

```
    echo "Authorizes an ingress rule for an Amazon Elastic Compute Cloud (Amazon
EC2) security group."
    echo "  -g security_group_id - The ID of the security group."
    echo "  -i ip_address - The IP address or CIDR block to authorize."
    echo "  -p protocol - The protocol to authorize (e.g., tcp, udp, icmp)."
    echo "  -f from_port - The start of the port range to authorize."
    echo "  -t to_port - The end of the port range to authorize."
    echo ""
}

# Retrieve the calling parameters.
while getopts "g:i:p:f:t:h" option; do
    case "${option}" in
        g) security_group_id="${OPTARG}" ;;
        i) ip_address="${OPTARG}" ;;
        p) protocol="${OPTARG}" ;;
        f) from_port="${OPTARG}" ;;
        t) to_port="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$security_group_id" ]]; then
    errecho "ERROR: You must provide a security group ID with the -g parameter."
    usage
    return 1
fi

if [[ -z "$ip_address" ]]; then
    errecho "ERROR: You must provide an IP address or CIDR block with the -i
parameter."
    usage
    return 1
fi
```

```

if [[ -z "$protocol" ]]; then
    errecho "ERROR: You must provide a protocol with the -p parameter."
    usage
    return 1
fi

if [[ -z "$from_port" ]]; then
    errecho "ERROR: You must provide a start port with the -f parameter."
    usage
    return 1
fi

if [[ -z "$to_port" ]]; then
    errecho "ERROR: You must provide an end port with the -t parameter."
    usage
    return 1
fi

response=$(aws ec2 authorize-security-group-ingress \
    --group-id "$security_group_id" \
    --cidr "${ip_address}/32" \
    --protocol "$protocol" \
    --port "$from_port-$to_port" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports authorize-security-group-ingress operation failed.
$response"
    return 1
}

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

```

```

}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Para API obtener más información, consulte [AuthorizeSecurityGroupIngress](#) la Referencia de AWS CLI comandos.

## CreateKeyPair

En el siguiente ejemplo de código, se muestra cómo usar CreateKeyPair.

## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_create_keypair
#
# This function creates an Amazon Elastic Compute Cloud (Amazon EC2) ED25519 or
# 2048-bit RSA key pair
# and writes it to a file.
#
# Parameters:
#     -n key_pair_name - A key pair name.
#     -f file_path - File to store the key pair.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_create_keypair() {
    local key_pair_name file_path response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_create_keypair"
        echo "Creates an Amazon Elastic Compute Cloud (Amazon EC2) ED25519 or 2048-bit
        RSA key pair"
        echo " and writes it to a file."
        echo "  -n key_pair_name - A key pair name."
        echo "  -f file_path - File to store the key pair."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:f:h" option; do
        case "${option}" in
            n) key_pair_name="${OPTARG}" ;;

```

```
f) file_path="${OPTARG}" ;;
h)
    usage
    return 0
    ;;
\?)
    echo "Invalid parameter"
    usage
    return 1
    ;;
esac
done
export OPTIND=1

if [[ -z "$key_pair_name" ]]; then
    errecho "ERROR: You must provide a key name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$file_path" ]]; then
    errecho "ERROR: You must provide a file path with the -f parameter."
    usage
    return 1
fi

response=$(aws ec2 create-key-pair \
    --key-name "$key_pair_name" \
    --query 'KeyMaterial' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports create-access-key operation failed.$response"
    return 1
}

if [[ -n "$file_path" ]]; then
    echo "$response" >"$file_path"
fi

return 0
}
```

## Las funciones de utilidad utilizadas en este ejemplo.

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}
```

- Para API obtener más información, consulte [CreateKeyPair](#) la Referencia de AWS CLI comandos.

## CreateSecurityGroup

En el siguiente ejemplo de código, se muestra cómo usar CreateSecurityGroup.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_create_security_group
#
# This function creates an Amazon Elastic Compute Cloud (Amazon EC2) security group.
#
# Parameters:
#     -n security_group_name - The name of the security group.
#     -d security_group_description - The description of the security group.
#
# Returns:
#     The ID of the created security group, or an error message if the operation
#     fails.
# And:
#     0 - If successful.
#     1 - If it fails.
#
#####
function ec2_create_security_group() {
    local security_group_name security_group_description response

    # Function to display usage information
    function usage() {
        echo "function ec2_create_security_group"
        echo "Creates an Amazon Elastic Compute Cloud (Amazon EC2) security group."
    }
}
```



```
    echo " -n security_group_name - The name of the security group."
    echo " -d security_group_description - The description of the security group."
    echo ""
}

# Parse the command-line arguments
while getopts "n:d:h" option; do
    case "${option}" in
        n) security_group_name="${OPTARG}" ;;
        d) security_group_description="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

# Validate the input parameters
if [[ -z "$security_group_name" ]]; then
    errecho "ERROR: You must provide a security group name with the -n parameter."
    return 1
fi

if [[ -z "$security_group_description" ]]; then
    errecho "ERROR: You must provide a security group description with the -d
parameter."
    return 1
fi

# Create the security group
response=$(aws ec2 create-security-group \
    --group-name "$security_group_name" \
    --description "$security_group_description" \
    --query "GroupId" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports create-security-group operation failed."
    errecho "$response"
```

```

    return 1
}

echo "$response"
return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then

```

```

    errecho " The system environment or configuration was invalid."
elif [ "$err_code" == 254 ]; then
    errecho " The service returned an error."
elif [ "$err_code" == 255 ]; then
    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Para API obtener más información, consulte [CreateSecurityGroup](#) la Referencia de AWS CLI comandos.

## DeleteKeyPair

En el siguiente ejemplo de código, se muestra cómo usar DeleteKeyPair.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function ec2_delete_keypair
#
# This function deletes an Amazon EC2 ED25519 or 2048-bit RSA key pair.
#
# Parameters:
#     -n key_pair_name - A key pair name.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_delete_keypair() {
    local key_pair_name response

```

```
local option OPTARG # Required to use getopt command in a function.
# bashsupport disable=BP5008
function usage() {
    echo "function ec2_delete_keypair"
    echo "Deletes an Amazon EC2 ED25519 or 2048-bit RSA key pair."
    echo "  -n key_pair_name - A key pair name."
    echo ""
}

# Retrieve the calling parameters.
while getopt "n:h" option; do
    case "${option}" in
        n) key_pair_name="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$key_pair_name" ]]; then
    errecho "ERROR: You must provide a key pair name with the -n parameter."
    usage
    return 1
fi

response=$(aws ec2 delete-key-pair \
    --key-name "$key_pair_name") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports delete-key-pair operation failed.$response"
    return 1
}

return 0
}
```

## Las funciones de utilidad utilizadas en este ejemplo.

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}
}
```

- Para API obtener más información, consulte [DeleteKeyPair](#) la Referencia de AWS CLI comandos.

## DeleteSecurityGroup

En el siguiente ejemplo de código, se muestra cómo usar DeleteSecurityGroup.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_delete_security_group
#
# This function deletes an Amazon Elastic Compute Cloud (Amazon EC2) security group.
#
# Parameters:
#     -i security_group_id - The ID of the security group to delete.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_delete_security_group() {
    local security_group_id response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_delete_security_group"
        echo "Deletes an Amazon Elastic Compute Cloud (Amazon EC2) security group."
        echo "  -i security_group_id - The ID of the security group to delete."
        echo ""
    }
}
```

```

# Retrieve the calling parameters.
while getopts "i:h" option; do
  case "${option}" in
    i) security_group_id="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$security_group_id" ]]; then
  errecho "ERROR: You must provide a security group ID with the -i parameter."
  usage
  return 1
fi

response=$(aws ec2 delete-security-group --group-id "$security_group_id" --output
text) || {
  aws_cli_error_log ${?}
  errecho "ERROR: AWS reports delete-security-group operation failed.$response"
  return 1
}

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

```

```

}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Para API obtener más información, consulte [DeleteSecurityGroup](#) la Referencia de AWS CLI comandos.

## DescribeImages

En el siguiente ejemplo de código, se muestra cómo usar DescribeImages.



## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_describe_images
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
# images.
#
# Parameters:
#   -i image_ids - A space-separated list of image IDs (optional).
#   -h - Display help.
#
# And:
#   0 - If successful.
#   1 - If it fails.
#####
function ec2_describe_images() {
    local image_ids response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_describe_images"
        echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2) images."
        echo "  -i image_ids - A space-separated list of image IDs (optional)."
        echo "  -h - Display help."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "i:h" option; do
        case "${option}" in
            i) image_ids="${OPTARG}" ;;
            h)
                usage
                return 0
        esac
    done
}
```

```

        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

local aws_cli_args=()

if [[ -n "$image_ids" ]]; then
    # shellcheck disable=SC2206
    aws_cli_args+=("--image-ids" $image_ids)
fi

response=$(aws ec2 describe-images \
    "${aws_cli_args[@]}" \
    --query 'Images[*].[Description,Architecture,ImageId]' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports describe-images operation failed.$response"
    return 1
}

echo "$response"

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

```

```
#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Para API obtener más información, consulte [DescribeImages](#) la Referencia de AWS CLI comandos.

## DescribeInstanceTypes

En el siguiente ejemplo de código, se muestra cómo usar DescribeInstanceTypes.

## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# ec2_describe_instance_types
#
# This function describes EC2 instance types filtered by processor architecture
# and optionally by instance type. It takes the following arguments:
#
# -a, --architecture ARCHITECTURE Specify the processor architecture (e.g., x86_64)
# -t, --type INSTANCE_TYPE       Comma-separated list of instance types (e.g.,
#                                 t2.micro)
# -h, --help                       Show the usage help
#
# The function prints the instance type and supported architecture for each
# matching instance type.
#####
function ec2_describe_instance_types() {
    local architecture=""
    local instance_types=""

    # bashsupport disable=BP5008
    function usage() {
        echo "Usage: ec2_describe_instance_types [-a|--architecture ARCHITECTURE] [-t|--
type INSTANCE_TYPE] [-h|--help]"
        echo "  -a, --architecture ARCHITECTURE Specify the processor architecture
(e.g., x86_64)"
        echo "  -t, --type INSTANCE_TYPE       Comma-separated list of instance types
(e.g., t2.micro)"
        echo "  -h, --help                       Show this help message"
    }

    while [[ $# -gt 0 ]]; do
        case "$1" in
            -a | --architecture)
                architecture="$2"
                shift 2

```

```
    ;;
    -t | --type)
        instance_types="$2"
        shift 2
        ;;
    -h | --help)
        usage
        return 0
        ;;
    *)
        echo "Unknown argument: $1"
        return 1
        ;;
esac
done

if [[ -z "$architecture" ]]; then
    errecho "Error: Architecture not specified."
    usage
    return 1
fi

if [[ -z "$instance_types" ]]; then
    errecho "Error: Instance type not specified."
    usage
    return 1
fi

local tmp_json_file="temp_ec2.json"
echo -n '['
{
    "Name": "processor-info.supported-architecture",
    "Values": [' >"$tmp_json_file"

local items
IFS=',' read -ra items <<<"$architecture"
local array_size
array_size=${#items[@]}
for i in $(seq 0 $((array_size - 1))); do
    echo -n '"${items[$i]}"' >>"$tmp_json_file"
    if [[ $i -lt $((array_size - 1)) ]]; then
        echo -n ',' >>"$tmp_json_file"
    fi
done
```

```

echo -n ']],
{
  "Name": "instance-type",
  "Values": [' >>"$tmp_json_file"
IFS=', ' read -ra items <<<"$instance_types"
local array_size
array_size=${#items[@]}
for i in $(seq 0 $((array_size - 1))); do
  echo -n '""${items[$i]}""' >>"$tmp_json_file"
  if [[ $i -lt $((array_size - 1)) ]]; then
    echo -n ', ' >>"$tmp_json_file"
  fi
done

echo -n ']]]' >>"$tmp_json_file"

local response
response=$(aws ec2 describe-instance-types --filters file://"${tmp_json_file}" \
  --query 'InstanceTypes[*].[InstanceType]' --output text)

local error_code=$?

rm "$tmp_json_file"

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  echo "ERROR: AWS reports describe-instance-types operation failed."
  return 1
fi

echo "$response"
return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {

```

```

    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Para API obtener más información, consulte [DescribeInstanceTypes](#) la Referencia de AWS CLI comandos.

## DescribeInstances

En el siguiente ejemplo de código, se muestra cómo usar DescribeInstances.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_describe_instances
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
# instances.
#
# Parameters:
#   -i instance_id - The ID of the instance to describe (optional).
#   -q query - The query to filter the response (optional).
#   -h - Display help.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
function ec2_describe_instances() {
    local instance_id query response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_describe_instances"
        echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
instances."
        echo "  -i instance_id - The ID of the instance to describe (optional)."
        echo "  -q query - The query to filter the response (optional)."
        echo "  -h - Display help."
        echo ""
    }

    # Retrieve the calling parameters.
```



```

while getopts "i:q:h" option; do
  case "${option}" in
    i) instance_id="${OPTARG}" ;;
    q) query="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

local aws_cli_args=()

if [[ -n "$instance_id" ]]; then
  # shellcheck disable=SC2206
  aws_cli_args+=("--instance-ids" $instance_id)
fi

local query_arg=""
if [[ -n "$query" ]]; then
  query_arg="--query '$query'"
else
  query_arg="--query Reservations[*].Instances[*].
[InstanceId,ImageId,InstanceType,KeyName,VpcId,PublicIpAddress,State.Name]"
fi

# shellcheck disable=SC2086
response=$(aws ec2 describe-instances \
  "${aws_cli_args[@]}" \
  $query_arg \
  --output text) || {
  aws_cli_error_log ${?}
  errecho "ERROR: AWS reports describe-instances operation failed.$response"
  return 1
}

echo "$response"

```

```
    return 0
}
```

Las funciones de utilidad utilizadas en este ejemplo.

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
```

```

    errecho " 255 is a catch-all error."
fi

return 0
}

```

- Para API obtener más información, consulte [DescribeInstances](#) la Referencia de AWS CLI comandos.

## DescribeKeyPairs

En el siguiente ejemplo de código, se muestra cómo usar DescribeKeyPairs.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function ec2_describe_key_pairs
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2) key
# pairs.
#
# Parameters:
#     -h - Display help.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_describe_key_pairs() {
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_describe_key_pairs"
    }
}

```

```

    echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2) key
pairs."
    echo " -h - Display help."
    echo ""
}

# Retrieve the calling parameters.
while getopts "h" option; do
    case "${option}" in
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

local response

response=$(aws ec2 describe-key-pairs \
    --query 'KeyPairs[*].[KeyName, KeyFingerprint]' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports describe-key-pairs operation failed.$response"
    return 1
}

echo "$response"

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#

```

```

# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Para API obtener más información, consulte [DescribeKeyPairs](#) la Referencia de AWS CLI comandos.

## DescribeSecurityGroups

En el siguiente ejemplo de código, se muestra cómo usar DescribeSecurityGroups.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_describe_security_groups
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
# security groups.
#
# Parameters:
#   -g security_group_id - The ID of the security group to describe (optional).
#
# And:
#   0 - If successful.
#   1 - If it fails.
#####
function ec2_describe_security_groups() {
    local security_group_id response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_describe_security_groups"
        echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2) security
groups."
        echo "  -g security_group_id - The ID of the security group to describe
(optional)."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "g:h" option; do
        case "${option}" in
            g) security_group_id="${OPTARG}" ;;
        esac
    done
}
```

```

h)
  usage
  return 0
  ;;
\?)
  echo "Invalid parameter"
  usage
  return 1
  ;;
esac
done
export OPTIND=1

local query="SecurityGroups[*].[GroupName, GroupId, VpcId, IpPermissions[*].
[IpProtocol, FromPort, ToPort, IpRanges[*].CidrIp]]"

if [[ -n "$security_group_id" ]]; then
  response=$(aws ec2 describe-security-groups --group-ids "$security_group_id" --
query "${query}" --output text)
else
  response=$(aws ec2 describe-security-groups --query "${query}" --output text)
fi

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports describe-security-groups operation failed.$response"
  return 1
fi

echo "$response"

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).

```

```
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Para API obtener más información, consulte [DescribeSecurityGroups](#) la Referencia de AWS CLI comandos.



## DisassociateAddress

En el siguiente ejemplo de código, se muestra cómo usar `DisassociateAddress`.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_disassociate_address
#
# This function disassociates an Elastic IP address from an Amazon Elastic Compute
# Cloud (Amazon EC2) instance.
#
# Parameters:
#     -a association_id - The association ID that represents the association of
#     the Elastic IP address with an instance.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#
#####
function ec2_disassociate_address() {
    local association_id response

    # Function to display usage information
    function usage() {
        echo "function ec2_disassociate_address"
        echo "Disassociates an Elastic IP address from an Amazon Elastic Compute Cloud
(Amazon EC2) instance."
        echo " -a association_id - The association ID that represents the association
of the Elastic IP address with an instance."
        echo ""
    }

    # Parse the command-line arguments
    while getopts "a:h" option; do
        case "${option}" in
```

```

    a) association_id="${OPTARG}" ;;
    h)
        usage
        return 0
        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
esac
done
export OPTIND=1

# Validate the input parameters
if [[ -z "$association_id" ]]; then
    errecho "ERROR: You must provide an association ID with the -a parameter."
    return 1
fi

response=$(aws ec2 disassociate-address \
    --association-id "$association_id") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports disassociate-address operation failed."
    errecho "$response"
    return 1
}

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

```

```
#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Para API obtener más información, consulte [DisassociateAddress](#) la Referencia de AWS CLI comandos.

## ReleaseAddress

En el siguiente ejemplo de código, se muestra cómo usar ReleaseAddress.

## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_release_address
#
# This function releases an Elastic IP address from an Amazon Elastic Compute Cloud
  (Amazon EC2) instance.
#
# Parameters:
#   -a allocation_id - The allocation ID of the Elastic IP address to release.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#
#####
function ec2_release_address() {
    local allocation_id response

    # Function to display usage information
    function usage() {
        echo "function ec2_release_address"
        echo "Releases an Elastic IP address from an Amazon Elastic Compute Cloud
(Amazon EC2) instance."
        echo "  -a allocation_id - The allocation ID of the Elastic IP address to
release."
        echo ""
    }

    # Parse the command-line arguments
    while getopts "a:h" option; do
        case "${option}" in
            a) allocation_id="${OPTARG}" ;;
            h)
                usage
                return 0
        esac
    done
}
```

```

        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
    ;;
esac
done
export OPTIND=1

# Validate the input parameters
if [[ -z "$allocation_id" ]]; then
    errecho "ERROR: You must provide an allocation ID with the -a parameter."
    return 1
fi

response=$(aws ec2 release-address \
    --allocation-id "$allocation_id") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports release-address operation failed."
    errecho "$response"
    return 1
}

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.

```

```
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}
```

- Para API obtener más información, consulte [ReleaseAddress](#) la Referencia de AWS CLI comandos.

## RunInstances

En el siguiente ejemplo de código, se muestra cómo usar RunInstances.

## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_run_instances
#
# This function launches one or more Amazon Elastic Compute Cloud (Amazon EC2)
# instances.
#
# Parameters:
#   -i image_id - The ID of the Amazon Machine Image (AMI) to use.
#   -t instance_type - The instance type to use (e.g., t2.micro).
#   -k key_pair_name - The name of the key pair to use.
#   -s security_group_id - The ID of the security group to use.
#   -c count - The number of instances to launch (default: 1).
#   -h - Display help.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
function ec2_run_instances() {
    local image_id instance_type key_pair_name security_group_id count response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_run_instances"
        echo "Launches one or more Amazon Elastic Compute Cloud (Amazon EC2) instances."
        echo "  -i image_id - The ID of the Amazon Machine Image (AMI) to use."
        echo "  -t instance_type - The instance type to use (e.g., t2.micro)."
        echo "  -k key_pair_name - The name of the key pair to use."
        echo "  -s security_group_id - The ID of the security group to use."
        echo "  -c count - The number of instances to launch (default: 1)."
        echo "  -h - Display help."
        echo ""
    }
}
```

```
# Retrieve the calling parameters.
while getopts "i:t:k:s:c:h" option; do
  case "${option}" in
    i) image_id="${OPTARG}" ;;
    t) instance_type="${OPTARG}" ;;
    k) key_pair_name="${OPTARG}" ;;
    s) security_group_id="${OPTARG}" ;;
    c) count="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$image_id" ]]; then
  errecho "ERROR: You must provide an Amazon Machine Image (AMI) ID with the -i
parameter."
  usage
  return 1
fi

if [[ -z "$instance_type" ]]; then
  errecho "ERROR: You must provide an instance type with the -t parameter."
  usage
  return 1
fi

if [[ -z "$key_pair_name" ]]; then
  errecho "ERROR: You must provide a key pair name with the -k parameter."
  usage
  return 1
fi

if [[ -z "$security_group_id" ]]; then
  errecho "ERROR: You must provide a security group ID with the -s parameter."
  usage
```



```

    return 1
fi

if [[ -z "$count" ]]; then
    count=1
fi

response=$(aws ec2 run-instances \
  --image-id "$image_id" \
  --instance-type "$instance_type" \
  --key-name "$key_pair_name" \
  --security-group-ids "$security_group_id" \
  --count "$count" \
  --query 'Instances[*].[InstanceId]' \
  --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports run-instances operation failed.$response"
    return 1
}

echo "$response"

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:

```

```
# $1 - The error code returned by the AWS CLI.
#
# Returns:
# 0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi


    return 0
}
```

- Para API obtener más información, consulte [RunInstances](#) la Referencia de AWS CLI comandos.

## StartInstances

En el siguiente ejemplo de código, se muestra cómo usar StartInstances.

## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_start_instances
#
# This function starts one or more Amazon Elastic Compute Cloud (Amazon EC2)
instances.
#
# Parameters:
#     -i instance_id - The ID(s) of the instance(s) to start (comma-separated).
#     -h - Display help.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_start_instances() {
    local instance_ids
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_start_instances"
        echo "Starts one or more Amazon Elastic Compute Cloud (Amazon EC2) instances."
        echo "  -i instance_id - The ID(s) of the instance(s) to start (comma-
separated)."
        echo "  -h - Display help."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "i:h" option; do
        case "${option}" in
            i) instance_ids="${OPTARG}" ;;
            h)
                usage
        esac
    done
}
```

```

        return 0
        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

if [[ -z "$instance_ids" ]]; then
    errecho "ERROR: You must provide one or more instance IDs with the -i
parameter."
    usage
    return 1
fi

response=$(aws ec2 start-instances \
--instance-ids "${instance_ids}") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports start-instances operation failed with $response."
    return 1
}

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#

```

```

# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Para API obtener más información, consulte [StartInstances](#) la Referencia de AWS CLI comandos.

## StopInstances

En el siguiente ejemplo de código, se muestra cómo usar StopInstances.

## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_stop_instances
#
# This function stops one or more Amazon Elastic Compute Cloud (Amazon EC2)
# instances.
#
# Parameters:
#   -i instance_id - The ID(s) of the instance(s) to stop (comma-separated).
#   -h - Display help.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
function ec2_stop_instances() {
    local instance_ids
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_stop_instances"
        echo "Stops one or more Amazon Elastic Compute Cloud (Amazon EC2) instances."
        echo "  -i instance_id - The ID(s) of the instance(s) to stop (comma-
separated)."
        echo "  -h - Display help."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "i:h" option; do
        case "${option}" in
            i) instance_ids="${OPTARG}" ;;
            h)
                usage
            )
        esac
    done
}
```

```

        return 0
        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

if [[ -z "$instance_ids" ]]; then
    errecho "ERROR: You must provide one or more instance IDs with the -i
parameter."
    usage
    return 1
fi

response=$(aws ec2 stop-instances \
--instance-ids "${instance_ids}") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports stop-instances operation failed with $response."
    return 1
}

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function aws_cli_error_log()
#

```

```

# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#     $1 - The error code returned by the AWS CLI.
#
# Returns:
#     0: - Success.
#
#####
function aws_cli_error_log() {
    local err_code=$1
    errecho "Error code : $err_code"
    if [ "$err_code" == 1 ]; then
        errecho " One or more S3 transfers failed."
    elif [ "$err_code" == 2 ]; then
        errecho " Command line failed to parse."
    elif [ "$err_code" == 130 ]; then
        errecho " Process received SIGINT."
    elif [ "$err_code" == 252 ]; then
        errecho " Command syntax invalid."
    elif [ "$err_code" == 253 ]; then
        errecho " The system environment or configuration was invalid."
    elif [ "$err_code" == 254 ]; then
        errecho " The service returned an error."
    elif [ "$err_code" == 255 ]; then
        errecho " 255 is a catch-all error."
    fi

    return 0
}

```

- Para API obtener más información, consulte [StopInstances](#) la Referencia de AWS CLI comandos.

## TerminateInstances

En el siguiente ejemplo de código, se muestra cómo usar TerminateInstances.



## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function ec2_terminate_instances
#
# This function terminates one or more Amazon Elastic Compute Cloud (Amazon EC2)
# instances using the AWS CLI.
#
# Parameters:
#     -i instance_ids - A space-separated list of instance IDs.
#     -h - Display help.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function ec2_terminate_instances() {
    local instance_ids response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function ec2_terminate_instances"
        echo "Terminates one or more Amazon Elastic Compute Cloud (Amazon EC2)
instances."
        echo "  -i instance_ids - A space-separated list of instance IDs."
        echo "  -h - Display help."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "i:h" option; do
        case "${option}" in
            i) instance_ids="${OPTARG}" ;;
            h)
                usage

```

```

        return 0
        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

# Check if instance ID is provided
if [[ -z "${instance_ids}" ]]; then
    echo "Error: Missing required instance IDs parameter."
    usage
    return 1
fi

# shellcheck disable=SC2086
response=$(aws ec2 terminate-instances \
    "--instance-ids" $instance_ids \
    --query 'TerminatingInstances[*].[InstanceId,CurrentState.Name]' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports terminate-instances operation failed.$response"
    return 1
}

return 0
}

```

Las funciones de utilidad utilizadas en este ejemplo.

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

```

```
#####  
# function aws_cli_error_log()  
#  
# This function is used to log the error messages from the AWS CLI.  
#  
# The function expects the following argument:  
#     $1 - The error code returned by the AWS CLI.  
#  
# Returns:  
#     0: - Success.  
#  
#####  
function aws_cli_error_log() {  
    local err_code=$1  
    errecho "Error code : $err_code"  
    if [ "$err_code" == 1 ]; then  
        errecho " One or more S3 transfers failed."  
    elif [ "$err_code" == 2 ]; then  
        errecho " Command line failed to parse."  
    elif [ "$err_code" == 130 ]; then  
        errecho " Process received SIGINT."  
    elif [ "$err_code" == 252 ]; then  
        errecho " Command syntax invalid."  
    elif [ "$err_code" == 253 ]; then  
        errecho " The system environment or configuration was invalid."  
    elif [ "$err_code" == 254 ]; then  
        errecho " The service returned an error."  
    elif [ "$err_code" == 255 ]; then  
        errecho " 255 is a catch-all error."  
    fi  
  
    return 0  
}
```

- Para API obtener más información, consulte [TerminateInstances](#) la Referencia de AWS CLI comandos.

## HealthImaging ejemplos de uso AWS CLI con el script Bash

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del script AWS Command Line Interface with Bash with HealthImaging

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Acciones](#)

## Acciones

### CreateDatastore

En el siguiente ejemplo de código, se muestra cómo usar CreateDatastore.

#### AWS CLI con el script Bash

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function imaging_create_datastore
#
# This function creates an AWS HealthImaging data store for importing DICOM P10
# files.
#
# Parameters:
#     -n data_store_name - The name of the data store.
#
# Returns:
#     The datastore ID.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
```

```
function imaging_create_datastore() {
    local datastore_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function imaging_create_datastore"
        echo "Creates an AWS HealthImaging data store for importing DICOM P10 files."
        echo "  -n data_store_name - The name of the data store."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:h" option; do
        case "${option}" in
            n) datastore_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$datastore_name" ]]; then
        errecho "ERROR: You must provide a data store name with the -n parameter."
        usage
        return 1
    fi

    response=$(aws medical-imaging create-datastore \
        --datastore-name "$datastore_name" \
        --output text \
        --query 'datastoreId')

    local error_code=${?}

    if [[ $error_code -ne 0 ]]; then
        aws_cli_error_log $error_code
    fi
}
```

```

    errecho "ERROR: AWS reports medical-imaging create-datastore operation failed.
$response"
    return 1
fi

echo "$response"

return 0
}

```

- Para API obtener más información, consulte [CreateDatastore](#) la Referencia de AWS CLI comandos.

### Note

Hay más información GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

## DeleteDatastore

En el siguiente ejemplo de código, se muestra cómo usar DeleteDatastore.

AWS CLI con el script Bash

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function imaging_delete_datastore
#
# This function deletes an AWS HealthImaging data store.
#
# Parameters:
#     -i datastore_id - The ID of the data store.

```

```

#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function imaging_delete_datastore() {
    local datastore_id response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function imaging_delete_datastore"
        echo "Deletes an AWS HealthImaging data store."
        echo "  -i datastore_id - The ID of the data store."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "i:h" option; do
        case "${option}" in
            i) datastore_id="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$datastore_id" ]]; then
        errecho "ERROR: You must provide a data store ID with the -i parameter."
        usage
        return 1
    fi

    response=$(aws medical-imaging delete-datastore \
        --datastore-id "$datastore_id")

    local error_code=${?}

```

```

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports medical-imaging delete-datastore operation failed.
$response"
    return 1
fi

return 0
}

```

- Para API obtener más información, consulte [DeleteDatastore](#) la Referencia de AWS CLI comandos.

### Note

Hay más información GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

## GetDatastore

En el siguiente ejemplo de código, se muestra cómo usar GetDatastore.

AWS CLI con el script Bash

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function imaging_get_datastore
#
# Get a data store's properties.
#
# Parameters:

```



```

#     -i data_store_id - The ID of the data store.
#
# Returns:
#     [datastore_name, datastore_id, datastore_status, datastore_arn,  created_at,
updated_at]
#     And:
#     0 - If successful.
#     1 - If it fails.
#####
function imaging_get_datastore() {
    local datastore_id option OPTARG # Required to use getopt command in a function.
    local error_code
    # bashsupport disable=BP5008
    function usage() {
        echo "function imaging_get_datastore"
        echo "Gets a data store's properties."
        echo "  -i datastore_id - The ID of the data store."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "i:h" option; do
        case "${option}" in
            i) datastore_id="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$datastore_id" ]]; then
        errecho "ERROR: You must provide a data store ID with the -i parameter."
        usage
        return 1
    fi

    local response

```

```

response=$(
  aws medical-imaging get-datastore \
    --datastore-id "$datastore_id" \
    --output text \
    --query "[ datastoreProperties.datastoreName,
datastoreProperties.datastoreId, datastoreProperties.datastoreStatus,
datastoreProperties.datastoreArn,  datastoreProperties.createdAt,
datastoreProperties.updatedAt]"
)
error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports list-datastores operation failed.$response"
  return 1
fi

echo "$response"

return 0
}

```

- Para API obtener más información, consulte [GetDatastore](#) la Referencia de AWS CLI comandos.

### Note

Hay más información GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

## ListDatastores

En el siguiente ejemplo de código, se muestra cómo usar ListDatastores.

AWS CLI con el script Bash

```

#####
# function errecho
#

```

```

# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function imaging_list_datastores
#
# List the HealthImaging data stores in the account.
#
# Returns:
#     [[datastore_name, datastore_id, datastore_status]]
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function imaging_list_datastores() {
    local option OPTARG # Required to use getopt command in a function.
    local error_code
    # bashsupport disable=BP5008
    function usage() {
        echo "function imaging_list_datastores"
        echo "Lists the AWS HealthImaging data stores in the account."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "h" option; do
        case "${option}" in
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    local response

```

```
response=$(aws medical-imaging list-datastores \
  --output text \
  --query "datastoreSummaries[*][datastoreName, datastoreId, datastoreStatus]")
error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports list-datastores operation failed.$response"
  return 1
fi

echo "$response"

return 0
}
```

- Para API obtener más información, consulte [ListDatastores](#) la Referencia de AWS CLI comandos.

#### Note

Hay más información GitHub. Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

## IAEjemplos de uso AWS CLI con el script Bash

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del script AWS Command Line Interface with Bash with. IAM

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Los escenarios son ejemplos de código que muestran cómo llevar a cabo una tarea específica a través de llamadas a varias funciones dentro del servicio o combinado con otros Servicios de AWS.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

### Temas

- [Acciones](#)
- [Escenarios](#)

## Acciones

### AttachRolePolicy

En el siguiente ejemplo de código, se muestra cómo usar AttachRolePolicy.

AWS CLI con el script Bash

#### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####  
# function errecho  
#  
# This function outputs everything sent to it to STDERR (standard error output).  
#####  
function errecho() {  
    printf "%s\n" "$*" 1>&2  
}  
  
#####  
# function iam_attach_role_policy  
#  
# This function attaches an IAM policy to a role.  
#  
# Parameters:  
#     -n role_name -- The name of the IAM role.  
#     -p policy_ARN -- The IAM policy document ARN..  
#  
# Returns:  
#     0 - If successful.  
#     1 - If it fails.  
#####  
function iam_attach_role_policy() {  
    local role_name policy_arn response
```

```
local option OPTARG # Required to use getopt command in a function.

# bashsupport disable=BP5008
function usage() {
    echo "function iam_attach_role_policy"
    echo "Attaches an AWS Identity and Access Management (IAM) policy to an IAM
role."
    echo "  -n role_name    The name of the IAM role."
    echo "  -p policy_ARN -- The IAM policy document ARN."
    echo ""
}

# Retrieve the calling parameters.
while getopt "n:p:h" option; do
    case "${option}" in
        n) role_name="${OPTARG}" ;;
        p) policy_arn="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy ARN with the -p parameter."
    usage
    return 1
fi

response=$(aws iam attach-role-policy \
    --role-name "$role_name" \
```

```

    --policy-arn "$policy_arn")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports attach-role-policy operation failed.\n$response"
    return 1
fi

echo "$response"

return 0
}

```

- Para API obtener más información, consulte [AttachRolePolicy](#) la Referencia de AWS CLI comandos.

## CreateAccessKey

En el siguiente ejemplo de código, se muestra cómo usar CreateAccessKey.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####

```

```

# function iam_create_user_access_key
#
# This function creates an IAM access key for the specified user.
#
# Parameters:
#     -u user_name -- The name of the IAM user.
#     [-f file_name] -- The optional file name for the access key output.
#
# Returns:
#     [access_key_id access_key_secret]
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_create_user_access_key() {
    local user_name file_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_create_user_access_key"
        echo "Creates an AWS Identity and Access Management (IAM) key pair."
        echo "  -u user_name   The name of the IAM user."
        echo "  [-f file_name] Optional file name for the access key output."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "u:f:h" option; do
        case "${option}" in
            u) user_name="${OPTARG}" ;;
            f) file_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

```



```
if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
fi

response=$(aws iam create-access-key \
    --user-name "$user_name" \
    --output text)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-access-key operation failed.$response"
    return 1
fi

if [[ -n "$file_name" ]]; then
    echo "$response" >"$file_name"
fi

local key_id key_secret
# shellcheck disable=SC2086
key_id=$(echo $response | cut -f 2 -d ' ')
# shellcheck disable=SC2086
key_secret=$(echo $response | cut -f 4 -d ' ')

echo "$key_id $key_secret"

return 0
}
```

- Para API obtener más información, consulte [CreateAccessKey](#) la Referencia de AWS CLI comandos.

## CreatePolicy

En el siguiente ejemplo de código, se muestra cómo usar CreatePolicy.

## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_create_policy
#
# This function creates an IAM policy.
#
# Parameters:
#     -n policy_name -- The name of the IAM policy.
#     -p policy_json -- The policy document.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_create_policy() {
    local policy_name policy_document response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_create_policy"
        echo "Creates an AWS Identity and Access Management (IAM) policy."
        echo "  -n policy_name  The name of the IAM policy."
        echo "  -p policy_json -- The policy document."
        echo ""
    }
}
```

```
# Retrieve the calling parameters.
while getopts "n:p:h" option; do
  case "${option}" in
    n) policy_name="${OPTARG}" ;;
    p) policy_document="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$policy_name" ]]; then
  errecho "ERROR: You must provide a policy name with the -n parameter."
  usage
  return 1
fi

if [[ -z "$policy_document" ]]; then
  errecho "ERROR: You must provide a policy document with the -p parameter."
  usage
  return 1
fi

response=$(aws iam create-policy \
  --policy-name "$policy_name" \
  --policy-document "$policy_document" \
  --output text \
  --query Policy.Arn)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports create-policy operation failed.\n$response"
  return 1
fi
```

```
    echo "$response"
  }
```

- Para API obtener más información, consulte [CreatePolicy](#) la Referencia de AWS CLI comandos.

## CreateRole

En el siguiente ejemplo de código, se muestra cómo usar CreateRole.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_create_role
#
# This function creates an IAM role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#     -p policy_json -- The assume role policy document.
#
# Returns:
#     The ARN of the role.
#
# And:
#     0 - If successful.
#     1 - If it fails.
```

```
#####  
function iam_create_role() {  
    local role_name policy_document response  
    local option OPTARG # Required to use getopt command in a function.  
  
    # bashsupport disable=BP5008  
    function usage() {  
        echo "function iam_create_user_access_key"  
        echo "Creates an AWS Identity and Access Management (IAM) role."  
        echo "  -n role_name    The name of the IAM role."  
        echo "  -p policy_json -- The assume role policy document."  
        echo ""  
    }  
  
    # Retrieve the calling parameters.  
    while getopt "n:p:h" option; do  
        case "${option}" in  
            n) role_name="${OPTARG}" ;;  
            p) policy_document="${OPTARG}" ;;  
            h)  
                usage  
                return 0  
                ;;  
            \?)  
                echo "Invalid parameter"  
                usage  
                return 1  
                ;;  
        esac  
    done  
    export OPTIND=1  
  
    if [[ -z "$role_name" ]]; then  
        errecho "ERROR: You must provide a role name with the -n parameter."  
        usage  
        return 1  
    fi  
  
    if [[ -z "$policy_document" ]]; then  
        errecho "ERROR: You must provide a policy document with the -p parameter."  
        usage  
        return 1  
    fi  
}
```

```

response=$(aws iam create-role \
  --role-name "$role_name" \
  --assume-role-policy-document "$policy_document" \
  --output text \
  --query Role.Arn)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports create-role operation failed.\n$response"
  return 1
fi

echo "$response"

return 0
}

```

- Para API obtener más información, consulte [CreateRole](#) la Referencia de AWS CLI comandos.

## CreateUser

En el siguiente ejemplo de código, se muestra cómo usar CreateUser.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
  if [[ $VERBOSE == true ]]; then

```

```

    echo "$@"
  fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#####
# function iam_create_user
#
# This function creates the specified IAM user, unless
# it already exists.
#
# Parameters:
#   -u user_name  -- The name of the user to create.
#
# Returns:
#   The ARN of the user.
#
# And:
#   0 - If successful.
#   1 - If it fails.
#####
function iam_create_user() {
  local user_name response
  local option OPTARG # Required to use getopt command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_create_user"
    echo "Creates an WS Identity and Access Management (IAM) user. You must supply a
username:"
    echo "  -u user_name    The name of the user. It must be unique within the
account."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopt "u:h" option; do

```

```
case "${option}" in
  u) user_name="${OPTARG}" ;;
  h)
    usage
    return 0
    ;;
  \?)
    echo "Invalid parameter"
    usage
    return 1
    ;;
esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
  errecho "ERROR: You must provide a username with the -u parameter."
  usage
  return 1
fi

iecho "Parameters:\n"
iecho "  User name:  $user_name"
iecho ""

# If the user already exists, we don't want to try to create it.
if (iam_user_exists "$user_name"); then
  errecho "ERROR: A user with that name already exists in the account."
  return 1
fi

response=$(aws iam create-user --user-name "$user_name" \
  --output text \
  --query 'User.Arn')

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports create-user operation failed.$response"
  return 1
fi

echo "$response"
```



```

    return 0
}

```

- Para API obtener más información, consulte [CreateUser](#) la Referencia de AWS CLI comandos.

## DeleteAccessKey

En el siguiente ejemplo de código, se muestra cómo usar DeleteAccessKey.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_delete_access_key
#
# This function deletes an IAM access key for the specified IAM user.
#
# Parameters:
#     -u user_name -- The name of the user.
#     -k access_key -- The access key to delete.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_access_key() {

```

```
local user_name access_key response
local option OPTARG # Required to use getopt command in a function.

# bashsupport disable=BP5008
function usage() {
    echo "function iam_delete_access_key"
    echo "Deletes an WS Identity and Access Management (IAM) access key for the
specified IAM user"
    echo "  -u user_name    The name of the user."
    echo "  -k access_key    The access key to delete."
    echo ""
}

# Retrieve the calling parameters.
while getopt "u:k:h" option; do
    case "${option}" in
        u) user_name="${OPTARG}" ;;
        k) access_key="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
fi

if [[ -z "$access_key" ]]; then
    errecho "ERROR: You must provide an access key with the -k parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
```

```

iecho "    Username:  $user_name"
iecho "    Access key:  $access_key"
iecho ""

response=$(aws iam delete-access-key \
  --user-name "$user_name" \
  --access-key-id "$access_key")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports delete-access-key operation failed.\n$response"
  return 1
fi

iecho "delete-access-key response:$response"
iecho

return 0
}

```

- Para API obtener más información, consulte [DeleteAccessKey](#) la Referencia de AWS CLI comandos.

## DeletePolicy

En el siguiente ejemplo de código, se muestra cómo usar DeletePolicy.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function iecho
#

```

```
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_delete_policy
#
# This function deletes an IAM policy.
#
# Parameters:
#     -n policy_arn -- The name of the IAM policy arn.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_policy() {
    local policy_arn response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_delete_policy"
        echo "Deletes an WS Identity and Access Management (IAM) policy"
        echo "  -n policy_arn -- The name of the IAM policy arn."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:h" option; do
```

```
case "${option}" in
  n) policy_arn="${OPTARG}" ;;
  h)
    usage
    return 0
    ;;
  \?)
    echo "Invalid parameter"
    usage
    return 1
    ;;
esac
done
export OPTIND=1

if [[ -z "$policy_arn" ]]; then
  errecho "ERROR: You must provide a policy arn with the -n parameter."
  usage
  return 1
fi

iecho "Parameters:\n"
iecho "  Policy arn: $policy_arn"
iecho ""

response=$(aws iam delete-policy \
  --policy-arn "$policy_arn")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports delete-policy operation failed.\n$response"
  return 1
fi

iecho "delete-policy response:$response"
iecho

return 0
}
```

- Para API obtener más información, consulte [DeletePolicy](#) la Referencia de AWS CLI comandos.

## DeleteRole

En el siguiente ejemplo de código, se muestra cómo usar DeleteRole.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_delete_role
#
# This function deletes an IAM role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#
# Returns:
#     0 - If successful.
```

```

#      1 - If it fails.
#####
function iam_delete_role() {
    local role_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_delete_role"
        echo "Deletes an WS Identity and Access Management (IAM) role"
        echo "  -n role_name -- The name of the IAM role."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:h" option; do
        case "${option}" in
            n) role_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    echo "role_name:$role_name"
    if [[ -z "$role_name" ]]; then
        errecho "ERROR: You must provide a role name with the -n parameter."
        usage
        return 1
    fi

    iecho "Parameters:\n"
    iecho "  Role name: $role_name"
    iecho ""

    response=$(aws iam delete-role \
        --role-name "$role_name")

```

```

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-role operation failed.\n$response"
    return 1
fi

iecho "delete-role response:$response"
iecho

return 0
}

```

- Para API obtener más información, consulte [DeleteRole](#) la Referencia de AWS CLI comandos.

## DeleteUser

En el siguiente ejemplo de código, se muestra cómo usar DeleteUser.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

```



```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_delete_user
#
# This function deletes the specified IAM user.
#
# Parameters:
#     -u user_name  -- The name of the user to create.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_user() {
    local user_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_delete_user"
        echo "Deletes an WS Identity and Access Management (IAM) user. You must supply a
username:"
        echo "  -u user_name    The name of the user."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "u:h" option; do
        case "${option}" in
            u) user_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"

```

```
        usage
        return 1
    ;;
esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    User name:    $user_name"
iecho ""

# If the user does not exist, we don't want to try to delete it.
if (! iam_user_exists "$user_name"); then
    errecho "ERROR: A user with that name does not exist in the account."
    return 1
fi

response=$(aws iam delete-user \
    --user-name "$user_name")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-user operation failed.$response"
    return 1
fi

iecho "delete-user response:$response"
iecho

return 0
}
```

- Para API obtener más información, consulte [DeleteUser](#) la Referencia de AWS CLI comandos.

## DetachRolePolicy

En el siguiente ejemplo de código, se muestra cómo usar DetachRolePolicy.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_detach_role_policy
#
# This function detaches an IAM policy to a role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#     -p policy_ARN -- The IAM policy document ARN..
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_detach_role_policy() {
    local role_name policy_arn response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_detach_role_policy"
        echo "Detaches an AWS Identity and Access Management (IAM) policy to an IAM
role."
    }
}
```

```
    echo " -n role_name The name of the IAM role."
    echo " -p policy_ARN -- The IAM policy document ARN."
    echo ""
}

# Retrieve the calling parameters.
while getopts "n:p:h" option; do
    case "${option}" in
        n) role_name="${OPTARG}" ;;
        p) policy_arn="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy ARN with the -p parameter."
    usage
    return 1
fi

response=$(aws iam detach-role-policy \
    --role-name "$role_name" \
    --policy-arn "$policy_arn")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports detach-role-policy operation failed.\n$response"
```

```

    return 1
fi

echo "$response"

return 0
}

```

- Para API obtener más información, consulte [DetachRolePolicy](#) la Referencia de AWS CLI comandos.

## GetUser

En el siguiente ejemplo de código, se muestra cómo usar GetUser.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_user_exists
#
# This function checks to see if the specified AWS Identity and Access Management
# (IAM) user already exists.
#
# Parameters:
#     $1 - The name of the IAM user to check.

```

```
#
# Returns:
#     0 - If the user already exists.
#     1 - If the user doesn't exist.
#####
function iam_user_exists() {
    local user_name
    user_name=$1

    # Check whether the IAM user already exists.
    # We suppress all output - we're interested only in the return code.

    local errors
    errors=$(aws iam get-user \
        --user-name "$user_name" 2>&1 >/dev/null)

    local error_code=${?}

    if [[ $error_code -eq 0 ]]; then
        return 0 # 0 in Bash script means true.
    else
        if [[ $errors != *"error"*(NoSuchEntity)* ]]; then
            aws_cli_error_log $error_code
            errecho "Error calling iam get-user $errors"
        fi

        return 1 # 1 in Bash script means false.
    fi
}
```

- Para API obtener más información, consulte [GetUser](#) la Referencia de AWS CLI comandos.

## ListAccessKeys

En el siguiente ejemplo de código, se muestra cómo usar ListAccessKeys.

## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_list_access_keys
#
# This function lists the access keys for the specified user.
#
# Parameters:
#     -u user_name -- The name of the IAM user.
#
# Returns:
#     access_key_ids
#     And:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_list_access_keys() {

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_list_access_keys"
        echo "Lists the AWS Identity and Access Management (IAM) access key IDs for the
specified user."
        echo "  -u user_name  The name of the IAM user."
        echo ""
    }
}
```

```
local user_name response
local option OPTARG # Required to use getopt command in a function.
# Retrieve the calling parameters.
while getopt "u:h" option; do
  case "${option}" in
    u) user_name="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
  errecho "ERROR: You must provide a username with the -u parameter."
  usage
  return 1
fi

response=$(aws iam list-access-keys \
  --user-name "$user_name" \
  --output text \
  --query 'AccessKeyMetadata[].AccessKeyId')

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports list-access-keys operation failed.$response"
  return 1
fi

echo "$response"

return 0
}
```



- Para API obtener más información, consulte [ListAccessKeys](#) la Referencia de AWS CLI comandos.

## ListUsers

En el siguiente ejemplo de código, se muestra cómo usar ListUsers.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function iam_list_users
#
# List the IAM users in the account.
#
# Returns:
#     The list of users names
#     And:
#     0 - If the user already exists.
#     1 - If the user doesn't exist.
#####
function iam_list_users() {
    local option OPTARG # Required to use getopt command in a function.
    local error_code
    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_list_users"
        echo "Lists the AWS Identity and Access Management (IAM) user in the account."
    }
}
```

```
    echo ""
}

# Retrieve the calling parameters.
while getopts "h" option; do
    case "${option}" in
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

local response

response=$(aws iam list-users \
    --output text \
    --query "Users[].UserName")
error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports list-users operation failed.$response"
    return 1
fi

echo "$response"

return 0
}
```

- Para API obtener más información, consulte [ListUsers](#) la Referencia de AWS CLI comandos.

## Escenarios

### Crear un usuario y asumir un rol

En el siguiente ejemplo de código, se muestra cómo crear un usuario y asumir un rol.

#### Warning

Para evitar riesgos de seguridad, no utilice a IAM los usuarios para autenticarse cuando desarrolle software diseñado específicamente o trabaje con datos reales. En cambio, utilice la federación con un proveedor de identidades como [AWS IAM Identity Center](#).

- Crear un usuario que no tenga permisos.
- Crear un rol que conceda permiso para enumerar los buckets de Amazon S3 para la cuenta.
- Agregar una política para que el usuario asuma el rol.
- Asumir el rol y enumerar los buckets de S3 con credenciales temporales, y después limpiar los recursos.

### AWS CLI con el script Bash

#### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####  
# function iam_create_user_assume_role  
#  
# Scenario to create an IAM user, create an IAM role, and apply the role to the  
# user.  
#  
# "IAM access" permissions are needed to run this code.  
# "STS assume role" permissions are needed to run this code. (Note: It might be  
# necessary to  
# create a custom policy).  
#  
# Returns:
```

```
#      0 - If successful.
#      1 - If an error occurred.
#####
function iam_create_user_assume_role() {
{
    if [ "$IAM_OPERATIONS_SOURCED" != "True" ]; then

        source ./iam_operations.sh
    fi
}

echo_repeat "*" 88
echo "Welcome to the IAM create user and assume role demo."
echo
echo "This demo will create an IAM user, create an IAM role, and apply the role to
the user."
echo_repeat "*" 88
echo

echo -n "Enter a name for a new IAM user: "
get_input
user_name=$get_input_result

local user_arn
user_arn=$(iam_create_user -u "$user_name")

# shellcheck disable=SC2181
if [[ ${?} == 0 ]]; then
    echo "Created demo IAM user named $user_name"
else
    errecho "$user_arn"
    errecho "The user failed to create. This demo will exit."
    return 1
fi

local access_key_response
access_key_response=$(iam_create_user_access_key -u "$user_name")
# shellcheck disable=SC2181
if [[ ${?} != 0 ]]; then
    errecho "The access key failed to create. This demo will exit."
    clean_up "$user_name"
    return 1
fi
```

```
IFS=$'\t ' read -r -a access_key_values <<<"$access_key_response"
local key_name=${access_key_values[0]}
local key_secret=${access_key_values[1]}

echo "Created access key named $key_name"

echo "Wait 10 seconds for the user to be ready."
sleep 10
echo_repeat "*" 88
echo

local iam_role_name
iam_role_name=$(generate_random_name "test-role")
echo "Creating a role named $iam_role_name with user $user_name as the principal."

local assume_role_policy_document="{
  \"Version\": \"2012-10-17\",
  \"Statement\": [{
    \"Effect\": \"Allow\",
    \"Principal\": {\"AWS\": \"$user_arn\"},
    \"Action\": \"sts:AssumeRole\"
  }]
}"

local role_arn
role_arn=$(iam_create_role -n "$iam_role_name" -p "$assume_role_policy_document")

# shellcheck disable=SC2181
if [ ${?} == 0 ]; then
  echo "Created IAM role named $iam_role_name"
else
  errecho "The role failed to create. This demo will exit."
  clean_up "$user_name" "$key_name"
  return 1
fi

local policy_name
policy_name=$(generate_random_name "test-policy")
local policy_document="{
  \"Version\": \"2012-10-17\",
  \"Statement\": [{
    \"Effect\": \"Allow\",
    \"Action\": \"s3:ListAllMyBuckets\",
    \"Resource\": \"arn:aws:s3:::*\"}]}"
```

```
local policy_arn
policy_arn=$(iam_create_policy -n "$policy_name" -p "$policy_document")
# shellcheck disable=SC2181
if [[ ${?} == 0 ]]; then
    echo "Created IAM policy named $policy_name"
else
    errecho "The policy failed to create."
    clean_up "$user_name" "$key_name" "$iam_role_name"
    return 1
fi

if (iam_attach_role_policy -n "$iam_role_name" -p "$policy_arn"); then
    echo "Attached policy $policy_arn to role $iam_role_name"
else
    errecho "The policy failed to attach."
    clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn"
    return 1
fi

local assume_role_policy_document="{
    \"Version\": \"2012-10-17\",
    \"Statement\": [{
        \"Effect\": \"Allow\",
        \"Action\": \"sts:AssumeRole\",
        \"Resource\": \"$role_arn\"}]}"

local assume_role_policy_name
assume_role_policy_name=$(generate_random_name "test-assume-role-")

# shellcheck disable=SC2181
local assume_role_policy_arn
assume_role_policy_arn=$(iam_create_policy -n "$assume_role_policy_name" -p
"$assume_role_policy_document")
# shellcheck disable=SC2181
if [ ${?} == 0 ]; then
    echo "Created IAM policy named $assume_role_policy_name for sts assume role"
else
    errecho "The policy failed to create."
    clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn" "$policy_arn"
    return 1
fi
```

```
echo "Wait 10 seconds to give AWS time to propagate these new resources and
connections."
sleep 10
echo_repeat "*" 88
echo

echo "Try to list buckets without the new user assuming the role."
echo_repeat "*" 88
echo

# Set the environment variables for the created user.
# bashsupport disable=BP2001
export AWS_ACCESS_KEY_ID=$key_name
# bashsupport disable=BP2001
export AWS_SECRET_ACCESS_KEY=$key_secret

local buckets
buckets=$(s3_list_buckets)

# shellcheck disable=SC2181
if [ ${?} == 0 ]; then
    local bucket_count
    bucket_count=$(echo "$buckets" | wc -w | xargs)
    echo "There are $bucket_count buckets in the account. This should not have
happened."
else
    errecho "Because the role with permissions has not been assumed, listing buckets
failed."
fi

echo
echo_repeat "*" 88
echo "Now assume the role $iam_role_name and list the buckets."
echo_repeat "*" 88
echo

local credentials

credentials=$(sts_assume_role -r "$role_arn" -n "AssumeRoleDemoSession")
# shellcheck disable=SC2181
if [ ${?} == 0 ]; then
    echo "Assumed role $iam_role_name"
else
    errecho "Failed to assume role."
```

```
    export AWS_ACCESS_KEY_ID=""
    export AWS_SECRET_ACCESS_KEY=""
    clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn" "$policy_arn"
"$assume_role_policy_arn"
    return 1
fi

IFS=$'\t ' read -r -a credentials <<<"$credentials"

export AWS_ACCESS_KEY_ID=${credentials[0]}
export AWS_SECRET_ACCESS_KEY=${credentials[1]}
# bashsupport disable=BP2001
export AWS_SESSION_TOKEN=${credentials[2]}

buckets=$(s3_list_buckets)

# shellcheck disable=SC2181
if [ ${?} == 0 ]; then
    local bucket_count
    bucket_count=$(echo "$buckets" | wc -w | xargs)
    echo "There are $bucket_count buckets in the account. Listing buckets succeeded
because of "
    echo "the assumed role."
else
    errecho "Failed to list buckets. This should not happen."
    export AWS_ACCESS_KEY_ID=""
    export AWS_SECRET_ACCESS_KEY=""
    export AWS_SESSION_TOKEN=""
    clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn" "$policy_arn"
"$assume_role_policy_arn"
    return 1
fi

local result=0
export AWS_ACCESS_KEY_ID=""
export AWS_SECRET_ACCESS_KEY=""

echo
echo_repeat "*" 88
echo "The created resources will now be deleted."
echo_repeat "*" 88
echo
```



```

clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn" "$policy_arn"
"$assume_role_policy_arn"

# shellcheck disable=SC2181
if [[ ${?} -ne 0 ]]; then
    result=1
fi

return $result
}

```

Las IAM funciones utilizadas en este escenario.

```

#####
# function iam_user_exists
#
# This function checks to see if the specified AWS Identity and Access Management
# (IAM) user already exists.
#
# Parameters:
#     $1 - The name of the IAM user to check.
#
# Returns:
#     0 - If the user already exists.
#     1 - If the user doesn't exist.
#####
function iam_user_exists() {
    local user_name
    user_name=$1

    # Check whether the IAM user already exists.
    # We suppress all output - we're interested only in the return code.

    local errors
    errors=$(aws iam get-user \
        --user-name "$user_name" 2>&1 >/dev/null)

    local error_code=${?}

    if [[ $error_code -eq 0 ]]; then
        return 0 # 0 in Bash script means true.
    else

```

```

    if [[ $errors != *"error"*(NoSuchEntity)* ]]; then
        aws_cli_error_log $error_code
        errecho "Error calling iam get-user $errors"
    fi

    return 1 # 1 in Bash script means false.
fi
}

#####
# function iam_create_user
#
# This function creates the specified IAM user, unless
# it already exists.
#
# Parameters:
#     -u user_name  -- The name of the user to create.
#
# Returns:
#     The ARN of the user.
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_create_user() {
    local user_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_create_user"
        echo "Creates an WS Identity and Access Management (IAM) user. You must supply a
username:"
        echo "  -u user_name    The name of the user. It must be unique within the
account."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "u:h" option; do
        case "${option}" in
            u) user_name="${OPTARG}" ;;
            h)
                usage

```

```
        return 0
        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    User name:    $user_name"
iecho ""

# If the user already exists, we don't want to try to create it.
if (iam_user_exists "$user_name"); then
    errecho "ERROR: A user with that name already exists in the account."
    return 1
fi

response=$(aws iam create-user --user-name "$user_name" \
    --output text \
    --query 'User.Arn')

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-user operation failed.$response"
    return 1
fi

echo "$response"

return 0
}
```

```
#####
# function iam_create_user_access_key
#
# This function creates an IAM access key for the specified user.
#
# Parameters:
#   -u user_name -- The name of the IAM user.
#   [-f file_name] -- The optional file name for the access key output.
#
# Returns:
#   [access_key_id access_key_secret]
#   And:
#   0 - If successful.
#   1 - If it fails.
#####
function iam_create_user_access_key() {
    local user_name file_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_create_user_access_key"
        echo "Creates an AWS Identity and Access Management (IAM) key pair."
        echo "  -u user_name   The name of the IAM user."
        echo "  [-f file_name] Optional file name for the access key output."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "u:f:h" option; do
        case "${option}" in
            u) user_name="${OPTARG}" ;;
            f) file_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
}
```

```

export OPTIND=1

if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
fi

response=$(aws iam create-access-key \
    --user-name "$user_name" \
    --output text)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-access-key operation failed.$response"
    return 1
fi

if [[ -n "$file_name" ]]; then
    echo "$response" >"$file_name"
fi

local key_id key_secret
# shellcheck disable=SC2086
key_id=$(echo $response | cut -f 2 -d ' ')
# shellcheck disable=SC2086
key_secret=$(echo $response | cut -f 4 -d ' ')

echo "$key_id $key_secret"

return 0
}

#####
# function iam_create_role
#
# This function creates an IAM role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#     -p policy_json -- The assume role policy document.
#

```

```
# Returns:
#     The ARN of the role.
#     And:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_create_role() {
    local role_name policy_document response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_create_user_access_key"
        echo "Creates an AWS Identity and Access Management (IAM) role."
        echo "  -n role_name    The name of the IAM role."
        echo "  -p policy_json -- The assume role policy document."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:p:h" option; do
        case "${option}" in
            n) role_name="${OPTARG}" ;;
            p) policy_document="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$role_name" ]]; then
        errecho "ERROR: You must provide a role name with the -n parameter."
        usage
        return 1
    fi

    if [[ -z "$policy_document" ]]; then
```

```

    errecho "ERROR: You must provide a policy document with the -p parameter."
    usage
    return 1
fi

response=$(aws iam create-role \
  --role-name "$role_name" \
  --assume-role-policy-document "$policy_document" \
  --output text \
  --query Role.Arn)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-role operation failed.\n$response"
    return 1
fi

echo "$response"

return 0
}

#####
# function iam_create_policy
#
# This function creates an IAM policy.
#
# Parameters:
#   -n policy_name -- The name of the IAM policy.
#   -p policy_json -- The policy document.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
function iam_create_policy() {
    local policy_name policy_document response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_create_policy"
    }

```

```
    echo "Creates an AWS Identity and Access Management (IAM) policy."
    echo "  -n policy_name  The name of the IAM policy."
    echo "  -p policy_json -- The policy document."
    echo ""
}

# Retrieve the calling parameters.
while getopts "n:p:h" option; do
    case "${option}" in
        n) policy_name="${OPTARG}" ;;
        p) policy_document="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$policy_name" ]]; then
    errecho "ERROR: You must provide a policy name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$policy_document" ]]; then
    errecho "ERROR: You must provide a policy document with the -p parameter."
    usage
    return 1
fi

response=$(aws iam create-policy \
    --policy-name "$policy_name" \
    --policy-document "$policy_document" \
    --output text \
    --query Policy.Arn)

local error_code=${?}
```



```

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-policy operation failed.\n$response"
    return 1
fi

echo "$response"
}

#####
# function iam_attach_role_policy
#
# This function attaches an IAM policy to a role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#     -p policy_ARN -- The IAM policy document ARN..
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_attach_role_policy() {
    local role_name policy_arn response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_attach_role_policy"
        echo "Attaches an AWS Identity and Access Management (IAM) policy to an IAM
role."
        echo "  -n role_name    The name of the IAM role."
        echo "  -p policy_ARN -- The IAM policy document ARN."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:p:h" option; do
        case "${option}" in
            n) role_name="${OPTARG}" ;;
            p) policy_arn="${OPTARG}" ;;
            h)
                usage
                return 0
        esac
    done
}

```

```

        ;;
    \?)
        echo "Invalid parameter"
        usage
        return 1
    ;;
esac
done
export OPTIND=1

if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
fi

if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy ARN with the -p parameter."
    usage
    return 1
fi

response=$(aws iam attach-role-policy \
    --role-name "$role_name" \
    --policy-arn "$policy_arn")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports attach-role-policy operation failed.\n$response"
    return 1
fi

echo "$response"

return 0
}

#####
# function iam_detach_role_policy
#
# This function detaches an IAM policy to a role.
#

```

```

# Parameters:
#     -n role_name -- The name of the IAM role.
#     -p policy_ARN -- The IAM policy document ARN..
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_detach_role_policy() {
    local role_name policy_arn response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_detach_role_policy"
        echo "Detaches an AWS Identity and Access Management (IAM) policy to an IAM
role."
        echo "  -n role_name    The name of the IAM role."
        echo "  -p policy_ARN -- The IAM policy document ARN."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:p:h" option; do
        case "${option}" in
            n) role_name="${OPTARG}" ;;
            p) policy_arn="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
    export OPTIND=1

    if [[ -z "$role_name" ]]; then
        errecho "ERROR: You must provide a role name with the -n parameter."
        usage
        return 1
    fi
}

```

```

fi

if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy ARN with the -p parameter."
    usage
    return 1
fi

response=$(aws iam detach-role-policy \
    --role-name "$role_name" \
    --policy-arn "$policy_arn")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports detach-role-policy operation failed.\n$response"
    return 1
fi

echo "$response"

return 0
}

#####
# function iam_delete_policy
#
# This function deletes an IAM policy.
#
# Parameters:
#     -n policy_arn -- The name of the IAM policy arn.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_policy() {
    local policy_arn response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_delete_policy"
    }
}

```

```
    echo "Deletes an WS Identity and Access Management (IAM) policy"
    echo "  -n policy_arn -- The name of the IAM policy arn."
    echo ""
}

# Retrieve the calling parameters.
while getopts "n:h" option; do
    case "${option}" in
        n) policy_arn="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy arn with the -n parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "  Policy arn: $policy_arn"
iecho ""

response=$(aws iam delete-policy \
    --policy-arn "$policy_arn")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-policy operation failed.\n$response"
    return 1
fi

iecho "delete-policy response:$response"
```

```
iecho

return 0
}

#####
# function iam_delete_role
#
# This function deletes an IAM role.
#
# Parameters:
#     -n role_name -- The name of the IAM role.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_role() {
    local role_name response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function iam_delete_role"
        echo "Deletes an WS Identity and Access Management (IAM) role"
        echo "  -n role_name -- The name of the IAM role."
        echo ""
    }

    # Retrieve the calling parameters.
    while getopt "n:h" option; do
        case "${option}" in
            n) role_name="${OPTARG}" ;;
            h)
                usage
                return 0
                ;;
            \?)
                echo "Invalid parameter"
                usage
                return 1
                ;;
        esac
    done
```

```

export OPTIND=1

echo "role_name:$role_name"
if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    Role name: $role_name"
iecho ""

response=$(aws iam delete-role \
    --role-name "$role_name")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-role operation failed.\n$response"
    return 1
fi

iecho "delete-role response:$response"
iecho

return 0
}

#####
# function iam_delete_access_key
#
# This function deletes an IAM access key for the specified IAM user.
#
# Parameters:
#     -u user_name -- The name of the user.
#     -k access_key -- The access key to delete.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function iam_delete_access_key() {

```

```
local user_name access_key response
local option OPTARG # Required to use getopt command in a function.

# bashsupport disable=BP5008
function usage() {
    echo "function iam_delete_access_key"
    echo "Deletes an WS Identity and Access Management (IAM) access key for the
specified IAM user"
    echo "  -u user_name    The name of the user."
    echo "  -k access_key    The access key to delete."
    echo ""
}

# Retrieve the calling parameters.
while getopt "u:k:h" option; do
    case "${option}" in
        u) user_name="${OPTARG}" ;;
        k) access_key="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
fi

if [[ -z "$access_key" ]]; then
    errecho "ERROR: You must provide an access key with the -k parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
```



```

iecho "    Username:  $user_name"
iecho "    Access key:  $access_key"
iecho ""

response=$(aws iam delete-access-key \
  --user-name "$user_name" \
  --access-key-id "$access_key")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports delete-access-key operation failed.\n$response"
  return 1
fi

iecho "delete-access-key response:$response"
iecho

return 0
}

#####
# function iam_delete_user
#
# This function deletes the specified IAM user.
#
# Parameters:
#   -u user_name  -- The name of the user to create.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
function iam_delete_user() {
  local user_name response
  local option OPTARG # Required to use getopt command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_delete_user"
    echo "Deletes an WS Identity and Access Management (IAM) user. You must supply a
username:"
    echo "  -u user_name    The name of the user."
  }
}

```

```
    echo ""
}

# Retrieve the calling parameters.
while getopts "u:h" option; do
    case "${option}" in
        u) user_name="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
fi

iecho "Parameters:\n"
iecho "    User name:  $user_name"
iecho ""

# If the user does not exist, we don't want to try to delete it.
if (! iam_user_exists "$user_name"); then
    errecho "ERROR: A user with that name does not exist in the account."
    return 1
fi

response=$(aws iam delete-user \
    --user-name "$user_name")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-user operation failed.$response"
```

```
    return 1
  fi

  iecho "delete-user response:$response"
  iecho

  return 0
}
```

- Para API obtener más información, consulte los siguientes temas en AWS CLI Command Reference.
  - [AttachRolePolicy](#)
  - [CreateAccessKey](#)
  - [CreatePolicy](#)
  - [CreateRole](#)
  - [CreateUser](#)
  - [DeleteAccessKey](#)
  - [DeletePolicy](#)
  - [DeleteRole](#)
  - [DeleteUser](#)
  - [DeleteUserPolicy](#)
  - [DetachRolePolicy](#)
  - [PutUserPolicy](#)

## Ejemplos de Amazon S3 que se utilizan AWS CLI con el script Bash

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del script AWS Command Line Interface with Bash con Amazon S3.

Los conceptos básicos son ejemplos de código que muestran cómo realizar las operaciones esenciales dentro de un servicio.

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

## Temas

- [Conceptos básicos](#)
- [Acciones](#)

## Conceptos básicos

Aprenda los conceptos básicos

En el siguiente ejemplo de código, se muestra cómo:

- Creación de un bucket y cargar un archivo en el bucket.
- Descargar un objeto desde un bucket.
- Copiar un objeto en una subcarpeta de un bucket.
- Obtención de una lista de los objetos de un bucket.
- Eliminación del bucket y todos los objetos que incluye.

## AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####  
# function s3_getting_started  
#  
# This function creates, copies, and deletes S3 buckets and objects.  
#  
# Returns:  
#     0 - If successful.  
#     1 - If an error occurred.  
#####  
function s3_getting_started() {  
  {
```

```
if [ "$BUCKET_OPERATIONS_SOURCED" != "True" ]; then
    cd bucket-lifecycle-operations || exit

    source ./bucket_operations.sh
    cd ..
fi
}

echo_repeat "*" 88
echo "Welcome to the Amazon S3 getting started demo."
echo_repeat "*" 88

local bucket_name
bucket_name=$(generate_random_name "doc-example-bucket")

local region_code
region_code=$(aws configure get region)

if create_bucket -b "$bucket_name" -r "$region_code"; then
    echo "Created demo bucket named $bucket_name"
else
    errecho "The bucket failed to create. This demo will exit."
    return 1
fi

local file_name
while [ -z "$file_name" ]; do
    echo -n "Enter a file you want to upload to your bucket: "
    get_input
    file_name=$get_input_result

    if [ ! -f "$file_name" ]; then
        echo "Could not find file $file_name. Are you sure it exists?"
        file_name=""
    fi
done

local key
key="$(basename "$file_name")"

local result=0
if copy_file_to_bucket "$bucket_name" "$file_name" "$key"; then
    echo "Uploaded file $file_name into bucket $bucket_name with key $key."
else
```

```
    result=1
  fi

  local destination_file
  destination_file="$file_name.download"
  if yes_no_input "Would you like to download $key to the file $destination_file?
(y/n) "; then
    if download_object_from_bucket "$bucket_name" "$destination_file" "$key"; then
      echo "Downloaded $key in the bucket $bucket_name to the file
$destination_file."
    else
      result=1
    fi
  fi
fi

  if yes_no_input "Would you like to copy $key a new object key in your bucket? (y/
n) "; then
    local to_key
    to_key="demo/$key"
    if copy_item_in_bucket "$bucket_name" "$key" "$to_key"; then
      echo "Copied $key in the bucket $bucket_name to the $to_key."
    else
      result=1
    fi
  fi
fi

  local bucket_items
  bucket_items=$(list_items_in_bucket "$bucket_name")

  # shellcheck disable=SC2181
  if [[ $? -ne 0 ]]; then
    result=1
  fi

  echo "Your bucket contains the following items."
  echo -e "Name\t\tSize"
  echo "$bucket_items"

  if yes_no_input "Delete the bucket, $bucket_name, as well as the objects in it?
(y/n) "; then
    bucket_items=$(echo "$bucket_items" | cut -f 1)

    if delete_items_in_bucket "$bucket_name" "$bucket_items"; then
      echo "The following items were deleted from the bucket $bucket_name"
```

```

    echo "$bucket_items"
else
    result=1
fi

if delete_bucket "$bucket_name"; then
    echo "Deleted the bucket $bucket_name"
else
    result=1
fi
fi

return $result
}

```

Las funciones de Amazon S3 utilizadas en este escenario.

```

#####
# function create-bucket
#
# This function creates the specified bucket in the specified AWS Region, unless
# it already exists.
#
# Parameters:
#     -b bucket_name  -- The name of the bucket to create.
#     -r region_code  -- The code for an AWS Region in which to
#                       create the bucket.
#
# Returns:
#     The URL of the bucket that was created.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function create_bucket() {
    local bucket_name region_code response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function create_bucket"
        echo "Creates an Amazon S3 bucket. You must supply a bucket name:"
    }
}

```

```
    echo "  -b bucket_name    The name of the bucket. It must be globally unique."
    echo "  [-r region_code]    The code for an AWS Region in which the bucket is
created."
    echo ""
}

# Retrieve the calling parameters.
while getopts "b:r:h" option; do
    case "${option}" in
        b) bucket_name="${OPTARG}" ;;
        r) region_code="${OPTARG}" ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done

if [[ -z "$bucket_name" ]]; then
    errecho "ERROR: You must provide a bucket name with the -b parameter."
    usage
    return 1
fi

local bucket_config_arg
# A location constraint for "us-east-1" returns an error.
if [[ -n "$region_code" ]] && [[ "$region_code" != "us-east-1" ]]; then
    bucket_config_arg="--create-bucket-configuration LocationConstraint=
$region_code"
fi

iecho "Parameters:\n"
iecho "  Bucket name:    $bucket_name"
iecho "  Region code:    $region_code"
iecho ""

# If the bucket already exists, we don't want to try to create it.
if (bucket_exists "$bucket_name"); then
    errecho "ERROR: A bucket with that name already exists. Try again."
```



```

    return 1
fi

# shellcheck disable=SC2086
response=$(aws s3api create-bucket \
  --bucket "$bucket_name" \
  $bucket_config_arg)

# shellcheck disable=SC2181
if [[ ${?} -ne 0 ]]; then
  errecho "ERROR: AWS reports create-bucket operation failed.\n$response"
  return 1
fi
}

#####
# function copy_file_to_bucket
#
# This function creates a file in the specified bucket.
#
# Parameters:
#   $1 - The name of the bucket to copy the file to.
#   $2 - The path and file name of the local file to copy to the bucket.
#   $3 - The key (name) to call the copy of the file in the bucket.
#
# Returns:
#   0 - If successful.
#   1 - If it fails.
#####
function copy_file_to_bucket() {
  local response bucket_name source_file destination_file_name
  bucket_name=$1
  source_file=$2
  destination_file_name=$3

  response=$(aws s3api put-object \
    --bucket "$bucket_name" \
    --body "$source_file" \
    --key "$destination_file_name")

# shellcheck disable=SC2181
if [[ ${?} -ne 0 ]]; then
  errecho "ERROR: AWS reports put-object operation failed.\n$response"
  return 1

```

```

    fi
}

#####
# function download_object_from_bucket
#
# This function downloads an object in a bucket to a file.
#
# Parameters:
#     $1 - The name of the bucket to download the object from.
#     $2 - The path and file name to store the downloaded bucket.
#     $3 - The key (name) of the object in the bucket.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function download_object_from_bucket() {
    local bucket_name=$1
    local destination_file_name=$2
    local object_name=$3
    local response

    response=$(aws s3api get-object \
        --bucket "$bucket_name" \
        --key "$object_name" \
        "$destination_file_name")

    # shellcheck disable=SC2181
    if [[ ${?} -ne 0 ]]; then
        errecho "ERROR: AWS reports put-object operation failed.\n$response"
        return 1
    fi
}

#####
# function copy_item_in_bucket
#
# This function creates a copy of the specified file in the same bucket.
#
# Parameters:
#     $1 - The name of the bucket to copy the file from and to.
#     $2 - The key of the source file to copy.
#     $3 - The key of the destination file.

```

```

#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function copy_item_in_bucket() {
    local bucket_name=$1
    local source_key=$2
    local destination_key=$3
    local response

    response=$(aws s3api copy-object \
        --bucket "$bucket_name" \
        --copy-source "$bucket_name/$source_key" \
        --key "$destination_key")

    # shellcheck disable=SC2181
    if [[ $? -ne 0 ]]; then
        errecho "ERROR: AWS reports s3api copy-object operation failed.\n$response"
        return 1
    fi
}

#####
# function list_items_in_bucket
#
# This function displays a list of the files in the bucket with each file's
# size. The function uses the --query parameter to retrieve only the key and
# size fields from the Contents collection.
#
# Parameters:
#     $1 - The name of the bucket.
#
# Returns:
#     The list of files in text format.
#     And:
#     0 - If successful.
#     1 - If it fails.
#####
function list_items_in_bucket() {
    local bucket_name=$1
    local response

    response=$(aws s3api list-objects \

```

```

    --bucket "$bucket_name" \
    --output text \
    --query 'Contents[].{Key: Key, Size: Size}')
```

```

# shellcheck disable=SC2181
if [[ ${?} -eq 0 ]]; then
    echo "$response"
else
    errecho "ERROR: AWS reports s3api list-objects operation failed.\n$response"
    return 1
fi
}

#####
# function delete_items_in_bucket
#
# This function deletes the specified list of keys from the specified bucket.
#
# Parameters:
#     $1 - The name of the bucket.
#     $2 - A list of keys in the bucket to delete.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function delete_items_in_bucket() {
    local bucket_name=$1
    local keys=$2
    local response

    # Create the JSON for the items to delete.
    local delete_items
    delete_items="{\"Objects\":["
    for key in $keys; do
        delete_items="$delete_items{\"Key\": \"$key\"},"
    done
    delete_items=${delete_items%?} # Remove the final comma.
    delete_items="$delete_items]"

    response=$(aws s3api delete-objects \
        --bucket "$bucket_name" \
        --delete "$delete_items")

```

```

# shellcheck disable=SC2181
if [[ $? -ne 0 ]]; then
    errecho "ERROR: AWS reports s3api delete-object operation failed.\n$response"
    return 1
fi
}

#####
# function delete_bucket
#
# This function deletes the specified bucket.
#
# Parameters:
#     $1 - The name of the bucket.

# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function delete_bucket() {
    local bucket_name=$1
    local response

    response=$(aws s3api delete-bucket \
        --bucket "$bucket_name")

    # shellcheck disable=SC2181
    if [[ $? -ne 0 ]]; then
        errecho "ERROR: AWS reports s3api delete-bucket failed.\n$response"
        return 1
    fi
}

```

- Para API obtener más información, consulte los siguientes temas en AWS CLI Command Reference.
  - [CopyObject](#)
  - [CreateBucket](#)
  - [DeleteBucket](#)
  - [DeleteObjects](#)
  - [GetObject](#)

- [ListObjectsV2](#)
- [PutObject](#)

## Acciones

### CopyObject

En el siguiente ejemplo de código, se muestra cómo usar CopyObject.

#### AWS CLI con script Bash

#### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function copy_item_in_bucket
#
# This function creates a copy of the specified file in the same bucket.
#
# Parameters:
#     $1 - The name of the bucket to copy the file from and to.
#     $2 - The key of the source file to copy.
#     $3 - The key of the destination file.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function copy_item_in_bucket() {
```

```

local bucket_name=$1
local source_key=$2
local destination_key=$3
local response

response=$(aws s3api copy-object \
  --bucket "$bucket_name" \
  --copy-source "$bucket_name/$source_key" \
  --key "$destination_key")

# shellcheck disable=SC2181
if [[ $? -ne 0 ]]; then
  errecho "ERROR: AWS reports s3api copy-object operation failed.\n$response"
  return 1
fi
}

```

- Para API obtener más información, consulte [CopyObject](#) la Referencia de AWS CLI comandos.

## CreateBucket

En el siguiente ejemplo de código, se muestra cómo usar CreateBucket.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

```

```
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function create-bucket
#
# This function creates the specified bucket in the specified AWS Region, unless
# it already exists.
#
# Parameters:
#     -b bucket_name -- The name of the bucket to create.
#     -r region_code -- The code for an AWS Region in which to
#                       create the bucket.
#
# Returns:
#     The URL of the bucket that was created.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function create_bucket() {
    local bucket_name region_code response
    local option OPTARG # Required to use getopt command in a function.

    # bashsupport disable=BP5008
    function usage() {
        echo "function create_bucket"
        echo "Creates an Amazon S3 bucket. You must supply a bucket name:"
        echo "  -b bucket_name    The name of the bucket. It must be globally unique."
        echo "  [-r region_code]  The code for an AWS Region in which the bucket is
created."
        echo ""
    }
}

# Retrieve the calling parameters.
while getopt "b:r:h" option; do
```



```
case "${option}" in
  b) bucket_name="${OPTARG}" ;;
  r) region_code="${OPTARG}" ;;
  h)
    usage
    return 0
    ;;
  \?)
    echo "Invalid parameter"
    usage
    return 1
    ;;
esac
done

if [[ -z "$bucket_name" ]]; then
  errecho "ERROR: You must provide a bucket name with the -b parameter."
  usage
  return 1
fi

local bucket_config_arg
# A location constraint for "us-east-1" returns an error.
if [[ -n "$region_code" ]] && [[ "$region_code" != "us-east-1" ]]; then
  bucket_config_arg="--create-bucket-configuration LocationConstraint=
$region_code"
fi

iecho "Parameters:\n"
iecho "    Bucket name:  $bucket_name"
iecho "    Region code:  $region_code"
iecho ""

# If the bucket already exists, we don't want to try to create it.
if (bucket_exists "$bucket_name"); then
  errecho "ERROR: A bucket with that name already exists. Try again."
  return 1
fi

# shellcheck disable=SC2086
response=$(aws s3api create-bucket \
  --bucket "$bucket_name" \
  $bucket_config_arg)
```

```
# shellcheck disable=SC2181
if [[ ${?} -ne 0 ]]; then
    errecho "ERROR: AWS reports create-bucket operation failed.\n$response"
    return 1
fi
}
```

- Para API obtener más información, consulte [CreateBucket](#) la Referencia de AWS CLI comandos.

## DeleteBucket

En el siguiente ejemplo de código, se muestra cómo usar DeleteBucket.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function delete_bucket
#
# This function deletes the specified bucket.
#
# Parameters:
#     $1 - The name of the bucket.
#
# Returns:
```

```

#      0 - If successful.
#      1 - If it fails.
#####
function delete_bucket() {
    local bucket_name=$1
    local response

    response=$(aws s3api delete-bucket \
        --bucket "$bucket_name")

    # shellcheck disable=SC2181
    if [[ $? -ne 0 ]]; then
        errecho "ERROR: AWS reports s3api delete-bucket failed.\n$response"
        return 1
    fi
}

```

- Para API obtener más información, consulte [DeleteBucket](#) la Referencia de AWS CLI comandos.

## DeleteObject

En el siguiente ejemplo de código, se muestra cómo usar DeleteObject.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

```

```
#####
# function delete_item_in_bucket
#
# This function deletes the specified file from the specified bucket.
#
# Parameters:
#     $1 - The name of the bucket.
#     $2 - The key (file name) in the bucket to delete.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function delete_item_in_bucket() {
    local bucket_name=$1
    local key=$2
    local response

    response=$(aws s3api delete-object \
        --bucket "$bucket_name" \
        --key "$key")

    # shellcheck disable=SC2181
    if [[ $? -ne 0 ]]; then
        errecho "ERROR: AWS reports s3api delete-object operation failed.\n$response"
        return 1
    fi
}

```

- Para API obtener más información, consulte [DeleteObject](#) la Referencia de AWS CLI comandos.

## DeleteObjects

En el siguiente ejemplo de código, se muestra cómo usar DeleteObjects.

## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function delete_items_in_bucket
#
# This function deletes the specified list of keys from the specified bucket.
#
# Parameters:
#     $1 - The name of the bucket.
#     $2 - A list of keys in the bucket to delete.

# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function delete_items_in_bucket() {
    local bucket_name=$1
    local keys=$2
    local response

    # Create the JSON for the items to delete.
    local delete_items
    delete_items="{\"Objects\":[\"
    for key in $keys; do
        delete_items=\"$delete_items{\\\"Key\\\": \\\"$key\\\"},\"
    done
    delete_items=${delete_items%?} # Remove the final comma.
```

```

delete_items="$delete_items]}"

response=$(aws s3api delete-objects \
  --bucket "$bucket_name" \
  --delete "$delete_items")

# shellcheck disable=SC2181
if [[ $? -ne 0 ]]; then
  errecho "ERROR: AWS reports s3api delete-object operation failed.\n$response"
  return 1
fi
}

```

- Para API obtener más información, consulte [DeleteObjects](#) la Referencia de AWS CLI comandos.

## GetObject

En el siguiente ejemplo de código, se muestra cómo usar GetObject.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#####
# function download_object_from_bucket
#

```

```

# This function downloads an object in a bucket to a file.
#
# Parameters:
#     $1 - The name of the bucket to download the object from.
#     $2 - The path and file name to store the downloaded bucket.
#     $3 - The key (name) of the object in the bucket.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function download_object_from_bucket() {
    local bucket_name=$1
    local destination_file_name=$2
    local object_name=$3
    local response

    response=$(aws s3api get-object \
        --bucket "$bucket_name" \
        --key "$object_name" \
        "$destination_file_name")

    # shellcheck disable=SC2181
    if [[ ${?} -ne 0 ]]; then
        errecho "ERROR: AWS reports put-object operation failed.\n$response"
        return 1
    fi
}

```

- Para API obtener más información, consulte [GetObject](#) la Referencia de AWS CLI comandos.

## HeadBucket

En el siguiente ejemplo de código, se muestra cómo usar HeadBucket.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function bucket_exists
#
# This function checks to see if the specified bucket already exists.
#
# Parameters:
#     $1 - The name of the bucket to check.
#
# Returns:
#     0 - If the bucket already exists.
#     1 - If the bucket doesn't exist.
#####
function bucket_exists() {
    local bucket_name
    bucket_name=$1

    # Check whether the bucket already exists.
    # We suppress all output - we're interested only in the return code.

    if aws s3api head-bucket \
        --bucket "$bucket_name" \
        >/dev/null 2>&1; then
        return 0 # 0 in Bash script means true.
    else
        return 1 # 1 in Bash script means false.
    fi
}
```

- Para API obtener más información, consulte [HeadBucket](#) la Referencia de AWS CLI comandos.

## ListObjectsV2

En el siguiente ejemplo de código, se muestra cómo usar ListObjectsV2.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).



```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function list_items_in_bucket
#
# This function displays a list of the files in the bucket with each file's
# size. The function uses the --query parameter to retrieve only the key and
# size fields from the Contents collection.
#
# Parameters:
#     $1 - The name of the bucket.
#
# Returns:
#     The list of files in text format.
#
# And:
#     0 - If successful.
#     1 - If it fails.
#####
function list_items_in_bucket() {
    local bucket_name=$1
    local response

    response=$(aws s3api list-objects \
        --bucket "$bucket_name" \
        --output text \
        --query 'Contents[].{Key: Key, Size: Size}')

    # shellcheck disable=SC2181
    if [[ ${?} -eq 0 ]]; then
        echo "$response"
    else
        errecho "ERROR: AWS reports s3api list-objects operation failed.\n$response"
        return 1
    fi
}

```

- Para API obtener más información, consulte la [ListObjectsversión 2](#) en la Referencia de AWS CLI comandos.

## PutObject

En el siguiente ejemplo de código, se muestra cómo usar PutObject.

AWS CLI con el script Bash

### Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function copy_file_to_bucket
#
# This function creates a file in the specified bucket.
#
# Parameters:
#     $1 - The name of the bucket to copy the file to.
#     $2 - The path and file name of the local file to copy to the bucket.
#     $3 - The key (name) to call the copy of the file in the bucket.
#
# Returns:
#     0 - If successful.
#     1 - If it fails.
#####
function copy_file_to_bucket() {
    local response bucket_name source_file destination_file_name
    bucket_name=$1
    source_file=$2
```

```
destination_file_name=$3

response=$(aws s3api put-object \
  --bucket "$bucket_name" \
  --body "$source_file" \
  --key "$destination_file_name")

# shellcheck disable=SC2181
if [[ ${?} -ne 0 ]]; then
  errecho "ERROR: AWS reports put-object operation failed.\n$response"
  return 1
fi
}
```

- Para API obtener más información, consulte [PutObject](#) la Referencia de AWS CLI comandos.

## AWS STS ejemplos de uso AWS CLI con el script Bash

Los siguientes ejemplos de código muestran cómo realizar acciones e implementar escenarios comunes mediante el uso del script AWS Command Line Interface with Bash with. AWS STS

Las acciones son extractos de código de programas más grandes y deben ejecutarse en contexto. Mientras las acciones muestran cómo llamar a las funciones de servicio individuales, es posible ver las acciones en contexto en los escenarios relacionados.

Cada ejemplo incluye un enlace al código fuente completo, donde puede encontrar instrucciones sobre cómo configurar y ejecutar el código en su contexto.

Temas

- [Acciones](#)

### Acciones

#### **AssumeRole**

En el siguiente ejemplo de código, se muestra cómo usar AssumeRole.

## AWS CLI con el script Bash

 Note

Hay más información. GitHub Busque el ejemplo completo y aprenda a configurar y ejecutar en el [Repositorio de ejemplos de código de AWS](#).

```
#####
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
#####
function iecho() {
    if [[ $VERBOSE == true ]]; then
        echo "$@"
    fi
}

#####
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####
function errecho() {
    printf "%s\n" "$*" 1>&2
}

#####
# function sts_assume_role
#
# This function assumes a role in the AWS account and returns the temporary
# credentials.
#
# Parameters:
#     -n role_session_name -- The name of the session.
#     -r role_arn -- The ARN of the role to assume.
#
# Returns:
#     [access_key_id, secret_access_key, session_token]
#
# And:
```

```

#      0 - If successful.
#      1 - If an error occurred.
#####
function sts_assume_role() {
    local role_session_name role_arn response
    local option OPTARG # Required to use getopt command in a function.

# bashsupport disable=BP5008
function usage() {
    echo "function sts_assume_role"
    echo "Assumes a role in the AWS account and returns the temporary credentials:"
    echo "  -n role_session_name -- The name of the session."
    echo "  -r role_arn -- The ARN of the role to assume."
    echo ""
}

while getopt n:r:h option; do
    case "${option}" in
        n) role_session_name=${OPTARG} ;;
        r) role_arn=${OPTARG} ;;
        h)
            usage
            return 0
            ;;
        \?)
            echo "Invalid parameter"
            usage
            return 1
            ;;
    esac
done

response=$(aws sts assume-role \
    --role-session-name "$role_session_name" \
    --role-arn "$role_arn" \
    --output text \
    --query "Credentials.[AccessKeyId, SecretAccessKey, SessionToken]")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-role operation failed.\n$response"
    return 1

```

```
fi

echo "$response"

return 0
}
```

- Para API obtener más información, consulte [AssumeRole](#) la Referencia de AWS CLI comandos.

# Seguridad en AWS Command Line Interface

La seguridad en la nube de AWS es la máxima prioridad. Como cliente de AWS, se beneficia de una arquitectura de red y un centro de datos diseñados para satisfacer los requisitos de seguridad de las organizaciones más exigentes.

La seguridad es una responsabilidad compartida entre AWS y usted. El [modelo de responsabilidad compartida](#) describe esto como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube: AWS es responsable de proteger la infraestructura que ejecuta los servicios de AWS en la nube de AWS. AWS también proporciona servicios que puede utilizar de forma segura. Los auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad como parte de los [Programas de conformidad de AWS](#) . Para obtener información sobre los programas de conformidad que se aplican a AWS Command Line Interface, consulte [Servicios de AWS en el ámbito del programa de conformidad](#).
- Seguridad en la nube: su responsabilidad viene determinada por el servicio de AWS que utilice. También es responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y la normativa aplicables.

Esta documentación le ayuda a comprender cómo puede aplicar el modelo de responsabilidad compartida cuando se utiliza AWS Command Line Interface (AWS CLI). En los siguientes temas, aprenderá a configurar AWS CLI conforme a sus objetivos de seguridad y conformidad. También aprenderá a usar la AWS CLI para que le ayude a monitorizar y proteger los recursos de AWS.

## Temas

- [Protección de datos en el AWS CLI](#)
- [Identity and Access Management](#)
- [Validación de conformidad para este AWS producto o servicio](#)
- [Resiliencia para este AWS producto o servicio](#)
- [Seguridad de infraestructura para este AWS producto o servicio](#)
- [Imponga una versión mínima de TLS](#)

# Protección de datos en el AWS CLI

El modelo de [responsabilidad AWS compartida modelo](#) se aplica a la protección de datos en AWS Command Line Interface. Como se describe en este modelo, AWS es responsable de proteger la infraestructura global que ejecuta todos los Nube de AWS. Usted es responsable de mantener el control sobre el contenido alojado en esta infraestructura. Usted también es responsable de las tareas de administración y configuración de seguridad para los Servicios de AWS que utiliza. Para obtener más información sobre la privacidad de los datos, consulte la sección [Privacidad de datos FAQ](#). Para obtener información sobre la protección de datos en Europa, consulte el [modelo de responsabilidad AWS compartida y](#) la entrada del GDPR blog sobre AWS seguridad.

Con fines de protección de datos, le recomendamos que proteja Cuenta de AWS las credenciales y configure los usuarios individuales con AWS IAM Identity Center o AWS Identity and Access Management (IAM). De esta manera, solo se otorgan a cada usuario los permisos necesarios para cumplir sus obligaciones laborales. También recomendamos proteger sus datos de la siguiente manera:

- Utilice la autenticación multifactorial (MFA) con cada cuenta.
- UseSSL/TLSpara comunicarse con AWS los recursos. Necesitamos TLS 1.2 y recomendamos TLS 1.3.
- Configure API y registre la actividad del usuario con AWS CloudTrail.
- Utilice soluciones de AWS cifrado, junto con todos los controles de seguridad predeterminados Servicios de AWS.
- Utilice servicios de seguridad administrados avanzados, como Amazon Macie, que lo ayuden a detectar y proteger los datos confidenciales almacenados en Amazon S3.
- Si necesita entre FIPS 140 y 3 módulos criptográficos validados para acceder a AWS través de una interfaz de línea de comandos o unaAPI, utilice un FIPS terminal. Para obtener más información sobre los FIPS puntos finales disponibles, consulte la [Norma federal de procesamiento de información \(\) FIPS 140-3](#).

Se recomienda encarecidamente no introducir nunca información confidencial o sensible, como, por ejemplo, direcciones de correo electrónico de clientes, en etiquetas o campos de formato libre, tales como el campo Nombre. Esto incluye cuando trabaja con AWS CLI o Servicios de AWS utiliza la consola, API AWS CLI, o. AWS SDKs Cualquier dato que ingrese en etiquetas o campos de formato libre utilizados para nombres se puede emplear para los registros de facturación o diagnóstico. Si



proporciona una URL a un servidor externo, le recomendamos encarecidamente que no incluya la información sobre las credenciales URL para validar la solicitud a ese servidor.

## Cifrado de datos

Una característica clave de cualquier servicio seguro es que la información se cifre cuando no se está utilizando activamente.

### Cifrado en reposo

Por sí mismo, no almacena ningún dato del cliente aparte de las credenciales que necesita para interactuar con los AWS servicios en nombre del usuario. AWS CLI

Si las utiliza AWS CLI para invocar un AWS servicio que transmite los datos de los clientes a su ordenador local para su almacenamiento, consulte el capítulo sobre seguridad y conformidad de la Guía del usuario de ese servicio para obtener información sobre cómo se almacenan, protegen y cifran esos datos.

### Cifrado en tránsito

De forma predeterminada, todos los datos transmitidos desde el ordenador cliente en el que se ejecutan los puntos finales AWS CLI y el AWS servicio se cifran enviándolos a través de una conexión HTTPS/TLS.

No necesita hacer nada para habilitar el uso de HTTPS/TLS. Siempre está habilitado, a menos que se deshabilite explícitamente para un determinado comando con la opción de línea de comandos `--no-verify-ssl`.

## Identity and Access Management

AWS Identity and Access Management (IAM) es un Servicio de AWS que ayuda al administrador a controlar de forma segura el acceso a AWS los recursos. IAM los administradores controlan quién puede autenticarse (iniciar sesión) y quién puede autorizarse (tener permisos) para usar AWS los recursos. IAM es un Servicio de AWS que puede utilizar sin coste adicional.

### Temas

- [Público](#)
- [Autenticación con identidades](#)
- [Administración de acceso mediante políticas](#)

- [¿Cómo Servicios de AWS trabajar con IAM](#)
- [Solución de problemas AWS de identidad y acceso](#)

## Público

La forma de usar AWS Identity and Access Management (IAM) varía según el trabajo en el que se realice AWS.

**Usuario del servicio:** si Servicios de AWS solía hacer su trabajo, el administrador le proporcionará las credenciales y los permisos que necesita. A medida que vaya utilizando más AWS funciones para realizar su trabajo, es posible que necesite permisos adicionales. Entender cómo se administra el acceso puede ayudarlo a solicitar los permisos correctos al administrador. Si no puede acceder a una función de AWS, consulte [Solución de problemas AWS de identidad y acceso](#) o consulte la guía del usuario de la Servicio de AWS que está utilizando.

**Administrador de servicios:** si está a cargo de AWS los recursos de su empresa, probablemente tenga acceso total a ellos AWS. Su trabajo consiste en determinar a qué AWS funciones y recursos deben acceder los usuarios del servicio. A continuación, debe enviar solicitudes a su IAM administrador para cambiar los permisos de los usuarios del servicio. Revise la información de esta página para comprender los conceptos básicos de IAM. Para obtener más información sobre cómo puede usarlo IAM su empresa AWS, consulte la guía del usuario del Servicio de AWS que está utilizando.

**IAM administrador:** si es IAM administrador, puede que desee obtener más información sobre cómo puede redactar políticas para administrar el acceso AWS. Para ver ejemplos de políticas AWS basadas en la identidad que puede utilizar IAM, consulte la guía del usuario de la Servicio de AWS que está utilizando.

## Autenticación con identidades

La autenticación es la forma de iniciar sesión AWS con sus credenciales de identidad. Debe estar autenticado (con quien haya iniciado sesión AWS) como IAM usuario o asumiendo un IAM rol.

Usuario raíz de la cuenta de AWS

Puede iniciar sesión AWS como una identidad federada mediante las credenciales proporcionadas a través de una fuente de identidad. AWS IAM Identity Center Los usuarios (IAM Identity Center), la autenticación de inicio de sesión único de su empresa y sus credenciales de Google o Facebook son ejemplos de identidades federadas. Al iniciar sesión como una identidad federada, el administrador

configuró previamente la federación de identidades mediante roles. IAM Cuando accede AWS mediante la federación, asume indirectamente un rol.

Según el tipo de usuario que sea, puede iniciar sesión en el portal AWS Management Console o en el de AWS acceso. Para obtener más información sobre cómo iniciar sesión AWS, consulte [Cómo iniciar sesión Cuenta de AWS en su](#) Guía del AWS Sign-In usuario.

Si accede AWS mediante programación, AWS incluye un kit de desarrollo de software (SDK) y una interfaz de línea de comandos (CLI) para firmar criptográficamente sus solicitudes con sus credenciales. Si no utilizas AWS herramientas, debes firmar las solicitudes tú mismo. Para obtener más información sobre cómo usar el método recomendado para firmar las solicitudes usted mismo, consulte [Firmar AWS API las solicitudes](#) en la Guía del IAM usuario.

Independientemente del método de autenticación que use, es posible que deba proporcionar información de seguridad adicional. Por ejemplo, le AWS recomienda que utilice la autenticación multifactorial (MFA) para aumentar la seguridad de su cuenta. Para obtener más información, consulte [Autenticación multifactorial](#) en la Guía del AWS IAM Identity Center usuario y [Uso de la autenticación multifactorial \(MFA\) AWS en](#) la Guía del IAM usuario.

## Cuenta de AWS usuario root

Al crear una Cuenta de AWS, comienza con una identidad de inicio de sesión que tiene acceso completo a todos Servicios de AWS los recursos de la cuenta. Esta identidad se denomina usuario Cuenta de AWS raíz y se accede a ella iniciando sesión con la dirección de correo electrónico y la contraseña que utilizaste para crear la cuenta. Recomendamos encarecidamente que no utilice el usuario raíz para sus tareas diarias. Proteja las credenciales del usuario raíz y utilícelas solo para las tareas que solo el usuario raíz pueda realizar. Para ver la lista completa de tareas que requieren que inicie sesión como usuario root, consulte [Tareas que requieren credenciales de usuario root](#) en la Guía del IAM usuario.

## Identidad federada

Como práctica recomendada, exija a los usuarios humanos, incluidos los que requieren acceso de administrador, que utilicen la federación con un proveedor de identidades para acceder Servicios de AWS mediante credenciales temporales.

Una identidad federada es un usuario del directorio de usuarios de su empresa, un proveedor de identidades web AWS Directory Service, el directorio del Centro de Identidad o cualquier usuario al que acceda Servicios de AWS mediante las credenciales proporcionadas a través de una fuente de

identidad. Cuando las identidades federadas acceden Cuentas de AWS, asumen funciones y las funciones proporcionan credenciales temporales.

Para una administración de acceso centralizada, le recomendamos que utilice AWS IAM Identity Center. Puede crear usuarios y grupos en IAM Identity Center, o puede conectarse y sincronizarse con un conjunto de usuarios y grupos de su propia fuente de identidad para usarlos en todas sus aplicaciones Cuentas de AWS. Para obtener información sobre IAM Identity Center, consulte [¿Qué es IAM Identity Center?](#) en la Guía AWS IAM Identity Center del usuario.

## Usuarios y grupos de IAM

Un [IAMusuario](#) es una identidad propia Cuenta de AWS que tiene permisos específicos para una sola persona o aplicación. Siempre que sea posible, recomendamos utilizar credenciales temporales en lugar de crear IAM usuarios con credenciales de larga duración, como contraseñas y claves de acceso. Sin embargo, si tiene casos de uso específicos que requieren credenciales a largo plazo con IAM los usuarios, le recomendamos que rote las claves de acceso. Para obtener más información, consulte [Rotar las claves de acceso con regularidad para los casos de uso que requieran credenciales de larga duración](#) en la Guía del IAM usuario.

Un [IAMgrupo](#) es una identidad que especifica un conjunto de IAM usuarios. No puede iniciar sesión como grupo. Puede usar los grupos para especificar permisos para varios usuarios a la vez. Los grupos facilitan la administración de los permisos para grandes conjuntos de usuarios. Por ejemplo, puede asignar un nombre a un grupo IAMAdmins y concederle permisos para administrar IAM los recursos.

Los usuarios son diferentes de los roles. Un usuario se asocia exclusivamente a una persona o aplicación, pero la intención es que cualquier usuario pueda asumir un rol que necesite. Los usuarios tienen credenciales de larga duración permanentes; no obstante, los roles proporcionan credenciales temporales. Para obtener más información, consulte [Cuándo crear un IAM usuario \(en lugar de un rol\)](#) en la Guía del IAM usuario.

## IAMroles

Un [IAMrol](#) es una identidad dentro de tu Cuenta de AWS que tiene permisos específicos. Es similar a un IAM usuario, pero no está asociado a una persona específica. Puede asumir temporalmente un IAM rol en el AWS Management Console [cambiando de rol](#). Puede asumir un rol llamando a una AWS API operación AWS CLI o utilizando una operación personalizada URL. Para obtener más información sobre los métodos de uso de roles, consulte [Uso de IAM roles](#) en la Guía del IAM usuario.

IAM los roles con credenciales temporales son útiles en las siguientes situaciones:

- **Acceso de usuario federado:** para asignar permisos a una identidad federada, puede crear un rol y definir sus permisos. Cuando se autentica una identidad federada, se asocia la identidad al rol y se le conceden los permisos define el rol. Para obtener información sobre los roles para la federación, consulte [Creación de un rol para un proveedor de identidad externo](#) en la Guía del IAM usuario. Si usa IAM Identity Center, configura un conjunto de permisos. Para controlar a qué pueden acceder sus identidades después de autenticarse, IAM Identity Center correlaciona el conjunto de permisos con un rol en. IAM Para obtener información acerca de los conjuntos de permisos, consulte [Conjuntos de permisos](#) en la Guía del usuario de AWS IAM Identity Center .
- **Permisos IAM de usuario temporales:** un IAM usuario o rol puede asumir un IAM rol para asumir temporalmente diferentes permisos para una tarea específica.
- **Acceso multicuenta:** puedes usar un IAM rol para permitir que alguien (un responsable de confianza) de una cuenta diferente acceda a los recursos de tu cuenta. Los roles son la forma principal de conceder acceso entre cuentas. Sin embargo, con algunos Servicios de AWS, puedes adjuntar una política directamente a un recurso (en lugar de usar un rol como proxy). Para conocer la diferencia entre las funciones y las políticas basadas en recursos para el acceso multicuenta, consulta el tema sobre el acceso a los [recursos entre cuentas IAM en](#) la Guía del IAM usuario.
- **Acceso entre servicios:** algunos Servicios de AWS utilizan funciones en otros. Servicios de AWS Por ejemplo, cuando realizas una llamada en un servicio, es habitual que ese servicio ejecute aplicaciones en Amazon EC2 o almacene objetos en Amazon S3. Es posible que un servicio haga esto usando los permisos de la entidad principal, usando un rol de servicio o usando un rol vinculado al servicio.
- **Sesiones de acceso directo (FAS):** cuando utilizas un IAM usuario o un rol para realizar acciones en AWS ellas, se te considera director. Cuando utiliza algunos servicios, es posible que realice una acción que desencadene otra acción en un servicio diferente. FAS utiliza los permisos del principal que llama a an Servicio de AWS, junto con los que solicitan, Servicio de AWS para realizar solicitudes a los servicios descendentes. FAS las solicitudes solo se realizan cuando un servicio recibe una solicitud que requiere interacciones con otros Servicios de AWS recursos para completarse. En este caso, debe tener permisos para realizar ambas acciones. Para obtener detalles sobre la política a la hora de realizar FAS solicitudes, consulte [Reenviar sesiones de acceso](#).
- **Función de servicio:** una función de servicio es una [IAM función](#) que un servicio asume para realizar acciones en su nombre. Un IAM administrador puede crear, modificar y eliminar un rol de servicio desde dentro IAM. Para obtener más información, consulte [Crear un rol para delegar permisos Servicio de AWS en un rol en el IAM Manual del usuario](#).

- **Función vinculada a un servicio:** una función vinculada a un servicio es un tipo de función de servicio que está vinculada a un. Servicio de AWS El servicio puede asumir el rol para realizar una acción en su nombre. Los roles vinculados al servicio aparecen en usted Cuenta de AWS y son propiedad del servicio. Un IAM administrador puede ver los permisos de los roles vinculados al servicio, pero no editarlos.
- **Aplicaciones que se ejecutan en Amazon EC2:** puedes usar un IAM rol para administrar las credenciales temporales de las aplicaciones que se ejecutan en una EC2 instancia y que realizan AWS CLI o AWS API solicitan. Esto es preferible a almacenar las claves de acceso en la EC2 instancia. Para asignar un AWS rol a una EC2 instancia y ponerlo a disposición de todas sus aplicaciones, debe crear un perfil de instancia adjunto a la instancia. Un perfil de instancia contiene el rol y permite que los programas que se ejecutan en la EC2 instancia obtengan credenciales temporales. Para obtener más información, consulte [Uso de un IAM rol para conceder permisos a aplicaciones que se ejecutan en EC2 instancias de Amazon](#) en la Guía del IAM usuario.

Para saber si se deben usar IAM roles o IAM usuarios, consulte [Cuándo crear un IAM rol \(en lugar de un usuario\)](#) en la Guía del IAM usuario.

## Administración de acceso mediante políticas

El acceso se controla AWS creando políticas y adjuntándolas a AWS identidades o recursos. Una política es un objeto AWS que, cuando se asocia a una identidad o un recurso, define sus permisos. AWS evalúa estas políticas cuando un director (usuario, usuario raíz o sesión de rol) realiza una solicitud. Los permisos en las políticas determinan si la solicitud se permite o se deniega. La mayoría de las políticas se almacenan AWS como JSON documentos. Para obtener más información sobre la estructura y el contenido de los documentos de JSON políticas, consulte [Descripción general de JSON las políticas](#) en la Guía del IAM usuario.

Los administradores pueden usar AWS JSON políticas para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

De forma predeterminada, los usuarios y los roles no tienen permisos. Para conceder a los usuarios permiso para realizar acciones en los recursos que necesitan, un IAM administrador puede crear IAM políticas. A continuación, el administrador puede añadir las IAM políticas a las funciones y los usuarios pueden asumir las funciones.

IAM las políticas definen los permisos para una acción independientemente del método que se utilice para realizar la operación. Por ejemplo, suponga que dispone de una política que permite la

acción `iam:GetRole`. Un usuario con esa política puede obtener información sobre el rol de AWS Management Console AWS CLI, el o el AWS API.

## Políticas basadas en identidad

Las políticas basadas en la identidad son documentos de política de JSON permisos que se pueden adjuntar a una identidad, como un IAM usuario, un grupo de usuarios o un rol. Estas políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener información sobre cómo crear una política basada en la identidad, consulte [Creación de IAM políticas](#) en la Guía del usuario. IAM

Las políticas basadas en identidades pueden clasificarse además como políticas insertadas o políticas administradas. Las políticas insertadas se integran directamente en un único usuario, grupo o rol. Las políticas administradas son políticas independientes que puede adjuntar a varios usuarios, grupos y funciones de su empresa. Cuenta de AWS Las políticas administradas incluyen políticas AWS administradas y políticas administradas por el cliente. Para saber cómo elegir entre una política gestionada o una política integrada, consulte [Elegir entre políticas gestionadas y políticas integradas en la Guía del IAM](#) usuario.

## Políticas basadas en recursos

Las políticas basadas en recursos son documentos de JSON política que se adjuntan a un recurso. Algunos ejemplos de políticas basadas en recursos son las políticas de confianza de IAM roles y las políticas de bucket de Amazon S3. En los servicios que admiten políticas basadas en recursos, los administradores de servicios pueden utilizarlos para controlar el acceso a un recurso específico. Para el recurso al que se asocia la política, la política define qué acciones puede realizar una entidad principal especificada en ese recurso y en qué condiciones. Debe [especificar una entidad principal](#) en una política en función de recursos. Los principales pueden incluir cuentas, usuarios, roles, usuarios federados o. Servicios de AWS

Las políticas basadas en recursos son políticas insertadas que se encuentran en ese servicio. No puede usar políticas AWS administradas desde una política IAM basada en recursos.

## Listas de control de acceso ( ) ACLs

Las listas de control de acceso (ACLs) controlan qué responsables (miembros de la cuenta, usuarios o roles) tienen permisos para acceder a un recurso. ACLs son similares a las políticas basadas en recursos, aunque no utilizan el formato de documento de JSON políticas.

Amazon S3, AWS WAF y Amazon VPC son ejemplos de servicios compatibles con ACLs. Para obtener más información sobre ACLs, consulte la [descripción general de la lista de control de acceso \(ACL\)](#) en la Guía para desarrolladores de Amazon Simple Storage Service.

## Otros tipos de políticas

AWS admite tipos de políticas adicionales y menos comunes. Estos tipos de políticas pueden establecer el máximo de permisos que los tipos de políticas más frecuentes le conceden.

- **Límites de permisos:** un límite de permisos es una función avanzada en la que se establecen los permisos máximos que una política basada en la identidad puede conceder a una entidad IAM (IAM usuario o rol). Puede establecer un límite de permisos para una entidad. Los permisos resultantes son la intersección de las políticas basadas en la identidad de la entidad y los límites de permisos. Las políticas basadas en recursos que especifiquen el usuario o rol en el campo `Principal` no estarán restringidas por el límite de permisos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para obtener más información sobre los límites de los permisos, consulte los [límites de los permisos para IAM las entidades](#) en la Guía del IAM usuario.
- **Políticas de control de servicios (SCPs):** SCPs son JSON políticas que especifican los permisos máximos para una organización o unidad organizativa (OU) AWS Organizations. AWS Organizations es un servicio para agrupar y administrar de forma centralizada varios de los Cuentas de AWS que son propiedad de su empresa. Si habilitas todas las funciones de una organización, puedes aplicar políticas de control de servicios (SCPs) a una o a todas tus cuentas. SCP limita los permisos de las entidades en las cuentas de los miembros, incluidas las de cada una Usuario raíz de la cuenta de AWS. Para obtener más información sobre Organizations SCPs, consulte las [políticas de control de servicios](#) en la Guía del AWS Organizations usuario.
- **Políticas de sesión:** las políticas de sesión son políticas avanzadas que se pasan como parámetro cuando se crea una sesión temporal mediante programación para un rol o un usuario federado. Los permisos de la sesión resultantes son la intersección de las políticas basadas en identidades del rol y las políticas de la sesión. Los permisos también pueden proceder de una política en función de recursos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para obtener más información, consulte [las políticas de sesión](#) en la Guía del IAM usuario.

## Varios tipos de políticas

Cuando se aplican varios tipos de políticas a una solicitud, los permisos resultantes son más complicados de entender. Para saber cómo se AWS determina si se debe permitir una solicitud



cuando se trata de varios tipos de políticas, consulte la [lógica de evaluación de políticas](#) en la Guía del IAM usuario.

## ¿Cómo Servicios de AWS trabajar con IAM

Para obtener una visión general de cómo Servicios de AWS funciona con la mayoría de las IAM funciones, consulte [AWS los servicios con los que funcionan IAM](#) en la Guía del IAM usuario.

Para obtener información sobre cómo usar una función específica Servicio de AWS IAM, consulta la sección de seguridad de la Guía del usuario del servicio correspondiente.

## Solución de problemas AWS de identidad y acceso

Utilice la siguiente información como ayuda para diagnosticar y solucionar los problemas más comunes que pueden surgir al trabajar con AWS yIAM.

### Temas

- [No estoy autorizado a realizar ninguna acción en AWS](#)
- [No estoy autorizado a realizar tareas como: PassRole](#)
- [Quiero permitir que personas ajenas a mí accedan Cuenta de AWS a mis AWS recursos](#)

## No estoy autorizado a realizar ninguna acción en AWS

Si recibe un error que indica que no tiene autorización para realizar una acción, las políticas se deben actualizar para permitirle realizar la acción.

El siguiente ejemplo de error se produce cuando el mateojackson IAM usuario intenta usar la consola para ver detalles sobre un *my-example-widget* recurso ficticio pero no tiene los `aws:GetWidget` permisos ficticios.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
aws:GetWidget on resource: my-example-widget
```

En este caso, la política del usuario mateojackson debe actualizarse para permitir el acceso al recurso *my-example-widget* mediante la acción `aws:GetWidget`.

Si necesita ayuda, póngase en contacto con AWS el administrador. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

## No estoy autorizado a realizar tareas como: PassRole

Si recibe un error que indica que no tiene autorización para realizar la acción `iam:PassRole`, las políticas deben actualizarse a fin de permitirle pasar un rol a AWS.

Algunos Servicios de AWS permiten transferir una función existente a ese servicio en lugar de crear una nueva función de servicio o una función vinculada a un servicio. Para ello, debe tener permisos para transferir el rol al servicio.

El siguiente ejemplo de error se produce cuando un IAM usuario denominado `marymajor` intenta utilizar la consola para realizar una acción en ella. Sin embargo, la acción requiere que el servicio cuente con permisos que otorguen un rol de servicio. Mary no tiene permisos para transferir el rol al servicio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

En este caso, las políticas de Mary se deben actualizar para permitirle realizar la acción `iam:PassRole`.

Si necesita ayuda, póngase en contacto con AWS el administrador. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

## Quiero permitir que personas ajenas a mí accedan Cuenta de AWS a mis AWS recursos

Puede crear un rol que los usuarios de otras cuentas o las personas externas a la organización puedan utilizar para acceder a sus recursos. Puede especificar una persona de confianza para que asuma el rol. En el caso de los servicios que respaldan políticas basadas en recursos o listas de control de acceso (ACLs), puedes usar esas políticas para permitir que las personas accedan a tus recursos.

Para más información, consulte lo siguiente:

- Para saber si AWS es compatible con estas funciones, consulte [¿Cómo Servicios de AWS trabajar con IAM](#)
- Para obtener información sobre cómo proporcionar acceso a los recursos de su propiedad, consulte [Proporcionar acceso a un IAM usuario en otro Cuenta de AWS de su propiedad](#) en la Guía del IAM usuario. Cuentas de AWS

- Para obtener información sobre cómo proporcionar acceso a tus recursos a terceros Cuentas de AWS, consulta [Cómo permitir el acceso a recursos que Cuentas de AWS son propiedad de terceros](#) en la Guía del IAM usuario.
- Para obtener información sobre cómo proporcionar acceso mediante la federación de identidades, consulte [Proporcionar acceso a usuarios autenticados externamente \(federación de identidades\)](#) en la Guía del IAM usuario.
- Para saber la diferencia entre el uso de roles y políticas basadas en recursos para el acceso entre cuentas, consulte el acceso a [recursos entre cuentas IAM en la Guía](#) del usuario. IAM

## Validación de conformidad para este AWS producto o servicio

Para saber si un programa de cumplimiento Servicio de AWS está dentro del ámbito de aplicación de programas de cumplimiento específicos, consulte [Servicios de AWS Alcance por programa](#) de de cumplimiento y elija el programa de cumplimiento que le interese. Para obtener información general, consulte Programas de [AWS cumplimiento > Programas AWS](#) .

Puede descargar informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulte [Descarga de informes en AWS Artifact](#) .

Su responsabilidad de cumplimiento al Servicios de AWS utilizarlos viene determinada por la confidencialidad de sus datos, los objetivos de cumplimiento de su empresa y las leyes y reglamentos aplicables. AWS proporciona los siguientes recursos para ayudar con el cumplimiento:

- [Guías de inicio rápido sobre seguridad y cumplimiento](#): estas guías de implementación analizan las consideraciones arquitectónicas y proporcionan los pasos para implementar entornos básicos centrados en AWS la seguridad y el cumplimiento.
- [Diseñando una arquitectura basada en la HIPAA seguridad y el cumplimiento en Amazon Web Services](#): en este documento técnico se describe cómo pueden utilizar las empresas AWS para crear HIPAA aplicaciones aptas.

### Note

No todos son aptos. Servicios de AWS HIPAA Para obtener más información, consulta la [Referencia de servicios HIPAA aptos](#).

- [AWS Recursos](#) de de cumplimiento: esta colección de libros de trabajo y guías puede aplicarse a su industria y ubicación.

- [AWS Guías de cumplimiento para clientes](#): comprenda el modelo de responsabilidad compartida desde el punto de vista del cumplimiento. En las guías se resumen las mejores prácticas para garantizar la seguridad Servicios de AWS y se orientan a los controles de seguridad en varios marcos (incluidos el Instituto Nacional de Estándares y Tecnología (NIST), el Consejo de Normas de Seguridad del Sector de Tarjetas de Pago (PCI) y la Organización Internacional de Normalización (ISO)).
- [Evaluación de los recursos con reglas](#) en la guía para AWS Config desarrolladores: el AWS Config servicio evalúa en qué medida las configuraciones de los recursos cumplen con las prácticas internas, las directrices del sector y las normas.
- [AWS Security Hub](#)— Este Servicio de AWS proporciona una visión completa del estado de su seguridad interior AWS. Security Hub utiliza controles de seguridad para evaluar sus recursos de AWS y comprobar su cumplimiento con los estándares y las prácticas recomendadas del sector de la seguridad. Para obtener una lista de los servicios y controles compatibles, consulte la [Referencia de controles de Security Hub](#).
- [Amazon GuardDuty](#): Servicio de AWS detecta posibles amenazas para sus cargas de trabajo Cuentas de AWS, contenedores y datos mediante la supervisión de su entorno para detectar actividades sospechosas y maliciosas. GuardDuty puede ayudarlo a cumplir con varios requisitos de conformidad, por ejemplo PCIDSS, cumpliendo con los requisitos de detección de intrusiones exigidos por ciertos marcos de cumplimiento.
- [AWS Audit Manager](#)— Esto le Servicio de AWS ayuda a auditar continuamente su AWS consumo para simplificar la gestión del riesgo y el cumplimiento de las normativas y los estándares del sector.

Este AWS producto o servicio sigue el [modelo de responsabilidad compartida](#) a través de los servicios específicos de Amazon Web Services (AWS) a los que da soporte. Para obtener información sobre la seguridad de los AWS servicios, consulte la [página de documentación sobre la seguridad del AWS servicio](#) y [AWS los servicios que se encuentran dentro del ámbito de aplicación de AWS las medidas de conformidad establecidas por el programa de conformidad](#).

## Resiliencia para este AWS producto o servicio

La infraestructura AWS global se basa en Regiones de AWS zonas de disponibilidad.

Regiones de AWS proporcionan varias zonas de disponibilidad aisladas y separadas físicamente, que están conectadas mediante redes de baja latencia, alto rendimiento y alta redundancia.

Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y bases de datos que realizan una conmutación por error automática entre las zonas sin interrupciones. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de uno o varios centros de datos.

[Para obtener más información sobre AWS las regiones y las zonas de disponibilidad, consulte Infraestructura global.AWS](#)

Este AWS producto o servicio sigue el [modelo de responsabilidad compartida](#) a través de los servicios específicos de Amazon Web Services (AWS) a los que da soporte. Para obtener información sobre la seguridad de los AWS servicios, consulte la [página de documentación sobre la seguridad del AWS servicio](#) y [AWS los servicios que se encuentran dentro del ámbito de aplicación de AWS las medidas de conformidad establecidas por el programa de conformidad](#).

## Seguridad de infraestructura para este AWS producto o servicio

Este AWS producto o servicio utiliza servicios gestionados y, por lo tanto, está protegido por la seguridad de la red AWS global. Para obtener información sobre los servicios AWS de seguridad y cómo se AWS protege la infraestructura, consulte [Seguridad AWS en la nube](#). Para diseñar su AWS entorno utilizando las mejores prácticas de seguridad de la infraestructura, consulte [Protección de infraestructuras en un marco](#) de buena AWS arquitectura basado en el pilar de la seguridad.

Utiliza las API llamadas AWS publicadas para acceder a este AWS producto o servicio a través de la red. Los clientes deben admitir lo siguiente:

- Seguridad de la capa de transporte (TLS). Necesitamos TLS 1.2 y recomendamos TLS 1.3.
- Cifre suites con perfecto secreto (PFS), como (Ephemeral Diffie-Hellman) o DHE ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La mayoría de los sistemas modernos como Java 7 y posteriores son compatibles con estos modos.

Además, las solicitudes deben firmarse con un identificador de clave de acceso y una clave de acceso secreta que esté asociada a un director. IAM También puede utilizar [AWS Security Token Service](#) (AWS STS) para generar credenciales de seguridad temporales para firmar solicitudes.

Este AWS producto o servicio sigue el [modelo de responsabilidad compartida](#) a través de los servicios específicos de Amazon Web Services (AWS) a los que da soporte. Para obtener información sobre la seguridad de los AWS servicios, consulte la [página de documentación sobre la](#)

[seguridad del AWS servicio](#) y [AWS los servicios que se encuentran dentro del ámbito de aplicación de AWS las medidas de conformidad establecidas por el programa de conformidad](#).

## Imponga una versión mínima de TLS

Cuando se utiliza AWS Command Line Interface (AWS CLI), el protocolo Transport Layer Security (TLS) desempeña un papel crucial a la hora de proteger la comunicación entre AWS CLI y Servicios de AWS. Para aumentar la seguridad a la hora de comunicarse con AWS los servicios, debe utilizar la versión TLS 1.2 o una versión posterior.

Servicio de AWS Pueden intercambiar datos de forma segura, con el TLS protocolo que proporciona cifrado, autenticación e integridad de los datos. AWS CLI Al aprovechar el TLS protocolo, AWS CLI garantiza que sus interacciones Servicios de AWS estén protegidas contra el acceso no autorizado y las filtraciones de datos, lo que mejora la seguridad general de su AWS ecosistema.

El [modelo de](#) se aplica a protección de datos en AWS Command Line Interface. Como se describe en este modelo, AWS es responsable de proteger la infraestructura global que ejecuta todos los Servicios de AWS. Usted es responsable de mantener el control sobre el contenido alojado en esta infraestructura. También es responsable de las tareas de configuración y administración de la seguridad Servicios de AWS que utilice. Para obtener más información sobre la protección de datos, consulte [the section called “Protección de los datos”](#).

Para garantizar que la AWS CLI versión 1 no utilice ninguna TLS versión anterior a la TLS 1.2, puede que tengas que volver a compilar Open SSL para aplicar este mínimo y, a continuación, volver a compilar Python para usar el Open recién creado. SSL

### Temas

- [Determinación de los protocolos admitidos actualmente](#)
- [Compila Open SSL y Python](#)

## Determinación de los protocolos admitidos actualmente

Primero, cree un certificado autofirmado para usarlo en el servidor de prueba y en Python SDK mediante OpenSSL.

```
$ openssl req -subj '/CN=localhost' -x509 -newkey rsa:4096 -nodes -keyout key.pem -out cert.pem -days 365
```

A continuación, active un servidor de prueba con OpenSSL.

```
$ openssl s_server -key key.pem -cert cert.pem -www
```

En una nueva ventana de terminal, cree un entorno virtual e instale el SDK para Python.

```
$ python3 -m venv test-env
source test-env/bin/activate
pip install botocore
```

Cree un nuevo script de Python con un nombre `check.py` que utilice SDK la HTTP biblioteca subyacente.

```
$ import urllib3
URL = 'https://localhost:4433/'

http = urllib3.PoolManager(
    ca_certs='cert.pem',
    cert_reqs='CERT_REQUIRED',
)
r = http.request('GET', URL)
print(r.data.decode('utf-8'))
```

Ejecute el nuevo script.

```
$ python check.py
```

Muestra detalles sobre la conexión realizada. Busque «Protocol:» en la salida. Si el resultado es "TLSv1.2" o posterior, el SDK valor predeterminado es TLS v1.2 o posterior. Si se trata de una versión anterior, tendrás que volver a compilar Open SSL y recompilar Python.

Sin embargo, incluso si la instalación de Python está predeterminada en la TLS v1.2 o posterior, es posible que Python renegocie a una versión anterior a la TLS v1.2 si el servidor no es compatible con la v1.2 o posterior. TLS Para comprobar que Python no renegocia automáticamente a versiones anteriores, reinicie el servidor de prueba con lo siguiente.

```
$ openssl s_server -key key.pem -cert cert.pem -no_tls1_3 -no_tls1_2 -www
```

Si utilizas una versión anterior de OpenSSL, es posible que la marca no esté disponible. `-no_tls_3`  
Si este es el caso, quita la marca porque la versión de Open SSL que estás usando no es compatible con la TLS versión 1.3. A continuación, vuelva a ejecutar el script de Python.

```
$ python check.py
```

Si tu instalación de Python correctamente no renegocia las versiones anteriores a la TLS 1.2, deberías recibir un SSL error.

```
$ urllib3.exceptions.MaxRetryError: HTTPSConnectionPool(host='localhost', port=4433): Max retries exceeded with url: / (Caused by SSLError(SSLError(1, '[SSL: UNSUPPORTED_PROTOCOL] unsupported protocol (_ssl.c:1108)')))
```

Si puede establecer una conexión, debe volver a compilar Open SSL y Python para deshabilitar la negociación de protocolos anteriores a la TLS versión 1.2.

## Compila Open SSL y Python

Para asegurarte de que SDK o AWS CLI no negocia nada anterior a la versión TLS 1.2, debes volver a compilar Open SSL y Python. Para ello, copie el siguiente contenido con el fin de crear un script y ejecútelo.

```
#!/usr/bin/env bash
set -e

OPENSSL_VERSION="1.1.1d"
OPENSSL_PREFIX="/opt/openssl-with-min-tls1_2"
PYTHON_VERSION="3.8.1"
PYTHON_PREFIX="/opt/python-with-min-tls1_2"

curl -O "https://www.openssl.org/source/openssl-$OPENSSL_VERSION.tar.gz"
tar -xzf "openssl-$OPENSSL_VERSION.tar.gz"
cd openssl-$OPENSSL_VERSION
./config --prefix=$OPENSSL_PREFIX no-ssl3 no-tls1 no-tls1_1 no-shared
make > /dev/null
sudo make install_sw > /dev/null

cd /tmp
curl -O "https://www.python.org/ftp/python/$PYTHON_VERSION/Python-$PYTHON_VERSION.tgz"
```



```
tar -xzf "Python-$PYTHON_VERSION.tgz"  
cd Python-$PYTHON_VERSION  
./configure --prefix=$PYTHON_PREFIX --with-openssl=$OPENSSL_PREFIX --disable-shared > /  
dev/null  
make > /dev/null  
sudo make install > /dev/null
```

Esto compila una versión de Python que tiene un Open vinculado estáticamente SSL que no negocia automáticamente nada anterior TLS a la 1.2. Esto también instala Open SSL en el `/opt/openssl-with-min-tls1_2` directorio e instala Python en el `/opt/python-with-min-tls1_2` directorio. Después de ejecutar este script, confirme la instalación de la nueva versión de Python.

```
$ /opt/python-with-min-tls1_2/bin/python3 --version
```

Se debería imprimir lo siguiente.

```
$ Python 3.8.1
```

Para confirmar que esta nueva versión de Python no negocia una versión anterior a la TLS 1.2, vuelva a ejecutar los pasos desde [Determinación de los protocolos admitidos actualmente](#) la versión de Python recién instalada (es decir, `/opt/python-with-min-tls1_2/bin/python3`).

# Solucionar errores AWS CLI

En esta sección se describen los errores comunes y los pasos de solución de problemas que se deben seguir para resolver el problema. Le sugerimos seguir primero la [solución de problemas generales](#).

## Contenido

- [Solución de problemas generales para probar primero](#)
  - [Comprueba el formato de tus AWS CLI comandos](#)
  - [Compruebe si Región de AWS su AWS CLI comando está utilizando](#)
  - [Confirme que está ejecutando una versión reciente de la AWS CLI](#)
  - [Use la opción --debug](#)
  - [Habilite y revise los registros del historial de AWS CLI comandos](#)
  - [Confirme que el suyo está configurado AWS CLI](#)
- [Errores de comando no encontrado](#)
- [El comando "aws --version" devuelve una versión diferente a la instalada](#)
- [El comando "aws --version" devuelve una versión tras desinstalar la AWS CLI](#)
- [AWS CLI Procesó un comando con un nombre de parámetro incompleto](#)
- [Errores de acceso denegado](#)
- [Errores de credenciales y claves no válidas](#)
- [Errores de firma que no coincide](#)
- [Errores de no se ha encontrado la consola de Windows](#)
- [SSLerrores de certificado](#)
- [JSONErrores no válidos](#)
- [Recursos adicionales de](#)

## Solución de problemas generales para probar primero

Si recibes un error o encuentras algún problema con el AWS CLI, te sugerimos los siguientes consejos generales para ayudarte a solucionarlo.

[\(Volver arriba\)](#)

## Comprueba el formato de tus AWS CLI comandos

Si aparece un error en el que se indica que no existe un comando o que no se reconoce un parámetro (`Parameter validation failed`) que según la documentación está disponible, es posible que el comando tenga un formato incorrecto. Le sugerimos que compruebe lo siguiente:

- Compruebe si hay errores ortográficos y de formato en el comando.
- Confirme que todas [las comillas y caracteres de escape apropiados para su terminal](#) son correctos en el comando.
- Generar un [esqueleto de AWS CLI](#) para confirmar la estructura del comando.
- Para elloJSON, consulte la [solución adicional de problemas de JSON valores](#). Si tienes problemas con el JSON formato de procesamiento de tu terminal, te sugerimos que saltes las reglas de cotización de la terminal y utilices [Blobs para pasar JSON los datos directamente](#) al. AWS CLI

Para obtener más información sobre cómo debe estructurarse un comando específico, consulta la guía de referencia de la de la [guía de AWS CLI referencia](#).

[\(Volver arriba\)](#)

## Compruebe si Región de AWS su AWS CLI comando está utilizando

### Note

Debe especificar una Región de AWS cuando utilice la AWS CLI, ya sea de forma explícita o estableciendo una región predeterminada. Para obtener una lista de todos los elementos Regiones de AWS que puede especificar, consulte [AWS Regiones y puntos finales](#) en el Referencia general de Amazon Web Services. Los Región de AWS designadores que utilizan AWS CLI son los mismos nombres que aparecen en AWS Management Console URLs los puntos finales del servicio.

Se pueden producir errores o resultados inesperados si un no Servicio de AWS está disponible para el especificado Región de AWS o si los recursos están ubicados en un lugar diferente. Región de AWS En orden de prioridad, Región de AWS se establece de las siguientes maneras:

- La opción de línea de comandos `--region`.
- La variable de entorno [AWS\\_DEFAULT\\_REGION](#).

- La configuración del [region](#) perfil.

Confirma que estás utilizando la correcta Región de AWS para tus recursos.

[\(Volver arriba\)](#)

## Confirme que está ejecutando una versión reciente de la AWS CLI

Si recibe un error que indica que un comando no existe o que no reconoce un parámetro que, según la [guía de AWS CLI referencia](#) de la , está disponible, compruebe primero que el comando tiene el formato correcto. Si el formato es correcto, le recomendamos que se actualice a la versión más reciente de la AWS CLI. Las versiones actualizadas de la AWS CLI se publican casi todos los días laborables. Se introducen nuevos AWS servicios, características y parámetros en esas nuevas versiones del AWS CLI. La única manera de obtener acceso a los nuevos servicios, características o parámetros es actualizándose a una versión que se lanzara después de que ese elemento se incorporara por primera vez.

La forma de actualizar la versión de AWS CLI depende de cómo la haya instalado originalmente, tal y como se describe en [Instalar la AWS CLI](#).

Si ha utilizado uno de los instaladores del paquete, debe eliminar la instalación existente antes de descargar e instalar la versión más reciente para su sistema operativo.

[\(Volver arriba\)](#)

## Use la opción **--debug**

Si AWS CLI detecta un error que no entiende inmediatamente o produce resultados inesperados, puede obtener más detalles sobre el error ejecutando de nuevo el comando con la `--debug` opción. Con esta opción, la AWS CLI emite detalles sobre cada paso que realiza para procesar su comando. Los detalles de la salida pueden ayudarlo a determinar cuándo se produce el error y proporciona pistas sobre dónde comenzó.

Puede enviar el resultado a un archivo de texto para revisarlo posteriormente o enviarlo a AWS Support cuando se le solicite.

Cuando se incluye la opción `--debug`, algunos detalles son:

- Búsqueda de credenciales
- Análisis de los parámetros proporcionados

- Construir la solicitud enviada a AWS los servidores
- El contenido de la solicitud enviada a AWS
- El contenido de la respuesta sin procesar
- La salida con formato

A continuación, se muestra un ejemplo de la ejecución de un comando con y sin la opción `--debug`:

```
$ aws iam list-groups --profile MyTestProfile
{
  "Groups": [
    {
      "Path": "/",
      "GroupName": "MyTestGroup",
      "GroupId": "AGPA0123456789EXAMPLE",
      "Arn": "arn:aws:iam::123456789012:group/MyTestGroup",
      "CreateDate": "2019-08-12T19:34:04Z"
    }
  ]
}
```

```
$ aws iam list-groups --profile MyTestProfile --debug
2019-08-12 12:36:18,305 - MainThread - awscli.clidriver - DEBUG - CLI version: aws-
cli/1.16.215 Python/3.7.3 Linux/4.14.133-113.105.amzn2.x86_64 botocore/1.12.205
2019-08-12 12:36:18,305 - MainThread - awscli.clidriver - DEBUG - Arguments entered to
CLI: ['iam', 'list-groups', '--debug']
2019-08-12 12:36:18,305 - MainThread - botocore.hooks - DEBUG - Event session-
initialized: calling handler <function add_scalar_parsers at 0x7fdf173161e0>
2019-08-12 12:36:18,305 - MainThread - botocore.hooks - DEBUG - Event session-
initialized: calling handler <function register_uri_param_handler at 0x7fdf17dec400>
2019-08-12 12:36:18,305 - MainThread - botocore.hooks - DEBUG - Event session-
initialized: calling handler <function inject_assume_role_provider_cache at
0x7fdf17da9378>
2019-08-12 12:36:18,307 - MainThread - botocore.credentials - DEBUG - Skipping
environment variable credential check because profile name was explicitly set.
2019-08-12 12:36:18,307 - MainThread - botocore.hooks - DEBUG - Event session-
initialized: calling handler <function attach_history_handler at 0x7fdf173ed9d8>
2019-08-12 12:36:18,308 - MainThread - botocore.loaders - DEBUG - Loading JSON
file: /home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/iam/2010-05-08/
service-2.json
2019-08-12 12:36:18,317 - MainThread - botocore.hooks - DEBUG - Event building-command-
table.iam: calling handler <function add_waiters at 0x7fdf1731a840>
```

```
2019-08-12 12:36:18,320 - MainThread - botocore.loaders - DEBUG - Loading JSON
file: /home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/iam/2010-05-08/
waiters-2.json
2019-08-12 12:36:18,321 - MainThread - awscli.clidriver - DEBUG - OrderedDict([('path-
prefix', <awscli.arguments.CLIArument object at 0x7fdf171ac780>), ('marker',
<awscli.arguments.CLIArument object at 0x7fdf171b09e8>), ('max-items',
<awscli.arguments.CLIArument object at 0x7fdf171b09b0>)])
2019-08-12 12:36:18,322 - MainThread - botocore.hooks - DEBUG - Event building-
argument-table.iam.list-groups: calling handler <function add_streaming_output_arg at
0x7fdf17316510>
2019-08-12 12:36:18,322 - MainThread - botocore.hooks - DEBUG - Event building-
argument-table.iam.list-groups: calling handler <function add_cli_input_json at
0x7fdf17da9d90>
2019-08-12 12:36:18,322 - MainThread - botocore.hooks - DEBUG - Event building-
argument-table.iam.list-groups: calling handler <function unify_paging_params at
0x7fdf17328048>
2019-08-12 12:36:18,326 - MainThread - botocore.loaders - DEBUG - Loading JSON
file: /home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/iam/2010-05-08/
paginators-1.json
2019-08-12 12:36:18,326 - MainThread - awscli.customizations.paginate - DEBUG -
Modifying paging parameters for operation: ListGroups
2019-08-12 12:36:18,326 - MainThread - botocore.hooks - DEBUG - Event building-
argument-table.iam.list-groups: calling handler <function add_generate_skeleton at
0x7fdf1737eae8>
2019-08-12 12:36:18,326 - MainThread - botocore.hooks - DEBUG - Event
before-building-argument-table-parser.iam.list-groups: calling handler
<bound method OverrideRequiredArgsArgument.override_required_args of
<awscli.customizations.cliinputjson.CliInputJSONArgument object at 0x7fdf171b0a58>>
2019-08-12 12:36:18,327 - MainThread - botocore.hooks - DEBUG - Event
before-building-argument-table-parser.iam.list-groups: calling handler
<bound method GenerateCliSkeletonArgument.override_required_args of
<awscli.customizations.generatecliskeleton.GenerateCliSkeletonArgument object at
0x7fdf171c5978>>
2019-08-12 12:36:18,327 - MainThread - botocore.hooks - DEBUG - Event operation-
args-parsed.iam.list-groups: calling handler functools.partial(<function
check_should_enable_pagination at 0x7fdf17328158>, ['marker', 'max-items'], {'max-
items': <awscli.arguments.CLIArument object at 0x7fdf171b09b0>}, OrderedDict([('path-
prefix', <awscli.arguments.CLIArument object at 0x7fdf171ac780>), ('marker',
<awscli.arguments.CLIArument object at 0x7fdf171b09e8>), ('max-items',
<awscli.customizations.paginate.PageArgument object at 0x7fdf171c58d0>), ('cli-
input-json', <awscli.customizations.cliinputjson.CliInputJSONArgument object at
0x7fdf171b0a58>), ('starting-token', <awscli.customizations.paginate.PageArgument
object at 0x7fdf171b0a20>), ('page-size', <awscli.customizations.paginate.PageArgument
object at 0x7fdf171c5828>), ('generate-cli-skeleton',
```

```
<awscli.customizations.generatecliskeleton.GenerateCliSkeletonArgument object at
0x7fdf171c5978>]]))
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event load-cli-
arg.iam.list-groups.path-prefix: calling handler <awscli.paramfile.URIArgumentHandler
object at 0x7fdf1725c978>
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event load-cli-
arg.iam.list-groups.marker: calling handler <awscli.paramfile.URIArgumentHandler object
at 0x7fdf1725c978>
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event load-cli-
arg.iam.list-groups.max-items: calling handler <awscli.paramfile.URIArgumentHandler
object at 0x7fdf1725c978>
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG -
Event load-cli-arg.iam.list-groups.cli-input-json: calling handler
<awscli.paramfile.URIArgumentHandler object at 0x7fdf1725c978>
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG -
Event load-cli-arg.iam.list-groups.starting-token: calling handler
<awscli.paramfile.URIArgumentHandler object at 0x7fdf1725c978>
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event load-cli-
arg.iam.list-groups.page-size: calling handler <awscli.paramfile.URIArgumentHandler
object at 0x7fdf1725c978>
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event
load-cli-arg.iam.list-groups.generate-cli-skeleton: calling handler
<awscli.paramfile.URIArgumentHandler object at 0x7fdf1725c978>
2019-08-12 12:36:18,329 - MainThread - botocore.hooks - DEBUG
- Event calling-command.iam.list-groups: calling handler
<bound method CliInputJSONArgument.add_to_call_parameters of
<awscli.customizations.cliinputjson.CliInputJSONArgument object at 0x7fdf171b0a58>>
2019-08-12 12:36:18,329 - MainThread - botocore.hooks - DEBUG -
Event calling-command.iam.list-groups: calling handler <bound
method GenerateCliSkeletonArgument.generate_json_skeleton of
<awscli.customizations.generatecliskeleton.GenerateCliSkeletonArgument object at
0x7fdf171c5978>>
2019-08-12 12:36:18,329 - MainThread - botocore.credentials - DEBUG - Looking for
credentials via: assume-role
2019-08-12 12:36:18,329 - MainThread - botocore.credentials - DEBUG - Looking for
credentials via: assume-role-with-web-identity
2019-08-12 12:36:18,329 - MainThread - botocore.credentials - DEBUG - Looking for
credentials via: shared-credentials-file
2019-08-12 12:36:18,329 - MainThread - botocore.credentials - INFO - Found credentials
in shared credentials file: ~/.aws/credentials
2019-08-12 12:36:18,330 - MainThread - botocore.loaders - DEBUG - Loading JSON file: /
home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/endpoints.json
2019-08-12 12:36:18,334 - MainThread - botocore.hooks - DEBUG - Event choose-service-
name: calling handler <function handle_service_name_alias at 0x7fdf1898eb70>
```

```
2019-08-12 12:36:18,337 - MainThread - botocore.hooks - DEBUG - Event creating-client-
class.iam: calling handler <function add_generate_presigned_url at 0x7fdf18a028c8>
2019-08-12 12:36:18,337 - MainThread - botocore.regions - DEBUG - Using partition
endpoint for iam, us-west-2: aws-global
2019-08-12 12:36:18,337 - MainThread - botocore.args - DEBUG - The s3 config key is not
a dictionary type, ignoring its value of: None
2019-08-12 12:36:18,340 - MainThread - botocore.endpoint - DEBUG - Setting iam timeout
as (60, 60)
2019-08-12 12:36:18,341 - MainThread - botocore.loaders - DEBUG - Loading JSON file: /
home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/_retry.json
2019-08-12 12:36:18,341 - MainThread - botocore.client - DEBUG - Registering retry
handlers for service: iam
2019-08-12 12:36:18,342 - MainThread - botocore.hooks - DEBUG - Event before-
parameter-build.iam.ListGroups: calling handler <function generate_idempotent_uuid at
0x7fdf189b10d0>
2019-08-12 12:36:18,342 - MainThread - botocore.hooks - DEBUG - Event before-
call.iam.ListGroups: calling handler <function inject_api_version_header_if_needed at
0x7fdf189b2a60>
2019-08-12 12:36:18,343 - MainThread - botocore.endpoint - DEBUG - Making
request for OperationModel(name=ListGroups) with params: {'url_path': '/',
'query_string': '', 'method': 'POST', 'headers': {'Content-Type': 'application/x-
www-form-urlencoded; charset=utf-8', 'User-Agent': 'aws-cli/1.16.215 Python/3.7.3
Linux/4.14.133-113.105.amzn2.x86_64 botocore/1.12.205'}, 'body': {'Action':
'ListGroups', 'Version': '2010-05-08'}, 'url': 'https://iam.amazonaws.com/',
'context': {'client_region': 'aws-global', 'client_config': <botoconfig.Config
object at 0x7fdf16e9a4a8>, 'has_streaming_input': False, 'auth_type': None}}
2019-08-12 12:36:18,343 - MainThread - botocore.hooks - DEBUG - Event request-
created.iam.ListGroups: calling handler <bound method RequestSigner.handler of
<botoconfig.signers.RequestSigner object at 0x7fdf16e9a470>>
2019-08-12 12:36:18,343 - MainThread - botocore.hooks - DEBUG - Event choose-
signer.iam.ListGroups: calling handler <function set_operation_specific_signer at
0x7fdf18996f28>
2019-08-12 12:36:18,343 - MainThread - botocore.auth - DEBUG - Calculating signature
using v4 auth.
2019-08-12 12:36:18,343 - MainThread - botocore.auth - DEBUG - CanonicalRequest:
POST
/

content-type:application/x-www-form-urlencoded; charset=utf-8
host:iam.amazonaws.com
x-amz-date:20190812T193618Z

content-type;host;x-amz-date
5f776d91EXAMPLE9b8cb5eb5d6d4a787a33ae41c8cd6eEXAMPLEca69080e1e1f
```



```

2019-08-12 12:36:18,344 - MainThread - botocore.auth - DEBUG - StringToSign:
AWS4-HMAC-SHA256
20190812T193618Z
20190812/us-east-1/iam/aws4_request
ab7e367eEXAMPLE2769f178ea509978cf8bfa054874b3EXAMPLE8d043fab6cc9
2019-08-12 12:36:18,344 - MainThread - botocore.auth - DEBUG - Signature:
d85a0EXAMPLEeb40164f2f539cdc76d4f294fe822EXAMPLE18ad1ddf58a1a3ce7
2019-08-12 12:36:18,344 - MainThread - botocore.endpoint - DEBUG - Sending
http request: <AWSPreparedRequest stream_output=False, method=POST,
url=https://iam.amazonaws.com/, headers={'Content-Type': b'application/
x-www-form-urlencoded; charset=utf-8', 'User-Agent': b'aws-cli/1.16.215
Python/3.7.3 Linux/4.14.133-113.105.amzn2.x86_64 botocore/1.12.205',
'X-Amz-Date': b'20190812T193618Z', 'Authorization': b'AWS4-HMAC-SHA256
Credential=AKIA01234567890EXAMPLE-east-1/iam/aws4_request, SignedHeaders=content-
type;host;x-amz-date, Signature=d85a07692aceb401EXAMPLEa1b18ad1ddf58a1a3ce7EXAMPLE',
'Content-Length': '36'}>
2019-08-12 12:36:18,344 - MainThread - urllib3.util.retry - DEBUG - Converted retries
value: False -> Retry(total=False, connect=None, read=None, redirect=0, status=None)
2019-08-12 12:36:18,344 - MainThread - urllib3.connectionpool - DEBUG - Starting new
HTTPS connection (1): iam.amazonaws.com:443
2019-08-12 12:36:18,664 - MainThread - urllib3.connectionpool - DEBUG - https://
iam.amazonaws.com:443 "POST / HTTP/1.1" 200 570
2019-08-12 12:36:18,664 - MainThread - botocore.parsers - DEBUG - Response headers:
{'x-amzn-RequestId': '74c11606-bd38-11e9-9c82-559da0adb349', 'Content-Type': 'text/
xml', 'Content-Length': '570', 'Date': 'Mon, 12 Aug 2019 19:36:18 GMT'}
2019-08-12 12:36:18,664 - MainThread - botocore.parsers - DEBUG - Response body:
b'<ListGroupResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">\n
<ListGroupResult>\n  <IsTruncated>>false</IsTruncated>\n  <Groups>\n
  <member>\n    <Path>/</Path>\n    <GroupName>MyTestGroup</GroupName>
\n    <Arn>arn:aws:iam::123456789012:group/MyTestGroup</Arn>\n
  <GroupId>AGPA1234567890EXAMPLE</GroupId>\n    <CreateDate>2019-08-12T19:34:04Z</
CreateDate>\n  </member>\n  </Groups>\n </ListGroupResult>\n
<ResponseMetadata>\n  <RequestId>74c11606-bd38-11e9-9c82-559da0adb349</RequestId>\n
</ResponseMetadata>\n</ListGroupResponse>\n'
2019-08-12 12:36:18,665 - MainThread - botocore.hooks - DEBUG - Event needs-
retry.iam.ListGroups: calling handler <botocore.retryhandler.RetryHandler object at
0x7fdf16e9a780>
2019-08-12 12:36:18,665 - MainThread - botocore.retryhandler - DEBUG - No retry needed.
2019-08-12 12:36:18,665 - MainThread - botocore.hooks - DEBUG - Event after-
call.iam.ListGroups: calling handler <function json_decode_policies at 0x7fdf189b1d90>
{
  "Groups": [
    {
      "Path": "/",

```

```
    "GroupName": "MyTestGroup",
    "GroupId": "AGPA123456789012EXAMPLE",
    "Arn": "arn:aws:iam::123456789012:group/MyTestGroup",
    "CreateDate": "2019-08-12T19:34:04Z"
  }
]
```

[\(Volver arriba\)](#)

## Habilite y revise los registros del historial de AWS CLI comandos

Puede habilitar los registros del historial de AWS CLI comandos mediante la configuración del [cli\\_history](#) archivo. Tras activar esta configuración, AWS CLI registra el historial de aws comandos.

Puede hacer una lista de su historial con el comando `aws history list` y utilizar los `command_ids` resultantes en el comando `aws history show` para obtener más información. Para obtener más información, consulte [aws history](#) en la Guía de referencia de AWS CLI .

Cuando se incluye la opción `--debug`, algunos detalles son:

- API llamadas realizadas a botocore
- Códigos de estado
- HTTP respuestas
- Encabezados
- Códigos de devolución

Puede utilizar esta información para confirmar que los datos de los parámetros y las API llamadas se comportan de la forma esperada y, a continuación, deducir en qué fase del proceso está fallando el comando.

[\(Volver arriba\)](#)

## Confirme que el suyo está configurado AWS CLI

Se pueden producir varios errores si sus `credentials` archivos `config` AND o su IAM usuario o rol no están configurados correctamente. Para obtener más información sobre cómo resolver errores con `credentials` los archivos `config` y con IAM el usuario o las funciones, consulte [the](#)

[section called “Errores de acceso denegado”](#) y [the section called “Errores de credenciales y claves no válidas”](#).

[\(Volver arriba\)](#)

## Errores de comando no encontrado

Este error significa que el sistema operativo no encuentra el AWS CLI comando. Puede que la instalación esté incompleta o sea necesario actualizarla.

Causa posible: estás intentando utilizar una AWS CLI función más reciente que la versión instalada o tienes un formato incorrecto

Ejemplo de texto de error:

```
$ aws s3 copy
usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:

    aws help
    aws <command> help
    aws <command> <subcommand> help
aws: error: argument subcommand: Invalid choice, valid choices are:

ls                | website
cp                | mv
.....
```

Pueden producirse varios errores si su comando tiene un formato incorrecto o si está utilizando una versión anterior a la publicación de la característica. Para obtener más información sobre la resolución de errores en torno a estos dos problemas, consulte [the section called “Comprueba el formato de tus AWS CLI comandos”](#) y [the section called “Confirme que está ejecutando una versión reciente de la AWS CLI”](#).

[\(Volver arriba\)](#)

Causa posible: es necesario reiniciar el terminal después de la instalación

Ejemplo de texto de error:

```
$ aws --version
```

```
command not found: aws
```

Si el aws comando no se encuentra después de instalarlo o actualizarlo por primera vez AWS CLI, puede que tengas que reiniciar el terminal para que reconozca PATH las actualizaciones.

[\(Volver arriba\)](#)

Causa posible: AWS CLI no se instaló por completo

Ejemplo de texto de error:

```
$ aws --version
command not found: aws
```

Si el aws comando no se encuentra después de instalarlo o actualizarlo por primera vez AWS CLI, es posible que no esté completamente instalado. Intente reinstalar siguiendo los pasos correspondientes a su plataforma en [Instalar la AWS CLI](#).

[\(Volver arriba\)](#)

Causa posible: AWS CLI no tiene permisos (Linux)

Si el aws comando no se encuentra después de instalarlo o actualizarlo por primera vez AWS CLI en Linux, es posible que no tenga execute permisos para la carpeta en la que se instaló. Ejecute el siguiente comando con PATH la en AWS CLI la instalación para proporcionar [chmod](#) permisos a AWS CLI:

```
$ sudo chmod -R 755 /usr/local/aws-cli/
```

[\(Volver arriba\)](#)

Causa posible: la variable **PATH** del sistema operativo no se ha actualizado durante la instalación

Ejemplo de texto de error:

```
$ aws --version
command not found: aws
```

Es posible que tenga que añadir el ejecutable aws a la variable de entorno PATH del sistema operativo. Para añadirlo AWS CLI a la suyaPATH, utilice las siguientes instrucciones para su sistema operativo.

## Linux and macOS

1. Busque el script de perfil de su shell en su directorio de usuario. Si no está seguro de cuál es el shell que tiene, ejecute `echo $SHELL`.

```
$ ls -a ~  
.  ..  .bash_logout  .bash_profile  .bashrc  Desktop  Documents  Downloads
```

- Bash: `.bash_profile`, `.profile`, o `.bash_login`
  - Zsh – `.zshrc`
  - Tcsh: `.tcshrc`, `.cshrc`, o `.login`
2. Añada un comando de exportación al script de su perfil. El siguiente comando agrega su bin local a la variable PATH actual.

```
export PATH=/usr/local/bin:$PATH
```

3. Vuelva a cargar el perfil actualizado en la sesión actual.

```
$ source ~/.bash_profile
```

## Windows

1. En un símbolo del sistema de Windows, utilice el comando `where` con el parámetro `/R` *path* para encontrar la ubicación del archivo de `aws`. Los resultados devuelven todas las carpetas que contienen `aws`.

```
C:\> where /R c:\ aws  
c:\Program Files\Amazon\AWSCLIV2\aws.exe  
...
```

De forma predeterminada, la AWS CLI versión 2 se encuentra en:

```
c:\Program Files\Amazon\AWSCLIV2\aws.exe
```

2. Pulse la tecla de Windows y escriba **environment variables**.
3. En la lista de sugerencias, elija `Edit environment variables for your account` (Editar variables de entorno para la cuenta).

4. Elija y PATH, a continuación, elija Editar.
5. Agregue la ruta que encontró en el primer paso al campo Variable value (Valor de variable), por ejemplo, ***C:\Program Files\Amazon\AWSCLIV2\aws.exe***.
6. Elija OK (Aceptar) dos veces para aplicar la nueva configuración.
7. Cierre los símbolos del sistema en ejecución y vuelva a abrir la ventana de símbolo del sistema.

[\(Volver arriba\)](#)

## El comando "**aws --version**" devuelve una versión diferente a la instalada

Es posible que su terminal devuelva un precio AWS CLI diferente PATH al esperado.

Causa posible: es necesario reiniciar el terminal después de la instalación

Si el comando `aws` muestra la versión incorrecta, es posible que tenga que reiniciar su terminal para que reconozca las actualizaciones de PATH. Todos los terminales abiertos deben estar cerrados, no solo el terminal activo.

[\(Volver arriba\)](#)

Causa posible: es necesario reiniciar el sistema después de la instalación

Si el comando `aws` muestra la versión incorrecta y el reinicio del terminal no funcionó, es posible que tenga que reiniciar el terminal para que reconozca las actualizaciones de PATH.

[\(Volver arriba\)](#)

Causa posible: tienes varias versiones del AWS CLI

Si actualizó AWS CLI y utilizó un método de instalación diferente al de la instalación anterior, es posible que se instalen varias versiones. Por ejemplo, si en Linux o macOS ha utilizado `pip` para su instalación actual, pero ha intentado la actualización mediante el archivo de instalación de `.pkg`, esto podría causar algunos conflictos, especialmente con su PATH apuntando a la versión antigua.

Para resolver esto, [desinstale todas las versiones de la AWS CLI](#) y realice una instalación limpia.

Después de desinstalar todas las versiones, siga las instrucciones correspondientes a su sistema operativo para instalar la versión que desee de la [AWS CLI versión 1](#) o la [AWS CLI versión 2](#).

**Note**

Si esto ocurre después de instalar la AWS CLI versión 2 con una instalación preexistente de la AWS CLI versión 1, siga las instrucciones de migración que aparecen en las instrucciones de [instalación al migrar desde AWS CLI la](#) versión 1.

[\(Volver arriba\)](#)

## El comando "**aws --version**" devuelve una versión tras desinstalar la AWS CLI

Esto suele ocurrir cuando todavía hay una AWS CLI instalada en algún lugar del sistema.

Causa posible: es necesario reiniciar el terminal después de la desinstalación

Si el comando `aws --version` sigue funcionando, es posible que tenga que reiniciar su terminal para que reconozca las actualizaciones del terminal.

[\(Volver arriba\)](#)

Causa posible: tiene varias versiones de AWS CLI en su sistema o no utilizó el mismo método de desinstalación que utilizó para instalar originalmente el AWS CLI

AWS CLI Es posible que no se desinstale correctamente si la desinstaló AWS CLI con un método diferente al que utilizó para instalarla o si instaló varias versiones. Por ejemplo, si utilizó `pip` para su instalación actual, debe utilizar `pip` para desinstalarla. Para solucionar este problema, desinstálo AWS CLI con el mismo método que utilizó para instalarlo.

1. Siga las instrucciones adecuadas para el sistema operativo y el método de instalación original para desinstalar la [AWS CLI versión 1](#) y [AWS CLI versión 2](#).
2. Cierre todos los terminales que tenga abiertos.
3. Abra su terminal preferido, ingrese el siguiente comando y confirme que no se devuelve ninguna versión.

```
$ aws --version
```

```
command not found: aws
```

Si aún tiene una versión en la lista de resultados, lo más probable es que AWS CLI se haya instalado con un método diferente o que haya varias versiones. Si no sabe qué método ha instalado AWS CLI, siga las instrucciones de cada método de desinstalación para las [AWS CLI versiones 1](#) y [AWS CLI 2](#) correspondientes a su sistema operativo hasta que no reciba ningún resultado de la versión.

**Note**

Si ha utilizado un administrador de paquetes para instalar la AWS CLI (pip, apt, brew, etc.), debe utilizar el mismo administrador de paquetes para desinstalarla. Asegúrese de seguir las instrucciones proporcionadas por el administrador de paquetes sobre cómo desinstalar todas las versiones de un paquete.

[\(Volver arriba\)](#)

## AWS CLI Procesó un comando con un nombre de parámetro incompleto

Posible causa: ha utilizado una abreviatura reconocida del parámetro de la AWS CLI

Como AWS CLI está creado con Python, AWS CLI utiliza la `argparse` biblioteca de Python, incluido el `allow_abbrev` argumento. Las abreviaturas de los parámetros son reconocidas por el AWS CLI y procesadas.

El siguiente ejemplo de `create-change-set` comando cambia el nombre de la CloudFormation pila. El parámetro `--change-set-n` se reconoce como una abreviatura `--change-set-name` de y AWS CLI procesa el comando.

```
$ aws cloudformation create-change-set --stack-name my-stack --change-set-n my-change-set
```

Cuando su abreviatura pueda ser varios comandos, el parámetro no se reconocerá como abreviatura.



El siguiente ejemplo de [create-change-set](#) comando cambia el nombre de la CloudFormation pila. El parámetro `--change-set` no se reconoce como una abreviatura, ya que existen múltiples parámetros de los que podría ser una abreviatura, como `--change-set-name` y `--change-set-type`. Por lo tanto, AWS CLI no procesa el comando.

```
$ aws cloudformation create-change-set --stack-name my-stack --change-set- my-change-set
```

### Warning

No utilice abreviaturas de parámetros a propósito. No son fiables y no son compatibles con versiones anteriores. Si se agrega algún parámetro nuevo a un comando que confunda sus abreviaturas, se interrumpirán sus comandos.

Además, si el parámetro es un argumento de un solo valor, puede provocar un comportamiento inesperado con sus comandos. Si se pasan varias instancias de un argumento de un solo valor, solo se ejecutará la última instancia. En el siguiente ejemplo, el parámetro `--filters` es un argumento de un solo valor. Se especifican los parámetros `--filters` y `--filter`. El parámetro `--filter` es una abreviatura de `--filters`. Esto hace que se apliquen dos instancias de `--filters` y solo se aplica el último argumento `--filter`.

```
$ aws ec2 describe-vpc-peering-connections \  
  --filters Name=tag:TagName,Values=VpcPeeringConnection \  
  --filter Name=status-code,Values=active
```

Confirme que utiliza parámetros válidos antes de ejecutar un comando para evitar comportamientos inesperados.

[\(Volver arriba\)](#)

## Errores de acceso denegado

Causa posible: el archivo del AWS CLI programa no tiene permiso de «ejecución»

En Linux o macOS, asegúrese de que el programa `aws` tiene permisos de ejecución para el usuario que realiza la llamada. Normalmente, los permisos se establecen en 755.

Para añadir el permiso de ejecución a su usuario, ejecute el siguiente comando, sustituyéndolo `~/.local/bin/aws` con la ruta al programa en su ordenador.

```
$ chmod +x ~/.local/bin/aws
```

[\(Volver arriba\)](#)

Causa posible: tu IAM identidad no tiene permiso para realizar la operación

Ejemplo de texto de error:

```
$ aws s3 ls
An error occurred (AccessDenied) when calling the ListBuckets operation: Access
denied.
```

Cuando ejecutas un AWS CLI comando, AWS las operaciones se realizan en tu nombre y utilizan credenciales que te asocian a una IAM cuenta o un rol. Las políticas adjuntas deben concederle permiso para ejecutar las API acciones que correspondan a los comandos que ejecute con el AWS CLI.

La mayoría de los comandos llaman a una única acción que tiene un nombre que coincide con el nombre del comando. Sin embargo, los comandos personalizados, como `aws s3 sync` llamar a varios APIs. Puede ver APIs a qué comandos llama mediante la `--debug` opción.

Si está seguro de que el usuario o el rol tienen los permisos adecuados asignados por la política, asegúrese de que el AWS CLI comando utilice las credenciales esperadas. Consulte la [siguiente sección sobre las credenciales](#) para comprobar que las credenciales que AWS CLI utiliza son las esperadas.

Para obtener información sobre la asignación de IAM permisos, consulte [Descripción general de la administración del acceso: permisos y políticas](#) en la Guía del IAM usuario.

[\(Volver arriba\)](#)

## Errores de credenciales y claves no válidas

Ejemplo de texto de error:

```
$ aws s3 ls
```

```
An error occurred (InvalidAccessKeyId) when calling the ListBuckets operation: The AWS
Access Key Id
you provided does not exist in our records.
```

```
$ aws s3 ls
```

```
An error occurred (InvalidClientTokenId) when calling the ListBuckets operation: The
security token
included in the request is invalid.
```

Causa posible: AWS CLI está leyendo credenciales incorrectas o proviene de una ubicación inesperada

AWS CLI Es posible que estén leyendo las credenciales de una ubicación diferente a la que esperabas o que la información de tu key pair sea incorrecta. Puede ejecutar `aws configure list` para confirmar las credenciales que se utilizan.

En el siguiente ejemplo se muestra cómo comprobar las credenciales utilizadas para el perfil predeterminado.

```
$ aws configure list
```

Name	Value	Type	Location
profile	<not set>	None	None
access_key	*****XYVA	shared-credentials-file	
secret_key	*****ZAGY	shared-credentials-file	
region	us-west-2	config-file	~/.aws/config

En el siguiente ejemplo se muestra cómo comprobar las credenciales de un perfil con nombre.

```
$ aws configure list --profile saanvi
```

Name	Value	Type	Location
profile	saanvi	manual	--profile
access_key	*****	shared-credentials-file	
secret_key	*****	shared-credentials-file	
region	us-west-2	config-file	~/.aws/config

Para confirmar los detalles de su par de claves, compruebe sus archivos `config` y `credentials`. Para obtener más información sobre los archivos `config` y `credentials`, consulte [the section called “Opciones de los archivos de configuración y credenciales”](#). Para

obtener más información sobre credenciales y autenticación, incluida la precedencia de las credenciales, consulte [Credenciales de autenticación y acceso](#).

[\(Volver arriba\)](#)

Causa posible: el reloj del equipo no está sincronizado

Si está utilizando credenciales válidas, puede que el reloj no esté sincronizado. En Linux o macOS, ejecute `date` para verificar la hora.

```
$ date
```

Si el reloj del sistema no es correcto en unos minutos, utilice `ntpd` para sincronizarlo.

```
$ sudo service ntpd stop
$ sudo ntpdate time.nist.gov
$ sudo service ntpd start
$ ntpstat
```

En Windows, utilice las opciones de fecha y hora del Panel de control para configurar el reloj del sistema.

[\(Volver arriba\)](#)

## Errores de firma que no coincide

Ejemplo de texto de error:

```
$ aws s3 ls
An error occurred (SignatureDoesNotMatch) when calling the ListBuckets operation: The
request signature we
calculated does not match the signature you provided. Check your key and signing
method.
```

Cuando AWS CLI ejecuta un comando, envía una solicitud cifrada a los AWS servidores para que realicen las operaciones de AWS servicio adecuadas. Sus credenciales (la clave de acceso y la clave secreta) participan en el cifrado y permiten autenticar AWS a la persona que realiza la solicitud. Tal y como se indica a continuación, hay varias cosas que pueden interferir con el funcionamiento correcto de este proceso.

## Causa posible: el reloj no está sincronizado con los servidores AWS

Para protegerse de los [ataques de reproducción](#), puede utilizar la hora actual durante el proceso de cifrado y descifrado. Si la hora del cliente y el servidor no coinciden en una cantidad superior a la permitida, puede que el proceso produzca un error y se rechace la solicitud. Esto también puede ocurrir cuando ejecuta un comando en una máquina virtual cuyo reloj no está sincronizado con el reloj de la máquina host. Una causa posible es cuando la máquina virtual está en hibernación y tarda algún tiempo en reactivarse para sincronizar el reloj con la máquina host.

En Linux o macOS, ejecute `date` para verificar la hora.

```
$ date
```

Si el reloj del sistema no es correcto en unos minutos, utilice `ntpd` para sincronizarlo.

```
$ sudo service ntpd stop
$ sudo ntpdate time.nist.gov
$ sudo service ntpd start
$ ntpstat
```

En Windows, utilice las opciones de fecha y hora del Panel de control para configurar el reloj del sistema.

[\(Volver arriba\)](#)

Causa posible: el sistema operativo está manejando mal AWS las claves que contienen ciertos caracteres especiales

Si AWS las claves incluyen ciertos caracteres especiales, como `-`, `,` o algunas variantes del sistema operativo, procesa la cadena de forma incorrecta y provoca que la cadena clave se interprete de forma incorrecta. `+` /

Si procesa las claves con otras herramientas o scripts, como las herramientas que crean el archivo de credenciales en una nueva instancia como parte de su creación, es posible que esas herramientas y scripts tengan su propio manejo de caracteres especiales, lo que provoca que se transformen en algo que ya AWS no reconozca.

Sugerimos regenerar la clave secreta para obtener una que no incluya el carácter especial que causa problemas.

[\(Volver arriba\)](#)

## Errores de no se ha encontrado la consola de Windows

Ejemplo de texto de error:

```
$ aws s3 ls
No Windows console found. Are you running cmd.exe?
```

Al utilizar un AWS CLI comando, aparece el mensaje «No se ha encontrado ninguna consola de Windows». ¿Está ejecutando cmd.exe?». Esto suele ser un error que se ve en la AWS CLI versión 1 si el Python `prompt_toolkit` que ha instalado está desactualizado. Para resolver este problema, instale una versión reciente del `prompt_toolkit` en el [sitio web de Python](#).

[\(Volver arriba\)](#)

## SSLerrores de certificado

Causa posible: AWS CLI no confía en el certificado de su proxy

Ejemplo de texto de error:

```
$ aws s3 ls
[SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed
```

Cuando utilizas un AWS CLI comando, recibes un mensaje `[SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed` de error. Esto se debe a que AWS CLI no confías en el certificado de tu apoderado debido a factores como el hecho de que el certificado de tu apoderado esté autofirmado y que tu empresa sea la autoridad de certificación (CA). Esto AWS CLI impide encontrar el certificado raíz de CA de su empresa en el registro de CA local.

Para solucionar este problema, indique AWS CLI dónde encontrar el archivo de su empresa mediante el ajuste del `.pem` archivo de [ca\\_bundle](#) configuración, la opción de línea de [--ca-bundle](#) comandos o la variable de [AWS\\_CA\\_BUNDLE](#) entorno.

[\(Volver arriba\)](#)

Causa posible: la configuración no apunta a la ubicación correcta del certificado raíz de la CA

Ejemplo de texto de error:

```
$ aws s3 ls
```

```
SSL validation failed for regionname [Errno 2] No such file or directory
```

Esto se debe a que la ubicación del archivo del paquete de la autoridad de certificación (CA) está configurada incorrectamente en la AWS CLI. Para solucionarlo, confirme dónde se encuentra su archivo `.pem` de su empresa y actualice la configuración de la AWS CLI mediante la opción del archivo de configuración [ca\\_bundle](#), la opción de la línea de comandos `--ca-bundle` o la variable de entorno [AWS\\_CA\\_BUNDLE](#).

[\(Volver arriba\)](#)

Causa posible: su configuración no utiliza la correcta Región de AWS

Ejemplo de texto de error:

```
$ aws s3 ls  
[SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed
```

Se pueden producir errores o resultados inesperados si una no Servicio de AWS está disponible para la información especificada Región de AWS o si los recursos están ubicados en una ubicación diferente Región de AWS. Para ver los pasos de solución de problemas, consulte [the section called “Compruebe si Región de AWS su AWS CLI comando está utilizando”](#).

[\(Volver arriba\)](#)

Causa posible: es necesario actualizar tu TLS versión

Ejemplo de texto de error:

```
$ aws s3 ls  
[SSL: UNSAFE_LEGACY_RENEGOTIATION_DISABLED] unsafe legacy renegotiation disabled
```

Servicio de AWS Está utilizando una versión TLS que no es compatible con la TLS versión de su dispositivo. Para resolver este problema, actualiza a una TLS versión compatible. Para obtener más información, consulte [the section called “Imponer una versión mínima TLS”](#).

[\(Volver arriba\)](#)

## JSONErrores no válidos

Ejemplo de texto de error:

```
$ aws dynamodb update-table \  
  --provisioned-throughput '{"ReadCapacityUnits':15,WriteCapacityUnits':10}' \  
  --table-name MyDDBTable  
Error parsing parameter '--provisioned-throughput': Invalid JSON: Expecting property  
name enclosed in  
double quotes: line 1 column 25 (char 24)  
JSON received: {"ReadCapacityUnits":15,WriteCapacityUnits":10}
```

Cuando utiliza un AWS CLI comando, recibe un mensaje de error `Invalid JSON` «». Por lo general, se trata de un error que se produce cuando se introduce un comando con el JSON formato esperado y AWS CLI no se puede leer JSON correctamente.

Causa posible: no has introducido un valor válido JSON AWS CLI para su uso

Confirme que ha JSON introducido un código válido para su comando. Te sugerimos que utilices un JSON validador JSON si tienes problemas con el formato.

Para un JSON uso más avanzado en la línea de comandos, considera usar un JSON procesador de línea de comandos, por ejemplo `jq`, para crear JSON cadenas. Para obtener más información `jq`, consulte el [repositorio jq](#) en GitHub.

[\(Volver arriba\)](#)

Causa posible: las reglas de cotización de su terminal impiden que JSON se envíe un documento válido al AWS CLI

Antes de AWS CLI recibir algo de un comando, tu terminal procesa el comando utilizando sus propias reglas de comillas y escape. Debido a las reglas de formato de un terminal, es posible que parte del JSON contenido se elimine antes de pasar el comando al. AWS CLI Al formular comandos, asegúrese de usar sus [reglas de entrecomillado del terminal](#).

Para solucionar problemas, utilice el comando `echo` para ver cómo el shell gestiona los parámetros:

```
$ echo {"ReadCapacityUnits":15,"WriteCapacityUnits":10}  
ReadCapacityUnits:15 WriteCapacityUnits:10
```

```
$ echo '{"ReadCapacityUnits":15,"WriteCapacityUnits":10}'  
{"ReadCapacityUnits":15,"WriteCapacityUnits":10}
```

Modifique el comando hasta que se devuelva el valor «hasta que JSON sea válido».



Para obtener una solución más detallada, use el parámetro `--debug` para ver los registros de depuración, ya que muestran exactamente lo que se pasó a la AWS CLI:

```
$ aws dynamodb update-table \  
  --provisioned-throughput '{"ReadCapacityUnits':15,WriteCapacityUnits':10}' \  
  --table-name MyDDBTable \  
  --debug  
2022-07-19 22:25:07,741 - MainThread - awscli.clidriver - DEBUG - CLI version: aws-  
cli/1.18.147  
Python/2.7.18 Linux/5.4.196-119.356.amzn2int.x86_64 botocore/1.18.6  
2022-07-19 22:25:07,741 - MainThread - awscli.clidriver - DEBUG - Arguments entered  
to CLI:  
['dynamodb', 'update-table', '--provisioned-throughput',  
'{"ReadCapacityUnits":15,WriteCapacityUnits":10}',  
 '--table-name', 'MyDDBTable', '--debug']
```

Utilice las reglas de cotización de su terminal para solucionar cualquier problema que tenga su JSON entrada al AWS CLI. Para obtener más información sobre las reglas de entrecomillado, consulte [the section called “Comillas con cadenas”](#).

#### Note

Si tiene problemas JSON para obtener la validez AWS CLI, le recomendamos que evite las reglas de cotización de un terminal para la entrada de JSON datos y utilice Blobs para pasar sus JSON datos directamente al. AWS CLI Para obtener más información acerca de Blobs, consulte o [Blob](#).

[\(Volver arriba\)](#)

## Recursos adicionales de

[Para obtener ayuda adicional con tus AWS CLI problemas, visita la AWS CLI comunidad de GitHubo la AWS re:Post comunidad.](#)

[\(Volver arriba\)](#)

# Historial de la documentación de la guía del usuario de AWS CLI

En la siguiente tabla se describen los cambios importantes de la Guía del usuario de AWS Command Line Interface, a partir de enero de 2019. Para obtener notificaciones sobre las actualizaciones de esta documentación, puede suscribirse a la fuente RSS.

Cambio	Descripción	Fecha
<a href="#">Información de credenciales y autenticación actualizada.</a>	Instrucciones y ejemplos de credenciales y métodos de autenticación actualizados. Esto incluye la actualización de las páginas de configuración pertinentes. Para dar cabida a este aumento de la documentación, los temas relevantes de las credenciales se trasladaron a la nueva sección <a href="#">Autenticación y credenciales de acceso</a> .	31 de marzo de 2023
<a href="#">El contenido de la V1 y V2 de AWS CLI ahora están separados en sus respectivas guías</a>	Para mayor claridad y facilidad, la versión 1 de AWS CLI y la versión 2 de AWS CLI ahora están separadas en sus propias guías. Para la versión 2 de AWS CLI, consulte la <a href="#">Guía del usuario de AWS Command Line Interface</a> más reciente	2 de noviembre de 2021
<a href="#">Se ha agregado información de alias de AWS CLI</a>	Se añadió información de alias de la AWS CLI. Los alias son accesos directos que se pueden crear en la	11 de marzo de 2021

AWS Command Line Interface (AWS CLI) para acortar los comandos o los scripts que utiliza con frecuencia.

[Se actualizó la información de salida del filtro](#)

Se actualizó la información de los filtros y se trasladó a su propia página.

1 de febrero de 2021

[Anuncio de obsolescencia de Python 2.7, 3.4 y 3.5](#)

Python 2.7 quedó obsoleto debido a Python Software Foundation desde el 1 de enero de 2020. En el futuro, los clientes que utilizan la versión 1 de la AWS CLI deberán usar Python 3, como mínimo Python 3.6. El soporte de Python 2.7 está obsoleto para las nuevas versiones de la versión 1 de la AWS CLI a partir del 19/7/2021. Python 3.4 y 3.5 está obsoleto a partir del 1/2/2021.

29 de enero de 2021

[Se ha agregado un ejemplo de scripting de Amazon S3](#)

Se agregó un ejemplo de scripting del ciclo de vida de Amazon S3.

15 de octubre de 2020

[Se ha agregado un ejemplo de scripting de Amazon EC2](#)

Se agregó un ejemplo de scripting de tipo de instancia de Amazon EC2.

15 de octubre de 2020

[Se ha agregado información de reintentos](#)

Se agregó una página de reintentos para las características y el comportamiento de los reintentos en la AWS CLI.

17 de septiembre de 2020

<a href="#">Página de paginación en el cliente y en el servidor</a>	Se actualizó la información de paginación y se centralizó en una sola página.	17 de agosto de 2020
<a href="#">Se actualizó la página de comandos de S3</a>	Se actualizó la página de comandos s3 de alto nivel con nuevos ejemplos y recursos.	30 de julio de 2020
<a href="#">Información de instalación actualizada</a>	Se ha actualizado la información de instalación, actualización y desinstalación de Linux, macOS y Windows.	19 de mayo de 2020
<a href="#">Se ha actualizado para eliminar la compatibilidad con Python 2.6 y 3.3 de la versión 1 de AWS CLI</a>	A partir del 10 de enero de 2020, la versión 1 de la AWS CLI ya no admite el uso de las versiones 2.6 o 3.3 de Python. Debe actualizar a una versión más reciente de Python para utilizar la versión 1.17 o posterior de la AWS CLI.	10 de enero de 2020
<a href="#">Nueva sección de MFA</a>	Se ha añadido una nueva sección que describe cómo obtener acceso a la CLI mediante la autenticación multifactor y los roles.	3 de mayo de 2019
<a href="#">Se ha actualizado la sección "Uso de la CLI"</a>	Se han mejorado y añadido instrucciones y procedimientos de uso.	7 de marzo de 2019
<a href="#">Se ha actualizado la sección "Instalación de la CLI"</a>	Se han mejorado y agregado instrucciones y procedimientos de instalación de la AWS CLI.	7 de marzo de 2019

[Se ha actualizado la sección  
"Configuración de la CLI"](#)

Se han mejorado y agregado instrucciones y procedimientos de configuración de la AWS CLI.

7 de marzo de 2019